Project acronym:	*SafeAdapt*
Project title:	*Safe Adaptive Software for Fully Electric Vehicles*
Grant Agreement number:	608945
Coordinator:	*Dr.-Ing. Dirk Eilers*
Funding Scheme:	*FP7-2013-ICT-GC*

Deliverable 5.1

Evaluation Methodology for the SafeAdapt Results

Due date of deliverable:	June 30, 2015
Actual submission Date:	August 28, 2015
Lead beneficiary for this deliverable:	Fraunhofer ESK

Dissemination level		
PU	**Public**	**X**
PP	**Restricted to other programme participants (including the Commission Services)**	
RE	**Restricted to a group specified by the consortium (including the Commission Services)**	
CO	**Confidential, only for members of the consortium (including the Commission Services)**	

The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013)

Document Information	
Title	Evaluation Methodology for the SafeAdapt Results
Creator	Fraunhofer: Christian Drabek, Philipp Schleiss, Gereon Weiss
Description	The document contains a detailed description of the methodologies and metrics used for the evaluation of SafeAdapt results
Publisher	Fraunhofer ESK
Contributors	CEA: Ansgar Rademacher, Önder Gürcan Delphi: Timo Feismann, Thorsten Rosenthal Duracar: Ken Lam Siemens: Cornel Klein, Jan Sawallisch, Andre Marek, Marc Zeller Tecnalia: Alejandra Ruiz, Mª Carmen Palacios, Garazi Juez, Iñaki Iglesias, Adrián Martín, Maite Álvarez TTTech: Andreas Eckel
Language	en-GB
Creation date	06.03.2015
Version number	0.4
Version date	28.08.2015
Audience	☐ internal ☒ public ☐ restricted

Table of Contents

List of Figures

List of tables

Executive Summary

In order to evaluate the concepts developed during the course of the SafeAdapt project, this document defines a sound set of metrics to measure the quantitative and qualitative performance of the project's novel approach for safe adaptation.

For this, strong emphasis was laid on developing a concept for safe adaptation that is based on industry-driven use-cases, which were already defined in the early phases of the project in deliverable 2.1. Consequently, this document builds upon these results by assigning each use case to a least one of the project's prototype platforms. Moreover, a detailed technical description is developed to define how these use-case-based goals can be evaluated within the respective prototype, thus safe-guarding the general applicability of the adaptation concepts.

In addition to this, the SafeAdapt project has defined quantitative goals with respect to expectable benefits when applying this novel method for safe adaptation in future e-vehicles. As such, this document provides an in-depth definition of different metrics and methods that will be utilised to determine how well these expected targets are met. Based on this, Deliverable 5.3 will then evaluate the concept's effectiveness for each use-case within the respective prototype and further apply the metrics and methods to then determine what benefits can be expected when migrating to this new type of E/E-architecture.

1 Introduction

Evaluation is a tool to assess the properties of the subject under investigation and make them measurable. This is done in an objective manner as possible. It is crucial to reflect on the results of a project in order to determine its value. Therefore, evaluation has been an integral part in the project plan of SafeAdapt. To be able to examine our approach in an objective manner, we chose our evaluation methods before implementing the prototype. The results of this decision and the methods themselves are described in this document.

In this document, the different methodologies and metrics for the evaluation of SafeAdapt results are specified. For the verification of the improvements in reliability, efficiency (e.g., energy, costs, etc.), and flexibility of Fully Electric Vehicles (FEVs) gained by the SafeAdapt results, specific evaluation methodologies are developed. The evaluation methodologies shall contemplate the validation and verification techniques for both direct functionality and associated safety functions. This should include at least a validation of requirements following traceability, Model-in-the-Loop for software models and algorithms, Software-in-the-Loop for implemented software, Hardware-in-the-Loop for integrated Hardware-Software, and a validation of X-by-Wire (Brake-by-Wire and Steer-by-Wire) and ADAS functionalities w.r.t. runtime behaviour and safety requirements. Moreover, it will be decided which of the scenarios and use cases specified in Deliverable 2.1 will be evaluated on which demonstrator prototype. Deliverable 5.3 will report the results of the evaluation that is described in this document.

For the evaluation and demonstration of the SafeAdapt results, all partners contribute their own tools, platforms, and applications to evaluate the project results. Therefore, the SafeAdapt consortium will set up a full scale e-vehicle prototype to assess the results of the project. The experience for building a prototype e-vehicle using the SafeAdapt results will be contributed by Duracar, Siemens, and Pininfarina. All partners are part of the dissemination and exploitation. The results of SafeAdapt are foreseen to encompass market impact by introducing results in correlating standardisations, like for example AUTOSAR.

This document is structured as follows: First the goals of the evaluation based on the project's objectives are summarised. Chapter 3 describes how the viability of the project's approach will be evaluated using different demonstrator prototypes. In Chapter 4 metrics to measure the efficiency of the SafeAdapt approach are defined. Finally, the evaluation strategy is summarised.

2 Goals

The goal of this document is to provide a strategy to evaluate the results of the SafeAdapt project. The results are reflected with regard to the objectives that are defined in the Description of Work. Therefore, this section gives a brief summary of the objectives of the project.

As the project is driven by both use-cases and predefined targets with respect to enhanced safety and reductions in complexity as well as unit and development cost, the goals are also two-fold.

Respectively, the first goal is to show that the developed approach for safe adaptation is viable and can be used to fulfil the functional aspects of the project's objectives, which are:

- Objective #1: Provide novel architecture concepts to enhance robustness, availability, and efficiency of safety-relevant systems while preserving the functional safety in fully electric vehicles
- Objective #2: Increase safety and availability through the ability to handle complex failures, especially failures where current systems do not degrade gracefully
- Objective #3: Reduced bill of material by reducing the number of ECUs by providing a generic failure management based on the SafeAdapt Platform Core
- Objective #4: Reduced development costs (time-to-market & testing costs) in future FEVs by providing a generic failure management and software update mechanism (dealer retrofit) based on a SafeAdapt Platform Core
- Objective #5: Increasing energy efficiency in automotive E/E architectures

Secondly, the required effort to apply the project's approach for safe adaptation must be evaluated with respect to state-of-the-art approaches. Therefore, the following major measurable results have been defined in the project descriptions. Metrics are defined that allow for a comparison of the efficiency in measurable quantities. The following non-functional aspects of the objectives are evaluated by the measurable results (MR):

- MR1: Optimise energy consumption of safety-relevant features by up to 30%
- MR2: Handle 20-30% of failures in safety-relevant systems through adaptation and reconfiguration
- MR3: Reduce development and testing costs by up to 20%
- MR4: Reduce certification cost by up to 20%
- MR5: Reduce complexity and hardware cost of safety-relevant systems by up to 20%
- MR6: Require 50% less extra ECUs than simple duplication of ECUs and still meet the safety requirements

With this two-fold evaluation we will be able to show that our approach for safe adaptation is both viable and efficient.

3 Strategy for Evaluating Viability

This chapter describes the evaluation methods used to verify the viability of the SafeAdapt approach. First a short overview of the platforms, that are used as basis for the demonstrators, i.e., RACE car and Dynacar, is provided. For each of these platforms, a more detailed description is given, as to why the platform is suitable for evaluation, and what will be demonstrated with the respective platform. For the demonstration, use cases from Deliverable 2.1 are selected that will be implemented. For each of these realised use cases, a detailed description is provided on how it will be implemented and evaluated.

3.1 Demonstrator Platforms

In general, the RACE car will be used to validate use-cases related to redundancy management, whereas the Dynacar will address use-cases concerned with energy management and vehicle-dynamics with respect to failover times. Moreover, a so called “SafeCar”, which is a conceptual model-car representation of the RACE car, will be used to demonstrate the feasibility of use-cases addressing hardware and software updates. The use-cases are summarised as described in Deliverable 2.1 in Table 1. A detailed description of the relationship between the respective demonstrator and use cases is provided in the following chapters.

Use Case	Prototype/ Demonstrator	Description
UC_110_01: Reconfiguration of Failed Cruise Control (Cold-Standby)	RACE	The driver is driving the vehicle with ACC (Adaptive Cruise Control) functionality on a road following another vehicle. Suddenly the vehicle in front rapidly decelerates so that the driver is in danger of a possible rear-end collision. The ACC function should react to slow down the vehicle but a malfunction in the ACC software component occurs in that moment. To reach a safe state, the vehicle should adapt the ACC function.
UC_111_01: Steer-by-Wire Adaptation after ECU-Failure (Core Node-Failover)	RACE	The vehicle is driving on a main road. A short circuit leads to an immediate failure of a core node.
UC_114_01: Adaptation after Brake-by-Wire Malfunction (Dependable Function)	RACE	The vehicle is driving on a road while the SomnoAlert function is monitoring the driver alertness. An obstacle appears on the road and the AEB (Automatic Emergency Brake) has to be activated to avoid a collision but the BbW (Brake-by-Wire) function does not react. Involved functions have different ASIL classification

		(A, C, and D). After the failure, not enough computing resources are available to keep all functions running. A safe state must be reached by passivating the least critical function (SomnoAlert) in order to maintain AEB and BbW functionality.
UC_116_01: Communication Failure with External Aggregate (Hot-Standby)	RACE	The vehicle is driving with an ACC function on a road following another vehicle. Suddenly the vehicle in the front rapidly decelerates, so that the driver is in danger of a possible rear-end collision. The ACC function should react to slow down the vehicle but a malfunction in sensor communication to the ACC is detected at this moment. The receiver will decide which path of the network is used.
UC_211_01: Installation of New Component	RACE (SafeCar)	The driver wants to upgrade the vehicle by installing a new component at an official maintenance service provider. Once in the garage, the maintenance service provider proceeds to perform all required overhaul operations.
UC_311_01: Update of Function	RACE (SafeCar)	The driver wants to upgrade his vehicle by installing new software at an official maintenance service provider. Once in the garage, the maintenance service provider proceeds to perform all required overhaul operations.
UC_411_01, Degradation of Steer-by-Wire Application	Dynacar	The vehicle is driving on a road and the SBW application experiments a failure while operating in regular mode so that it is not able to function correctly. As a SBW basic application is available to work in a degraded mode, it has to be activated to reach a safe state.
UC_511_01, Adaptation for Range Extension	Dynacar	The vehicle is driving on the road and the BMS detects a SOC (State Of Charge) of under 35%. In this case the use of energy has to be prioritised to increase the remaining range. The following adaptation will be performed: the BMS cuts off supply for auxiliary services and the IWM performance (torque and power) are reduced (50%). The BbW system activates rules to maximise the

		regenerative braking; during braking events the utilisation of the generator should be as large as possible to reduce dissipation of kinetic energy in form of heat.

Table 1: Mapping of use-cases to demonstrators

3.2 RACE Car Demonstrator

Depending on the characteristics of the application it can be integrated into a real vehicle platform or a virtual one. These platforms are generally considered to be two independent configurations:

- Vehicle Configuration: Consists of applications and systems that will be integrated onto a real vehicle (in this case the Siemens RACE vehicle). There might be more than one vehicle configuration, if the applications or system need to be tested separately.
- Simulated Configuration: Consists of applications which are problematic to be integrated into a real vehicle. These application may however still be simulated to avert the risk associated with integrating experimental functions into a real vehicle. There might be more than one simulated configuration, if applications or systems need to be tested separately.

3.2.1 Platform Description

The RACE car (s. Figure 1) has been developed within the German national funded project RACE (Robust and reliable Automotive Computing Environment for future eCars, see www.projekt-race.de). The goal is to demonstrate the software- and system-architecture developed in that project in a real working car, which could potentially be certified for usage on public roads. All functions, such as braking, driving, or steering, can be realised by the RACE architecture using a centralised computing platform.

Figure 1: The RACE Demonstrator Car

The car is built using a small-series commercial car from Roding Automobile (see http://www.roding-automobile.de). Its main characteristics are:

- Fully electric drive train with wheel hub motors: The two wheel hub motors on the rear axle provide a continuous power of 63kw and a peak power of 115kw. The maximum torque is 1250 Nm and the continuous torque is 500 Nm. This power is mainly used to recuperate a high degree of the energy during braking in order to enhance energy efficiency and fine dust emissions. That way, more than 70% of braking situations can be performed using only electric machines.
- Electrical controlled braking system, which is fully controlled by the RACE platform. “Brake Blending” is a function which tries to distribute the required braking force between the braking system and the wheel hub motors. Only in cases when the braking power of the wheel hub motors is insufficient, the electronic braking system is used.
- A full Steer-by-Wire system with redundant steering actors and sensors, implemented on top of the RACE E/E architecture. In contrast to commercially available products, e.g., from Nissan, the Steer-by-Wire system does not have a mechanical backup. High availability and hence safety is provided only by the redundant E/E system.
- A multitude of sensors like LIDAR, ultrasound, and cameras which can be used to develop and to demonstrate new automotive functions such as advanced drivers assistance systems or functions for autonomous driving. These are mostly off-the-shelf sensors, which are connected to the RACE system via gateways.

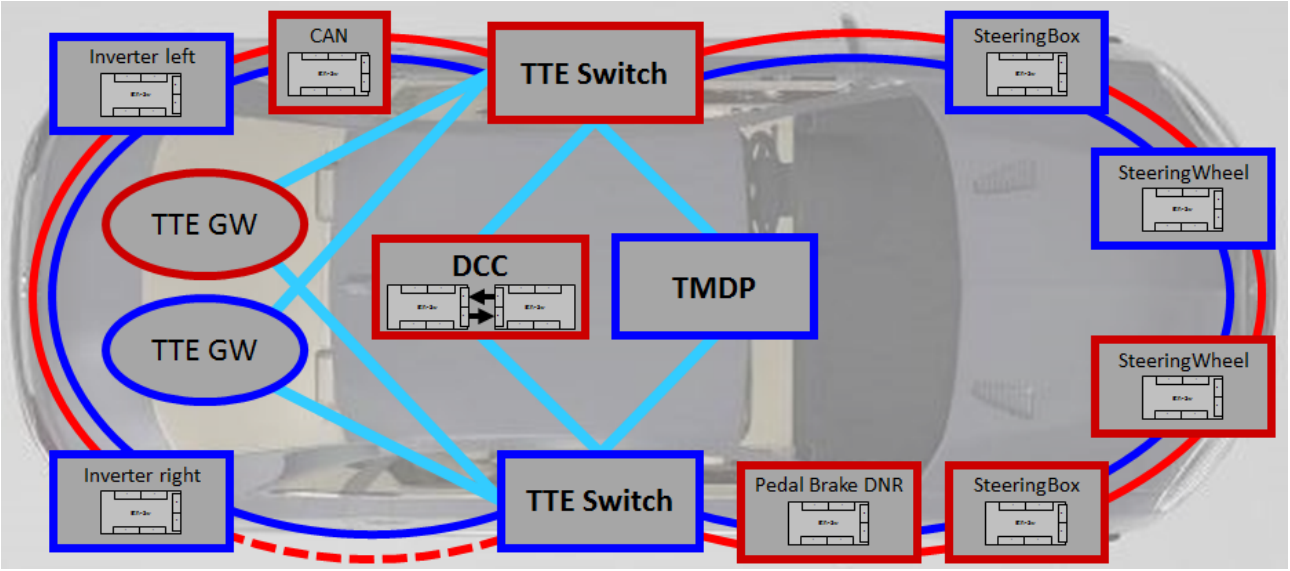

Figure 2: Redundant Power Supply, Computation & Communication of RACE Car

The RACE demonstrator car is enhanced with a fully redundant E/E architecture conforming to the SafeAdapt concepts (cf. Figure 2). Safety critical systems, like the Steer-by-Wire system or the pedal box, are supplied by two different power circuits. Likewise, the communication architecture of the car is fully redundant, using an Ethernet ring structure. To reduce cabling, less critical components like parking sensors can be connected in a non-redundant way. In SafeAdapt, the car is equipped with additional components of the partners (e.g., diverse TTTech TTE components). The central computing platform consists of a Siemens’ Duplex Control Computer (DCC) and Delphi’s Trusted Multi Domain Platform (TMDP) in order to provide a fail-operational system.

3.2.2 Use-Cases concerned with Failure / Redundancy Management

A test system already exists in the environment of the RACE project. In this model, each node (each RACE DCC and each RACE gateway) operates as a test server. An external computer is connected as test client via separate Ethernet wires and runs software to monitor and manipulate the system. Figure 3 shows the hardware-relevant setup where the test computer (test client) is connected with the RACE nodes (test servers).

During run time the test system will be served in each node at the end of each cycle (cycle duration 10 milliseconds). All status and user data transferred between all RTE modules (i.e., I/O management, platform management) and all applications are stored in an internal database which can be accessed by the test client. With this method the status of the system at the end of each cycle will be visible. The test client is able to display all this data as actual value and/or as time line visualization with graphical plots (monitoring). In the same manner, it is possible to modify dedicated values in this database, which enables the simulation of a new state at the beginning of the next cycle (manipulation). The relevant node has to react on this manipulation and at the end of the cycle the expected result can be verified by monitoring. Figure 4 shows the chronological sequence of a single cycle and the schedule of the test system for monitoring and manipulation.

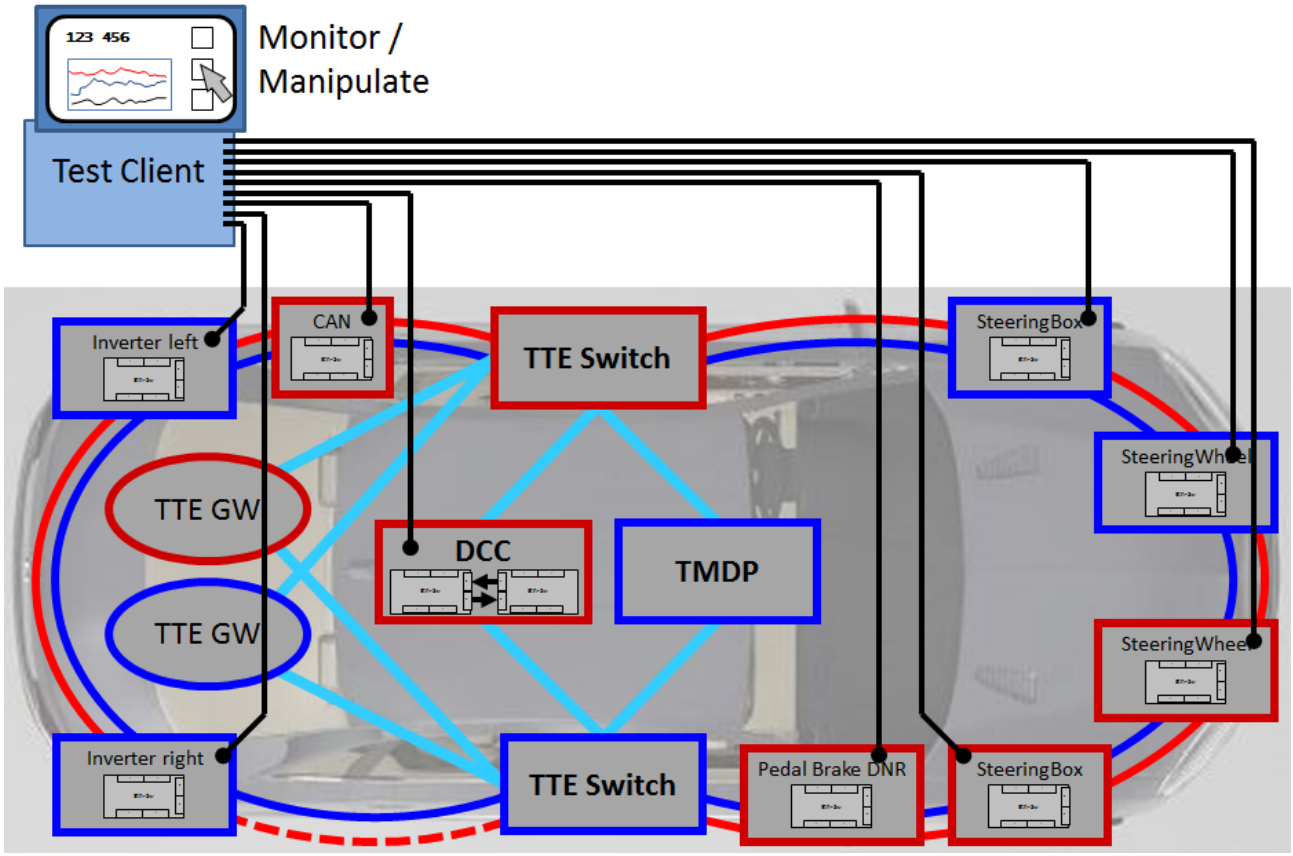

Figure 3: Setup with RACE Car and test system (Test client and several test servers)

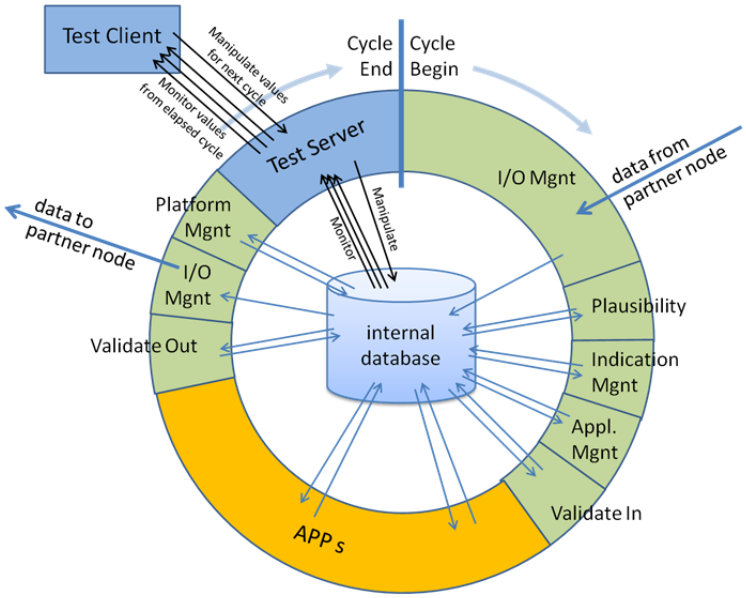

Figure 4: Chronological sequence of a single cycle with test system

This test system can be used in the scope of the SafeAdapt project, too. Monitoring and manipulation are restricted to RACE nodes. However, this is sufficient for the use-cases which have to be analysed.

3.2.2.1 Use case related to Reconfiguration of Failed Cruise Control (Cold Standby)

The use case UC_110_01, “Reconfiguration of Failed Cruise Control (Cold Standby)”, shows the safe operation of Adaptive Cruise Control (ACC) during a malfunction. For this test, the following conditions are given: A situation where an automatic deceleration based on a preceding braking car proceeds and during this action malfunction of ACC happens. The expected result is a speed adaptation of vehicle to follow the preceding car in a safe manner.

With the given test system (described in chapter above) a preceding braking car will be simulated for the ACC sensor system. The ACC software is running on the RACE DCC, a cold standby is available on the TMDP. Required resources for the ACC are available on the TMDP. The malfunction of ACC application will be realized by the test system through isolation of the ACC on the RACE DCC. This is possible by a manipulation of the application status value from active to passive/isolated. Provided that a proper configuration is implemented at all relevant nodes, the switch to the cold-standby-element will occur.

The use-case succeeds, if the ACC sensor values and the dynamic behaviour values of the vehicle reported by the test system indicate a speed adaptation of the vehicle to follow the preceding car in a safe manner. This is the case, if the distance between the two relevant cars measured by ACC sensors is always greater than a safe distance and no rapid acceleration or deceleration is performed, unless necessary. Furthermore, in a time less than 50 milliseconds the new designated application (previously cold standby) runs properly and delivers values for decreasing speed. As the ACC on the RACE DCC becomes isolated, this will only be the case, if the cold standby on the TMDP successfully activates.

3.2.2.2 Use case related to Steer-by-Wire Adaptation after ECU-Failure (Core Node Failover)

The use case UC_111_01, “Steer-by-Wire Adaptation after ECU-Failure (Core Node Failover)”, has been selected as representative for Failure / Redundancy Management. The vehicle is driving on a main road. A short circuit leads to an immediate failure of a core node. The goal is to check that the functionality of SbW and ACC is restored before the driver loses control of the vehicle.

Both core nodes are connected onto two different power supplies, and switched on. An active SbW is running on the RACE DCC, a hot-backup for SbW is running on the TMDP. The sensor signal (steering wheel angle) should be simulated as a continuously changing signal, as only an active steering process is useful for this use-case. Therefore, the steering wheel sensor input will be implemented with a signal generator, i.e., a triangle curve, to guarantee a changing input value as simulation for continuously steering. The actor (steering box) should receive the given signal and displays it. An injected failure of the RACE DCC triggers a change of the SbW on the TMDP into active mode. The failure injection can be realized by manual interaction or an automatically initiated interrupt. The steering actor (steering box) displays continuously receiving value of steering angle to demonstrate equivalent movement.

The use-case succeeds, if the receiving steering actor displays the steering angle as a triangle curve, too. In-between the injected failure into the RACE DCC and the performed hand-over to the TMDP, the duration of a measurable interruption of the continuous dataflow is less than 50 milliseconds.

3.2.2.3 Use case related to Adaptation after Brake-by-Wire Malfunction (Dependable Function)

The use case UC_114_01, “Adaptation after Brake-by-Wire Malfunction (Dependable Function)”, shows the safe operation of Automatic Emergency Brake (AEB) and Brake-by-Wire (BbW) during a malfunction. For this test two conditions are given: A situation where an emergency brake is detected and during this action a malfunction of BbW happens. The expected result is a stop of the vehicle to avoid a collision with a simulated obstacle.

With given test system an emergency brake will be simulated for the AEB system. The BbW software is running on the RACE DCC, a cold standby is available on the TMDP, but no resources are reserved for the BbW. Therefore, non-safety-critical applications on the TMDP must be passivated to free resources. The mentioned malfunction of BbW application will be realized by the test system through isolation of the BbW application on the RACE DCC. This is possible by a manipulation of the application status value from active to passive / isolated. Provided that a proper configuration is implemented at all relevant nodes the reconfiguration of BbW function onto an empty partition will occur.

The use-case succeeds, if the dynamic behaviour values of the vehicle reported by the test system indicate a complete braking-to-stop procedure of the vehicle in a safe manner. This is the case, if the distance between the vehicle and the simulated obstacle is always greater than a safe distance and, in the end, the own vehicle speed is equal to zero. Furthermore, in a time less than 50 milliseconds the new designated application runs properly and performs a safe brake of the vehicle.

3.2.2.4 Use case related to Communication Failure with External Aggregate (Hot-Standby)

The use case UC_116_01, “Communication Failure with External Aggregate (Hot-Standby)”, has been selected as representative of Failure / Redundancy Management. The vehicle is driving on a main road. A malfunction in sensor communication leads to an immediate failure of a core node. The goal is to check that the functionality of ACC is restored before the driver loses control of the vehicle.

Both core nodes are connected to ACC sensors. An active ACC is running on the RACE DCC, a hot-backup for ACC is running on the TMDP. The sensor signal (distance to preceding car) should be simulated as a continuously changing signal. The ACC sensor aggregate should be implemented with a signal generator (i.e., a triangle curve) to guarantee a changing input value as simulation representation for a continuously varying distance between the two relevant vehicles. An injected failure in the sensor communication of the RACE DCC triggers a change of the ACC on the TMDP into active mode. The communication loss can be realized by manually interaction or an automatically initiated interrupt. The actor (inverter for drive) should receive the given signal with the velocity. The value of the velocity is used to demonstrate equivalent movement.

The use-case succeeds, if the receiving actor displays velocity as a derived triangle curve, too. In-between the injected failure into the RACE DCC and the performed hand-over to the TMDP, the duration of a measurable interruption of continuous dataflow is less than 50 milliseconds.

3.2.3 Use-Case concerned with P&P and HW/SW-Updates

3.2.3.1 Use case related to Installation of New Component

The use case UC_211_01, “Installation of New Component (New)” has been selected to demonstrate that new components (SW or HW) can be integrated into the vehicle while the safety requirements of the overall system are preserved. In this use case, the driver wants to upgrade the vehicle by installing a new component at an official maintenance service provider. Once in the garage, the maintenance service provider proceeds to perform all required overhaul operations.

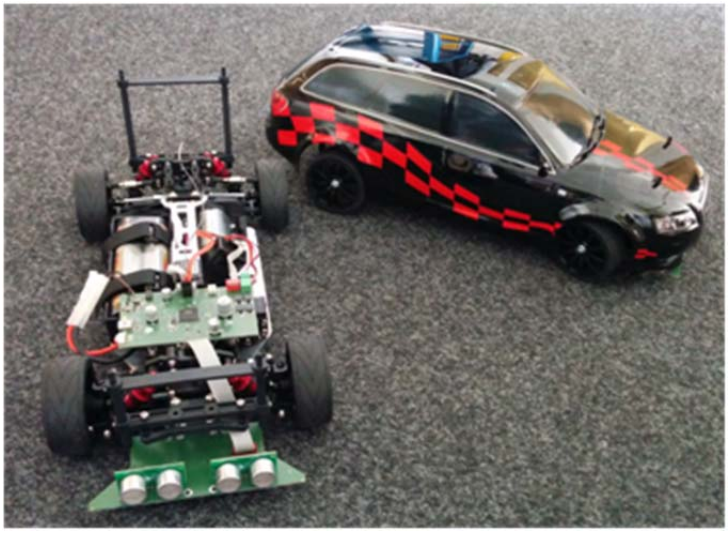

Figure 5: SafeCar demonstrator platform

The goal is to demonstrate that a new component can be integrated into the existing system while the safety requirements are preserved. Therefore, a safety analysis is performed to evaluate that the requirements are still met after the installation of the new component.

The SafeCar demonstrator, as a conceptual representation of the RACE car, is a radio controlled demonstrator vehicle as depicted in Figure 5. The radio controlled car includes two Ultrasonic Distance Sensors at the front of the car (UltrasonicSensor1 & UltrasonicSensor2). Moreover, the vehicle includes an Engine, a Steering, a Radio Receiver, and a Battery. The components RadioReceiver, UltrasonicSensor1, and UltrasonicSensor2 are so-called smart sensors, i.e., they consist both of hardware and software parts. The components Engine and Steering are smart actors, also both with hardware and software parts. The data is processed by an algorithm implemented in software running on a microcontroller.

The SafeCar demonstrator conceptually implements an emergency braking function within the remote control e-vehicle. Its basic functionality is to transmit the steering and throttle commands from the radio receiver to the steering and engine actuators. If one of the ultrasonic sensors detects an obstacle, the throttle and steering signals are no longer forwarded to the actors. The car is set into emergency braking mode instead, where the actors are used to safely brake the car and omit forward moving signals for a certain time span. The actual control logic is implemented by the so-called *Emergency Braking Control (EBC)* component – a software component running on the microcontroller. The EBC controls the engine and the steering to behave according to the signals received by the remote control. If an obstacle is detected by one of the ultrasonic sensors, the vehicle is stopped immediately. Figure 6 shows an overview of the system architecture of the emergency braking function with its components and their interconnections.

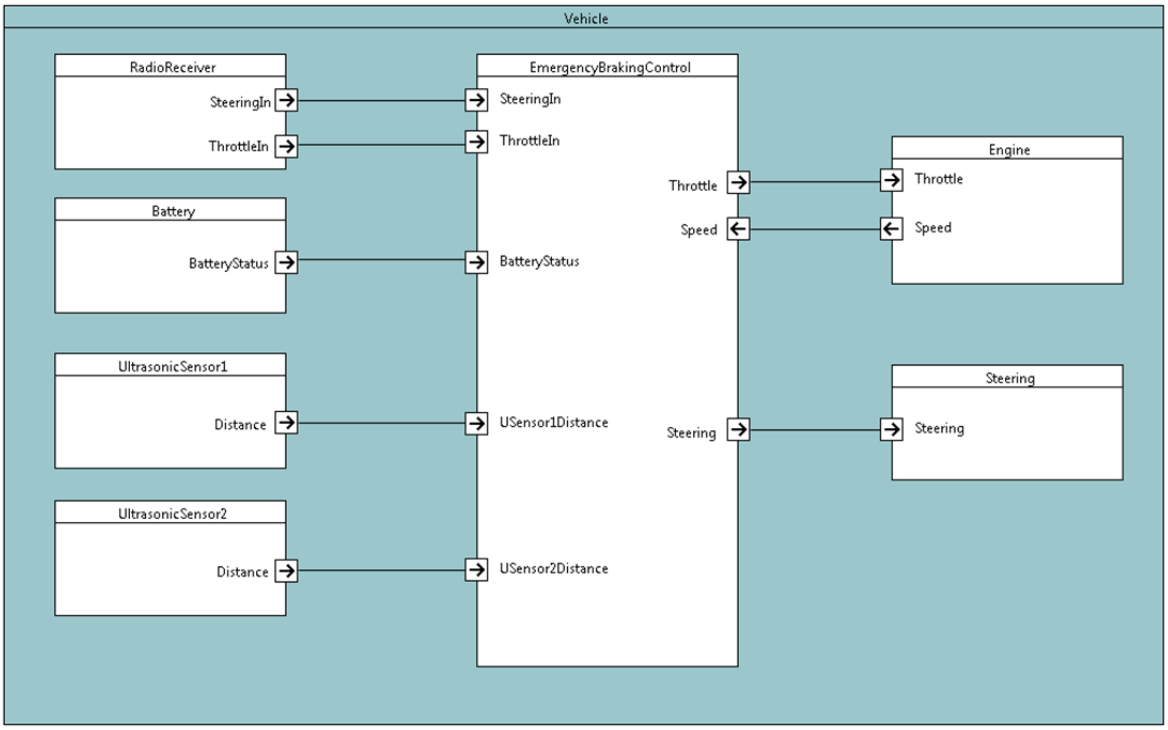

Figure 6: System architecture of the emergency braking function
(SysML Internal Block Diagram)

The case study UC_211_01 “Installation of New Component (New)” is implemented by providing a SCADE system (Papyrus SysML) model to describe the system architecture of the emergency braking function implemented on the radio controlled car. Moreover, a SCADE suite model is used to specify the EBC component which implements the actual control functionality. Based on these models C code is generated to run on the microcontroller of the SafeCar demonstrator.

In order to evaluate whether the systems’ safety requirements are fulfilled, a compositional safety analysis model of the emergency braking function is provided in form of a Component Fault Tree (CFT) using the composR tool. Hence, this compositional and model-based development strategy enables deductive safety analyses of the system in a qualitative as well as a quantitative manner (cf. Figure 7).

The integration of a new component is demonstrated by adding new components in the SCADE system design. The safety analysis model is adjusted accordingly in an automated manner by adding the CFT element of the new component (e.g., from a repository) and integrating it into the existing CFT model. Thereby, immediate feedback in terms of system safety is provided.

The use-case succeeds, if new components can be added to the system design and the system safety is automatically checked. By having the ability to automatically validate the system in terms of safety, we compare the results of the safety analyses before and after a new component is integrated to check whether the system still meets its predefined safety requirements.

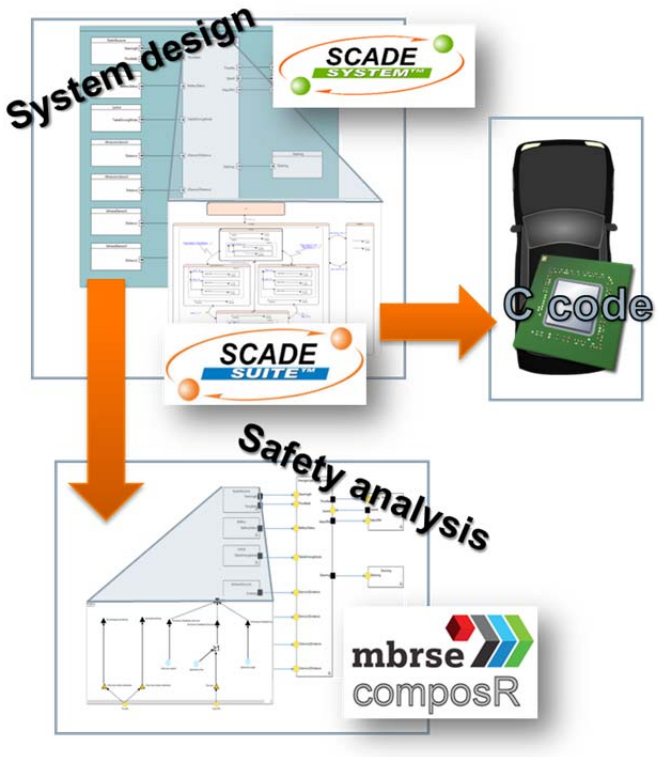

Figure 7: Compositional, model-based development strategy for the use case UC_211_01

3.2.3.2 Use case related to Update of Function

The use case UC_311_01, “Update of Function (Update)” has been selected to demonstrate how new software components can replace existing ones while the vehicle is in the field and the safety requirements of the overall system are preserved.

In this use case the driver wants to upgrade his vehicle by installing new software at an official maintenance service provider. Once in the garage, the maintenance service provider proceeds to perform all required overhaul operations. The goal is to demonstrate that a new version of a software component can replace an existing one while the safety requirements are preserved. Therefore, a safety analysis is performed to evaluate that the requirements are still met after the update of a software component.

The test bench set up is equal to the one used in Section 3.2.3.2.

The case study UC_311_01, “Update of Function (Update)” is implemented in the same way as use case UC_211_01 “Installation of New Component (New)” with the difference that an existing component is exchanged by a different version.

The use-case succeeds, if a software component can be replaced with an update in the system design and the system safety can be automatically checked. By having the ability to automatically check the system in terms of safety, we compare the results of the safety analyses before and after the update of an existing component to check whether the system still meets its predefined safety requirements.

3.3 Dynacar Demonstrator

The Dynacar demonstrator detects in an early stage of the development issues related to driving acceptance, effects on the vehicle dynamics, or energy reduction targets as defined within the SafeAdapt project. For that purpose two different scenarios have been identified, one regarding the energy efficiency increase (but also affecting the vehicle driving dynamics, case UC_511_01, “Adaptation for Range Extension”), and another related to a “Degradation of steer-by-wire application” (UC_411_01), where the main objective will be to determine the maximal failover times before driving the vehicle becomes unsafe. The scenarios will be analysed using the Driver in the Loop (DiL) approach. We define the failover time as the maximum time the vehicle can be without control before the hazard occurs. That will be the maximum time for the adaptation to occur.

3.3.1 Platform Description

Dynacar Real Time is a vehicle dynamics simulation software solution based on:

- Real time testing platform software (NI Veristand® real time framework)
- Graphic visualization system and vehicle control for real test driving in a virtual environment
- Full vehicle dynamics model running on real time equipment (PXI hardware)
- Fully Hardware-in-the Loop and Model-in-the Loop configurable capabilities

Dynacar allows running hardware / software against a vehicle dynamic model in order to see the vehicle dynamics behaviour. As Dynacar software is a full vehicle model, vehicle sensors and vehicle control variables can be used in real time to change the virtual vehicle dynamic behaviour. Figure 8 shows typical Dynacar configurations with the different hardware and software layers.

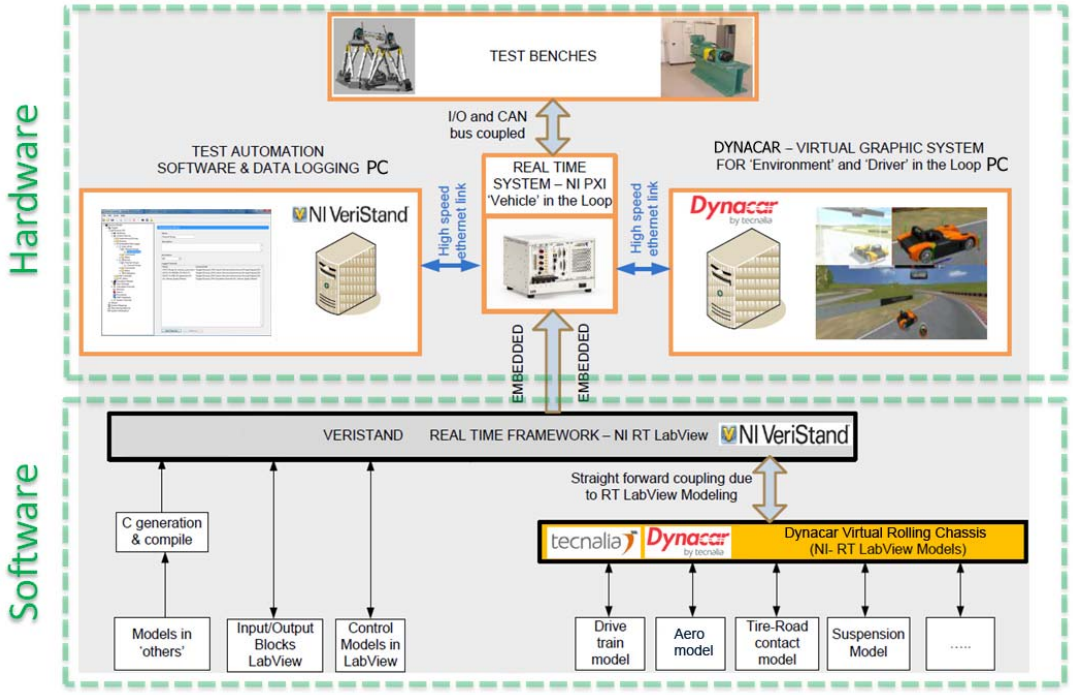

Figure 8: Schematic of Dynacar solution with software and hardware definition

Dynacar is designed to run a real time simulation of an entire vehicle, with Model-in-the-Loop, Hardware-in-the-Loop and Driver-in-the-Loop capabilities. The recommended hardware configuration is shown in Figure 9:

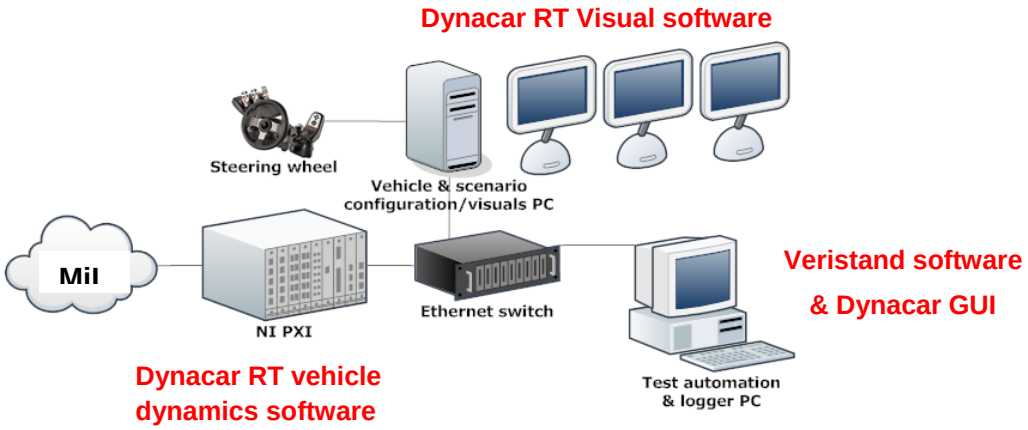

Figure 9: Recommended Hardware configuration for Dynacar RT.

- NI PXI as a real time platform running Veristand real time engine with Dynacar real time virtual vehicle model
- Computer for visual management (3D driving scenarios)
- Computer for Dynacar GUI (vehicle and scenario configuration) and Veristand test automation software

- Ethernet Switch
- Logitech G27 Steering Wheel

The Dynacar platform allows the validation of SafeAdapt use cases with a virtual vehicle model in certain scenario driving tasks. It integrates in a very early stage of development SafeAdapt vehicle system models of the final solution that will be finally tested on the RACE car. A Driver-in-the-Loop (DiL) approach is used, to detect early issues related to acceptability or drivability of the vehicle when the preliminary Safe Adaptation Platform Core (SAPC) system definition is running. Testing work performed with Dynacar platform will support the SafeAdapt preliminary system validation using Model-in-the Loop (MiL) that will help improving the correct definition of the final system capabilities. All the parameters addressed in the testing done with Dynacar software are related to the targets defined in the project, i.e., safety assurance and energy efficiency increase.

This testing approach will allow measuring the impact of the SafeAdapt system faults on the vehicle dynamic behaviour, showing if the impact is safe or not for the driver in terms of vehicle dynamics change. The approach is reflected in Figure 10. Dynacar platform is suitable for early evaluation of SAPC system because it permits the early evaluation of a system when only the preliminary code of working scheme is available. This DiL approach permits the initial detection of system issues in terms of driver acceptance, driveability, or safety. Without such a DiL approach, all of these issues are very difficult to detect during the development.

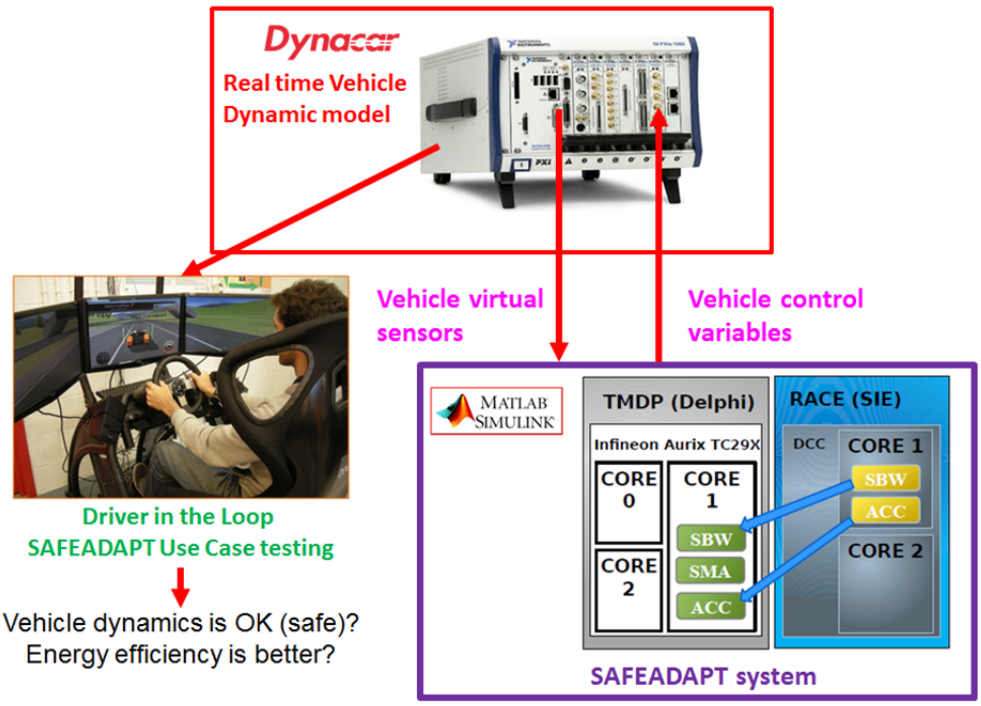

Figure 10: Dynacar platform use with DiL approach for early detection of drivability and safety issues, and energy efficiency checking

3.3.2 Use-Case related to Energy Management

The use case UC_511_01, “Adaptation for Range Extension”, has been selected as representative of the energy management, and also includes vehicle dynamics aspects like torque reduction in degraded mode.

In this use case the vehicle is driving on the road and the Battery Management System (BMS) detects a SOC (State Of Charge) less than 35%. The use of energy has to be prioritised to increase the remaining range, by affecting driving dynamics aspects like the In-Wheel-Motor (IWM) performance or the regenerative braking. The following adaptations are performed:

- The BMS cuts off the power supply for auxiliary services. Here, auxiliary services’ energy consumption will be defined from literature, obtaining the energy reduction percentage when the auxiliary services are disconnected (radio, audio system, navigation, and air conditioning).
- The IWM performance (torque and power) is reduced by 50%.
- The BbW system activates rules to maximise the regenerative braking; during braking events, the utilisation of the generator should be as large as possible to reduce dissipation of kinetic energy in form of heat. The regenerative braking will be maximised taking into account that the vehicle dynamics must remain safe. The maximum regenerative braking will be defined for each test case.

In Figure 11, a schematic of the “Adaptation for Range Extension” use case testing is presented.

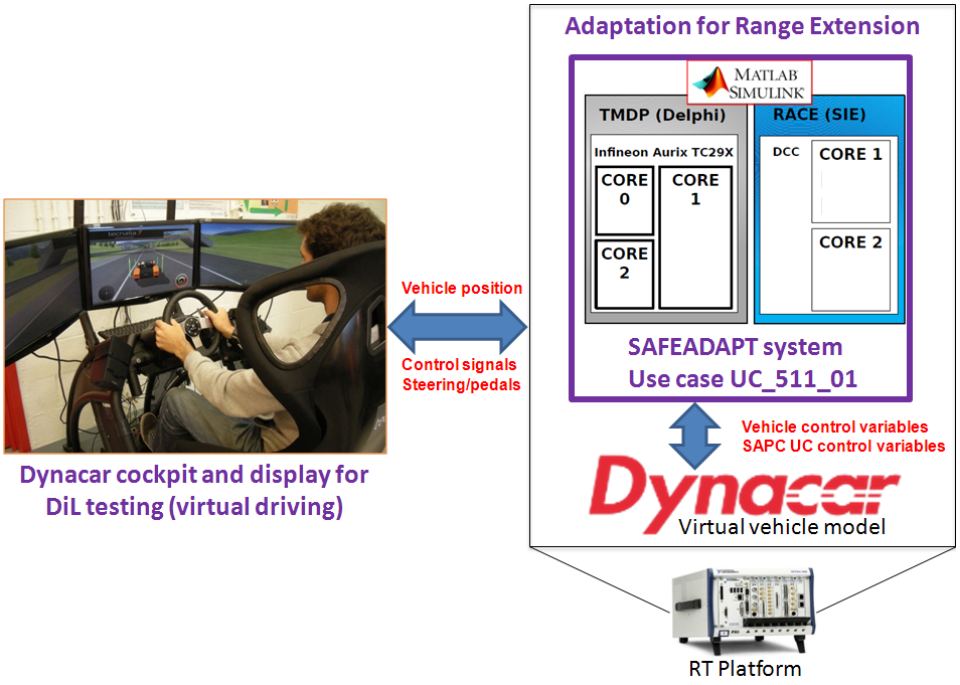

Figure 11: UC_511_01, schematic for “Adaptation for Range Extension” use case validation.

In order to perform the testing using the Dynacar platform, a correct RACE vehicle specification has to be compiled, with all the vehicle data affecting the vehicle dynamics.

The use case UC_511_01 implementation will require the availability of a BMS model in Matlab Simulink (BMS model), that will be provided by Ficosa. This BMS model will be integrated in the

preliminary functional programming of the SAPC system (also in Matlab Simulink), in order to be able to simulate correctly the SAPC system features. A preliminary SAPC model draft to be tested within the Dynacar demonstrator is presented in Figure 12.

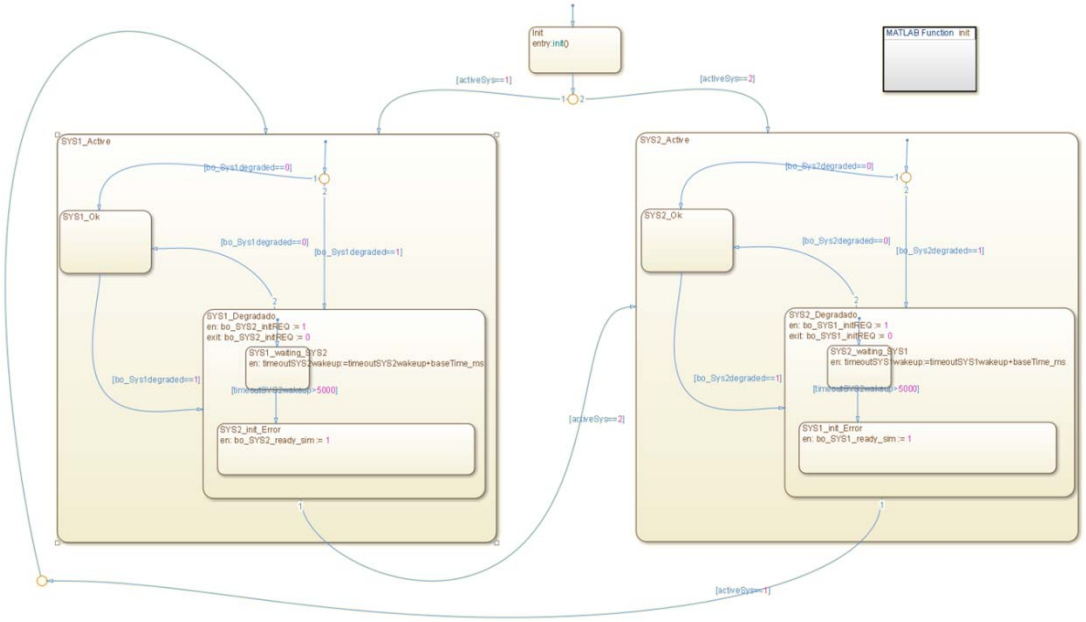

Figure 12: Preliminary SAPC model draft (under development) to be tested within the Dynacar demonstrator

Different test cases will be defined complementing the initial use case UC_511_01 definition described in Deliverable 2.1, in order to be able to identify the energy efficiency increase and the vehicle dynamics unsafe change behaviour, i.e., the drivability and user acceptance. Different parameters will be measured, e.g., system reaction time, distance from optimal path, steering angle angular gradient, and thresholds will be defined to confirm if the vehicle behaviour is acceptable in terms of vehicle dynamics or energy efficiency saving. The results from the test should be categorized so as to get the following possible results:

- Vehicle no longer controllable
- Vehicle reaction dangerous
- Vehicle reaction disturbing
- Vehicle reaction noticeable
- Nothing noticed

The following test cases are proposed in order to complement the use case for different driving situations:

- Slope testing. This test case will show how the degraded torque could affect the vehicle dynamics negatively.

- Overtaking testing. Overtaking is always a critical manoeuvre, and with this test case the SafeAdapt system will be tested when doing overtaking manoeuvres with different conditions (different speed and estimated overtaking time).
- Hard braking testing during a turn. In degraded mode the system will increase the regenerative braking in this use case, and this could lead to stability problems if a hard braking is required when taking a corner. Different brake manoeuvres will be performed during a turn (at different speed and braking force).

Following the previous test cases with a DiL approach, a preliminary validation of the SafeAdapt system will be done showing deficiencies in an early stage of the system development.

3.3.3 Use-Case related to Determining Maximal Failover Times (Vehicle not Under Control)

The use case UC_411_01, “Degradation of Steer-by-Wire Application” has been selected as representative of a critical use case, where a critical system for the vehicle safety is working in degraded mode.

In this use case the vehicle is driving on a road and the Steer-by-Wire (SbW) application experiences a failure while operating in regular mode so that it is not able to function correctly. As a SbW basic application is available to work in a degraded mode, it has to be activated to reach a safe state. For this use case an overall response time should be tested, and the testing objective of this use case within Dynacar demonstrator is to measure the maximum response time of the system, without any safety deviation perceived by the driver.

In Figure 13, a schematic of the “Degradation of Steer-by-Wire Application” use case is presented.

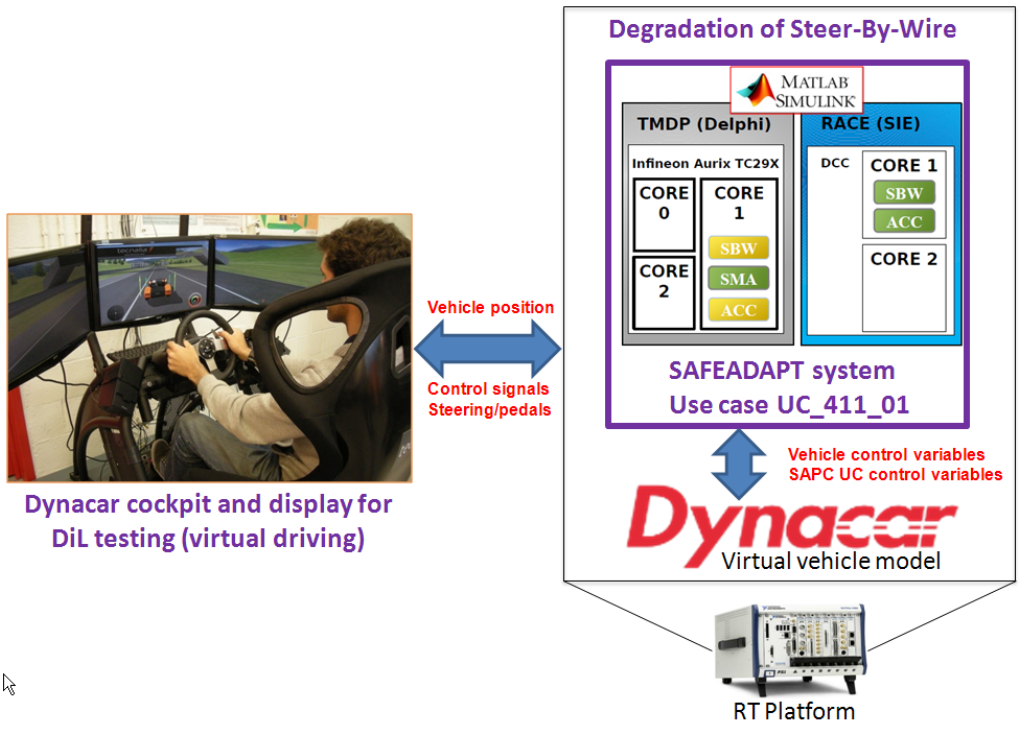

Figure 13: UC_411_01, schematic for “Degradation of Steer-by-Wire Application” use case validation

In order to perform the testing, the Dynacar vehicle model has to be correctly configured with all the vehicle dynamic specifications from the RACE car, e.g., its weight, track, wheelbase, and tyre specification.

The use case UC_411_01 implementation will require the availability of the SbW model. This SbW model will be integrated in the preliminary functional programming of the SAPC system in Matlab Simulink, simulating an adequately running SAPC system. All the system will be tested following a DiL approach with the test cases defined.

Different test cases will be defined complementing the initial use case definition UC_411_01 described in the Deliverable 2.1, in order to be able to identify the maximum time before the vehicle dynamics are unsafely affected, taking into account also the user acceptability or usability and drivability. Different metrics will be identified in order to measure these parameters, e.g., system reaction time, distance from optimal path, and steering angle angular gradient, and thresholds will be defined to confirm if the vehicle behaviour is acceptable in terms of vehicle dynamics. A maximum response time for the SAPC system will be checked with the detailed specification defined, confirming that the SAPC system definition avoids any kind of relevant influence on the car vehicle dynamic aspects, leading to an unsafe driving situation.

The following test cases are proposed in order to complete the use case in different driving situations:

- Low speed driving conditions (city driving)
- Medium speed (interurban driving)
- High speed (highway driving)

For all the test cases, different driving conditions will be checked, i.e., straight, bends, starts, and stops, to reflect normal driving conditions.

Through all these test cases, the unsafe aspects related to vehicle dynamic behaviour of the car, when a real driver is driving (DiL approach), will arise and the correcting actions can be defined.

4 Strategy for Evaluating Efficiency

This chapter describes the evaluation methods used to verify the efficiency of the SafeAdapt approach. Efficiency is "the ability to do something or produce something without wasting materials, time, or energy".[1] We show the efficiency of our SafeAdapt approach by comparing it to other approaches. Therefore, we define metrics for different aspects of the development of a FEV and detail how we obtain values for comparison. They are structured in the following by the measurable results defined in the SafeAdapt project description (cf. Chapter 2).

4.1 Architecture Overview

4.1.1 State-of-the-Art Fail-Operational Architectures

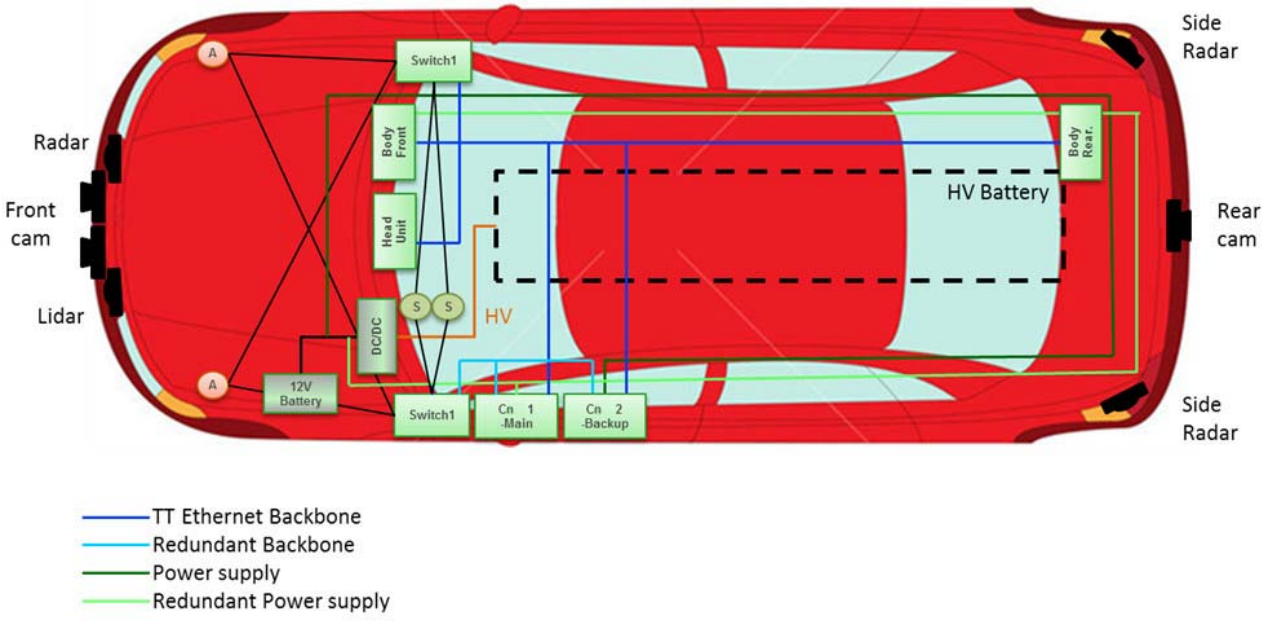

Figure 14: Block diagram of vehicle architecture before SafeAdapt

Figure 14 shows an extract of a state-of-the-art vehicle architecture for a fail operational system. Shown is only the part of architecture that is relevant for the functionality which is considered in SafeAdapt. Although some sensors for autonomous driving are also mentioned in the diagram, the connections between them and the calculating platform are omitted to keep the image clear. Nevertheless, the mechanisms developed by SafeAdapt are also highly relevant for autonomous driving.

This hypothetical state-of-the-art reference architecture will be the base for the evaluation, due to the reduced safety features currently implemented in real state-of-the-art FEV designs. In this architecture, the redundancy is implemented by introducing a "spare" ECU that is capable of taking over the responsibility for a safety critical system such as steer-by-wire or brake-by-wire. Both core nodes (Cn) have to be ASIL D compatible in this case. The second ECU executes exactly the

[1] http://www.merriam-webster.com/dictionary/efficiency

same operations as the main one, but with its outputs disabled. Only in case of a failure in the main core node, the second core node will activate its outputs to control the actuators.

In case of the demonstrator this system is in charge of functions like the above mentioned Steer- or Brake-by-Wire. As soon as such functionality will be extended with some autonomous driving features, the needed calculation power will increase significantly because of the large amount of data, e.g., due to image handling.

To be prepared for this functionality, a core node can be estimated to consist of one performance controller, e.g., on ARM base and one safety controller, e.g., the Infineon Aurix. Alternatively, it is possible to think of a system with two performance controllers running in loosely coupled lockstep to fulfil besides the calculation power also the safety requirements. With this background, the properties of the state-of-the-art system, that are relevant for the evaluation, can be assumed like shown in the following table:

State-of-the Art / Before SafeAdapt					
	Cn 1	Cn 2	Body computer Front	Body computer Rear	Wiring
Cost	100%	100%	70%	60%	150%
Weight incl. Housing, brackets, connectors	500g	500g	400g	350g	700g
Size	210cm^2	210cm^2	350cm^2	300cm^2	n.a.
Power consumption	25W	25W	20W	15W	n.a.
Ethernet Cost	t.b.d.	t.b.d.	t.b.d.	t.b.d.	t.b.d.

Table 2: Properties of a state-of-the-art system

All values are based on experience with comparable ECUs. The costs are given as relative values because the exact values are confidential numbers of the partners. The meaning of these numbers is, if the cost of one core node is rated as 100%, then the Body computer price is around 70% of the cost of the core node cost.

4.1.2 System with a SafeAdapt Architecture

Figure 15 shows the block diagram of vehicle architecture with respect to the results of SafeAdapt. The main difference is that the “spare” ECU is eliminated and the Body Front computer is now the instance to take over this responsibility. With the very same mechanism of SafeAdapt it is of course possible to easily extend such a system with an additional ECU. In this example it could be the head unit of the entertainment system. As more calculation power is available in the system, more adaptation scenarios are possible, e.g., it would be possible to establish the hot redundancy after an adaptation again.

For the comparison of the architectures it is sufficient to stay with a simple system consisting of only two core nodes. An additional ECU will mainly increase the effort for configuration of such a system and the software development, but not change the metric-relevant facts.

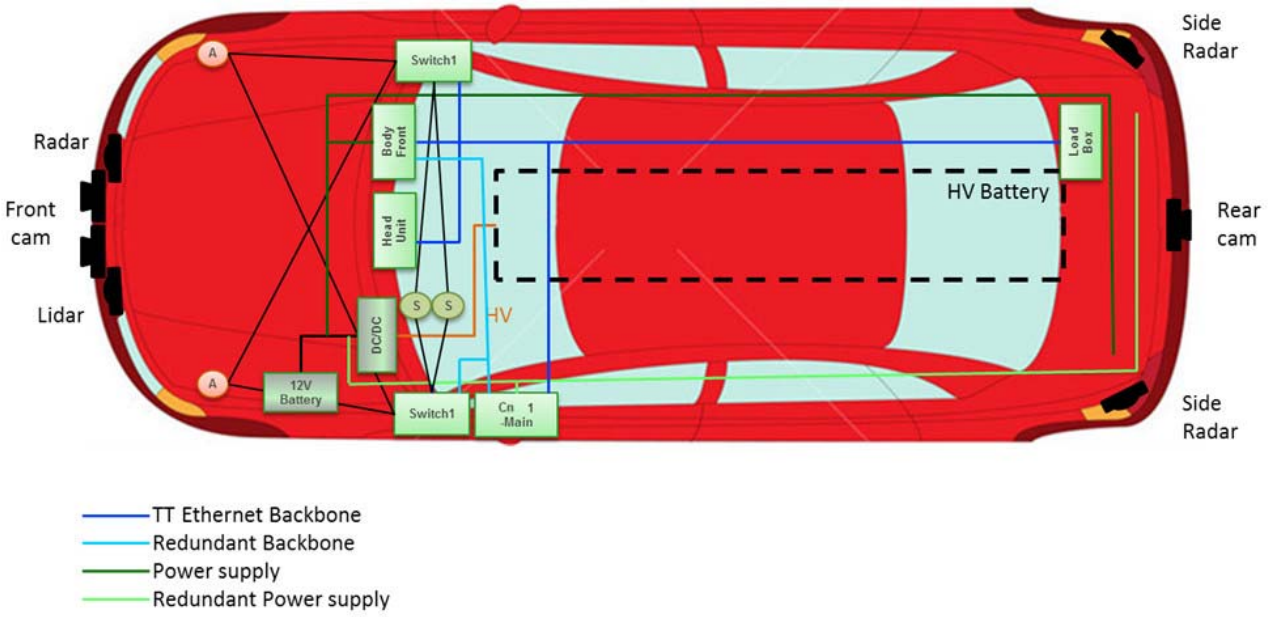

Figure 15: Block diagram of vehicle architecture after SafeAdapt

As for the state-of-the-art system the criteria for the evaluation are summarized in the following table:

After SafeAdapt					
	Cn 1	Cn 2	Body computer Front	Body computer Rear	Wiring
Cost	100%	n.a.	t.b.d.	t.b.d.	t.b.d.
Weight incl. housing, brackets, connectors	500g	n.a.	t.b.d.	t.b.d.	t.b.d.
Size	210cm^2	n.a.	t.b.d.	t.b.d.	t.b.d.
Power consumption	25W	n.a.	t.b.d.	t.b.d.	n.a.
TT Ethernet cost	t.b.d.	n.a.	t.b.d.	t.b.d.	t.b.d.

Table 3: Properties of a system after SafeAdapt

Of course, without completing the demonstrator, there are facts missing (marked by “t.b.d.”). Therefore, this table is incomplete, yet, but will be completed for the report on the evaluation with Deliverable 5.3.

4.2 MR1: Optimised Energy Consumption

The overall energy consumption of the state-of-the-art system is estimated to 85W as shown in Section 4.1.1. Target for SafeAdapt is to optimise the energy consumption of safety-relevant features by up to 30%. In this case, this equals a system power consumption of around 60W.

This target is not only depending on the eliminated core node in the system, but also on the increase of power consumption due to the deviation in calculation power needed in the Body computers.

A realistic estimation can be given at the end of the project, when the overhead due to SafeAdapt mechanisms is known.

4.3 MR2: Failures Handled by Adaptation

The adaptation mechanisms provided by SafeAdapt allow moving functionalities in a vehicle to a different ECU in case of ECU internal failures. This mechanism can be exploited for several purposes. First of all, for functions with high criticality – such as Steer-by-Wire – which needs a "fail-operational" behaviour (i.e., a redundant deployment), adaptation can be used to achieve the necessary ASIL-level, e.g., by switching over to another ECU in case of a probabilistic failure. The efficiency here is achieved by the avoidance of duplicate specific function-hardware, as it would be the case with existing architectures. This aspect is addressed in Section 4.2 "Improved redundancy concept".

Secondly, adaptation can be used also for functions with less criticality, i.e., for functions where fail-safe behaviour is sufficient. This is the case for most of todays' automotive functions. By this application, adaptation is used to enhance the "mission time" of a car, i.e., the time a car can safely drive without repairing some parts. When functions can be moved by adaptation to another ECU, the car can operate even with one or several faulty ECUs. Since about 55% of the defects in nowadays cars are related to electronics, software, cables and connectors [McKinsey/VDA, "HAWK 2015: Herausforderung Automobile Wertschöpfungskette",2003] this can largely improve customer satisfaction.

The goal of this metric therefore is to measure the improvement in terms of user-observable function failures. It can be evaluated by comparing a traditional E/E architecture with an E/E architecture as proposed by SafeAdapt. In a conventional architecture, we have to analyse for each function:

- The failures which lead to a user-perceived loss of that function
- The failure rates of these failures

In the SafeAdapt architecture, we have to analyse for each function:

- The failures which lead to a user-perceived loss of that function and which cannot be recovered by SafeAdapt
- The failure rates of these failures

The goal is to achieve a 20-30% increase in the number of failures which can be handled without a user-observable loss of functionality.

4.4 MR3: Cost Reduction

Concerning cost reduction a target of 20% is envisaged. The idea of using available resources w.r.t. at the moment unused ECU computing power on board of the vehicle goes back to the aerospace domain. It was first mentioned at Airbus in order to reduce redundant hardware in the aircraft. It is assumed that redundant systems will also find their way into the next generations of cars, when autonomy in the vehicles is raised step by step. Today, systems available on the market are understood as "assistance" systems, which are not built to guarantee their function under any circumstances, e.g., a safe emergency stop in case of an obstacle detected. They

assume that the driver is the highest instance and will also take full control in such situations. Next generations of systems also target partial autonomous and fully autonomous operation, meaning that also the responsibility for a safe fulfilment of safety-relevant operation will be covered by the function on its own. Thus, such a system will also have to care for the case that a component fails and will still have to provide the function.

When reconfiguration can make use of sufficiently powerful computing systems, not needed or irrelevant for safety at the moment, redundant systems will not be needed and the extra hardware does not need to be built in. Reconfiguration by software application will manage such reconfiguration selecting appropriate hardware and connecting all network components appropriately to cover the functionality in real time. This provides significant saving potential since additional hardware, else needed to cover the functionality, does not have to be built in.

This metric will take into account the cost of the hardware that does not have to be built in, in case such type of reconfiguration is used instead. The following aspects will be incorporated in the metric:

1) It saves complex, expensive hardware modules to be installed on the vehicle
2) It thus saves development cost of the now less complex system
3) It improves the availability of the system and reduced maintenance costs
4) It saves power and energy costs since less ECU modules will be installed or allows less expensive batteries to be used while preserving the same range
5) It reduces the complexity of the network in the vehicle, also reducing development costs
6) It reduces the development cost for supporting autonomous systems for safety relevant applications, by providing a generic failure management

The figures depend on the ECU count reduced and on the price of the ECUs that do not have to be installed. Each saved ECU simply adds up to the saving in cost. The saving follows directly from these figures.

4.5 MR4: Reduced Certification Cost

One of the main challenges we faced when trying to verify the SafeAdapt certification is that the architecture proposed in the project will be a prototype and not a market product. Thus, it is out of the scope of the project to certify the solution.

Measures related to verify this objective will be based on estimations from partners with a long term experience certifying their own products.

The certification cost reduction in based on the following perspectives:

- Reusable system architecture
- Tool qualification effort
- Safety goal verification effort
- Functional safety management effort

The goal of this metric is to analyse the certification cost reduction by applying the new software architecture defined on the SafeAdapt project. This software will be responsible for handling arbitrary failures by using adaptation mechanisms.

Using the modular approach introduce by applying the SAPC architecture, complexity of the developments decreases. This will end up reducing the effort needed on validation and verification of software applications that will run on top of the SAPC architecture.

SafeAdapt proposes the use of a default error handling mechanism to provide safe-operational functionality. As this mechanism is covered by the SAPC, the applications can leverage the safe-operational functionally with no special development. This way, the cost for certifying them is controlled.

The SafeAdapt tool chain methodology improves effectiveness in phases related to the concept, design and early verification and validation w.r.t. ISO 26262. Tools, e.g., such as Prossurance, in the tool chain support documentation management for certification related activities. This approach facilitates the re-use of evidence for safety case generation and argumentation or safety assurance in different projects.

Direct comparison in automotive industrial domain will be difficult since appropriate FEV based examples are missing. We will need to compare the situation before applying the use of the SAPC for proving the fail-operational functionality, and the tools and methods proposed on the SafeAdapt project after applying the proposed approach.

Data will be based on qualitative estimations and possibly may be derived from aerospace domain by study efforts. Interviews with partners with experience in certifying their own products are essential for data collection. We will apply the following guidance questions to recover the data:

1. How many components do we need to certify?
2. Do the developed methodology and tool chain support the product development?
3. How do the design methods and tools improve certification?
4. Do the developed methods and tools support early verification and validation?
5. Are the adaptation safety goals verified?
6. Are the evidences generated for the SAPC for complying with the ISO 26262 reusable?
7. Do the tools and methodology support the recompilation of complying evidences?

Based on the guidance covered before, information will be collected based on the actual state-of the-practice in the industry. After collecting data for current practices, we will make estimations based on results from the demonstrators, to which the SAPC architecture and the methods and tools will be applied. We are expecting to reduce the certification cost by up to 20%.

4.6 MR5: Reduced Complexity

SafeAdapt provides a generic mechanism for the safety-handling of the system. In contrast, in state-of-the-art systems, provisions for assuring the safety are implemented through additional components. This increases the costs for the additional installed parts as well as the complexity, i.e., exponentially increasing effort required to manage the system. For evaluating the advantages

of the SafeAdapt approach in comparison to state-of-the-art equipped systems with respect to the reduction of complexity, the difference in required effort, components, and cost are relevant.

The aimed goal of SafeAdapt is to reduce the complexity and hardware cost of safety-relevant systems by 20%. As the number of required hardware components through the improved redundancy concept is less (cf. Section 4.7), also the resulting complexity is expected to be reduced. Moreover, the generic safety mechanism of the SAPC can be provided per ECU and must not be intertwined with every application which requires additional safety measures. Thus, these mechanisms only have to be implemented and validated once and can be deployed to various ECUs. The reduced complexity can also directly be derived by the expected lower cost (cf. Section 4.4).

An evaluation of the SafeAdapt approach with respect to the reduced complexity can thus be carried out by comparing the SafeAdapt architecture with present state-of-the-art event-driven system architectures ensuring same safety, with respect to required hardware components as well as required development effort. The first can be calculated by the sole number of components. Required saved effort can be determined relatively by anticipating the needed lines of code for the safety functionality provided by the SAPC for every critical function individually. A comparison of the implementations through a generic mechanism with SafeAdapt and through state-of-the-art realisation per application will indicate the differences with respect to development effort.

We expect results underpinning the estimations of reductions of complexity of up to 20% through the SafeAdapt approach.

4.7 MR6: Improved Redundancy Concept

One of the major advantages of SafeAdapt is the extendibility of the system. Including an additional ECU requires, beside some restriction in the wiring harness, e.g., for redundant power supply, mainly effort for configuration and software coding. But with every additional ECU in the system the redundancy concept potentially improves. Although, only the first failure is in the scope of this project, more advanced adaptations are enabled: on an additional ECU, it becomes possible to re-establish the hot-standby of a function that failed on a different ECU. Alternatively, the number of functions, that take part in the adaptation, can be increased. This would lead to a kind of hierarchy of the functions, where only the ones with the lowest ranking will be eliminated.

One objective of the project is to improve current redundancy concepts in terms of duplication of ECUs by up to 50%, i.e., fulfil the redundancy concept requirements with less extra ECUs. The viability of the approach will be shown by the evaluation described in Chapter 3. Therefore, this metric focuses on the amount of computational power required. In the architecture example described in Section 4.1.2, we save one core node compared to the state-of-the-art reference architecture. However, we require additional calculation power in the body controller. For this metric, we will measure the computational overhead introduced by the SafeAdapt mechanisms to determine how much calculation power is needed to provide redundancy by adaptation. Using this measure, we can make a more general estimate on the number of ECUs required and compare this to the number of ECUs required when using duplication.

5 Summary

In this document, the evaluation methodologies and metrics to assess the properties of the project's results are presented. All partners contribute their own tools, platforms, and applications to evaluate the project results. The evaluation is two-fold.

The first part verifies the viability of the approach with several use cases, which were already defined in Deliverable 2.1. This viability evaluation of the SafeAdapt project results will be based on the full scale prototyping e-vehicle, radio controlled conceptual demonstrator vehicles, and a vehicle dynamics simulation software solution. This will lead to realistic statements about the advantages and drawbacks of using the Safe Adaptation Platform Core in future FEVs.

The second part measures the non-functional aspects of the approach with several metrics. Metrics are set up for the evaluation of non-functional aspects like reliability, availability, efficiency, and flexibility. Within these metrics, the results of the WP3 and WP4 will be compared to current state-of-the-art systems in the automotive domain.

The evaluation described in this document will be executed in the further progress of WP5. The results of the evaluation will be presented in Deliverable 5.3.

List of Abbreviations

Abbreviation	Definition
ACC	Adaptive Cruise Control
AEB	Automatic Emergency Brake
APP	Application
ASIL	Automotive Safety Integrity Level
BbW	Brake-by-Wire
BMS	Battery Management System
CCC	Central Computing Core
CDD	Complex Device Driver
CFT	Component Fault Tree
Cn	Core Node
DCC	Duplex Control Computer
DiL	Driver-in-the-Loop
EBC	Emergency Brake Control
E/E	Electric / Electronic
FEV	Fully Electric Vehicles
GW	Gateway
HiL	Hardware-in-the-Loop
HW	Hardware
I/O	Input / Output
IWM	In-Wheel-Motor
MiL	Model-in-the-Loop
RACE	Robust and reliable Automotive Computing Environment
RTE	Runtime Environment
SAPC	Safe Adaptation Platform Core
SbW	Steer-by-Wire
SOC	State Of Charge
SW	Software
TMDP	Trusted Multi Domain Platform
TT	Time-Triggered
TTE	Time-Triggered Ethernet