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 D2.2

Requirements for the Run-time Control for Safe Adaptation and Supporting Hardware Platforms

Due date of deliverable:	30.09.2014
Actual submission Date:	22.12.2014
Lead beneficiary for this deliverable:	TTTech Computertechnik AG

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	Requirements for the Run-time Control for Safe Adaptation and Supporting Hardware Platforms
Creator	Andreas Eckel
Description	A detailed collection of requirements from a platform perspective and a description of the governing requirements engineering process
Publisher	TTTech Computertechnik AG
Contributors	CEA: Ansgar Rademacher DEL: Timo Feismann Duracar: Ken Lam Ficosa: Daniel Bande, Andrea Saccagno Fraunhofer: Alexander Stante, Christian Drabek, Gereon Weiss Pinifarina: Elena Cischino and Sandro Morero Siemens: Cornel Klein, Michael Armbruster Tecnalia: Alejandra Ruiz, Maite Alvares Piernavieja
Language	en-GB
Creation date	2014-09-01
Version number	1.0
Version date	2014-12-19
Audience	☐ internal ☒ public ☐ restricted

Table of Contents

List of Figures

Executive Summary

This document summarises the SafeAdapt requirements. It also provides a brief summary on the reconfiguration's structural set-up, which will be integrated into the RACE platform. Such reconfiguration mechanisms and algorithms will form a central part of the SafeAdapt Platform Core, which encapsulates the basic adaptation mechanism for re-allocating and updating functionalities in the networked, automotive control systems. This will be the basis for an interoperable and standardised solution for adaptation and fault handling in AUTOSAR. The SafeAdapt approach also considers functional safety with respect to the ISO 26262.

The requirements address the necessary developments targeted in SafeAdapt in order to achieve the SafeAdapt Platform Core goals.

1 About this document

This document contains the requirements for the SafeAdapt project with respect to the run-time control for the safe adaptation and supporting hardware platforms.

It has been decided to select a requirements capture process allowing to use the IBM Rational DOORS (short within the following: DOORS) software for processing requirements supported by a professional tool. Since not all project partners have access to this tool, it was agreed to use a special Microsoft EXCEL template which can directly be read by the DOORS software as an input file.

DOORS[1] is a requirements management application for optimizing requirements communication, collaboration and verification.

The DOORS software for requirements processing supports:

1. Requirements Management in a centralised location for better team collaboration
2. Traceability by linking requirements to design items test plans and test cases and other requirements
3. Scalability to address the changing requirements management needs
4. Test tracking toolkit for manual test environments to link requirements to test cases
5. Integrations to help manage changes to requirements with either a predefined change proposal system or a more thorough customizable change control workflow

This document consists of two parts:

a) This word file with general comments and explanations on the process followed and the design goals targeted

b) The DOORS compliant EXCEL requirements sheet with the collected requirements for this part of the SafeAdapt Project

Concerning the EXCEL requirements sheet we followed the following approach:

The requirements were collected per Partner. This can be traced by the Requirement ID provided by each individual requirement. The numbering system used the following syntax:

Company short (i.e. TTTech: “TTT”) – 3 digit number XXX: <Company short name-XXX>

thus resulting in an identifier for a requirement for example like: “*TTT-001*” (first requirement by TTTech).

[1] See https://www.google.at/#q=DOORS+Requirements

The Excel sheet then identifies the following data per requirement:

Column A: Requirement Identifier

Column B: Category (functional/non-functional, could be extended if needed)

Column C: Sub Category (Efficiency/Hardware/Process/Software/System/Tools)

Column D: Short Description

Column E: Description

Column F: Verification Method

Column G: Rationale

Column H: Dependencies

Column I: Conflicts

Column J: Date (of issue)

Column K: Supporting material

Column L: Object Status (changed/new/, could be extended if needed)

Column M: Object Version

Column N: Review

2 Major design goals these requirements refer to

2.1 The demonstration set-up

It has been agreed to connect the Trusted Multi Domain Platform (TMDP) by Delphi with the Robust and reliable Automotive Computing Environment (RACE) Platform by Siemens by means of a TTEthernet[2] switched backbone network. The set-up is intended to be used to demonstrate the development of reconfiguration capabilities for a system running safety-relevant applications or functions. In case of a safety-relevant function failing is detected, the related SafeAdapt mechanisms target to recreate the function at least in a gracefully degraded version on different ECU of the system. Namely, this intends to first recognize the failure of the safety-relevant function and to autonomously initiate the reconfiguration process. As a next step the system will identify the appropriate “gracefully degraded” version of the function and will define another ECU in the system capable of hosting the degraded function. Either it is already allocated to this ECU or it will load the gracefully degraded function software to this ECU from a repository (i.e. stored on the TTE SCB) and autonomously initialize and start the function. By connecting actuators and sensors via the backbone to the new ECU by means of a switch, the newly configured system is capable of acting in place of the initial system configuration, running the full blown version of the failed function, even during run-time of the system. The system will also take care of disconnecting and switching off the failed function.

A block diagram of the intended system is depicted in Figure 1.

[2] SAE Standard SAE AS6802

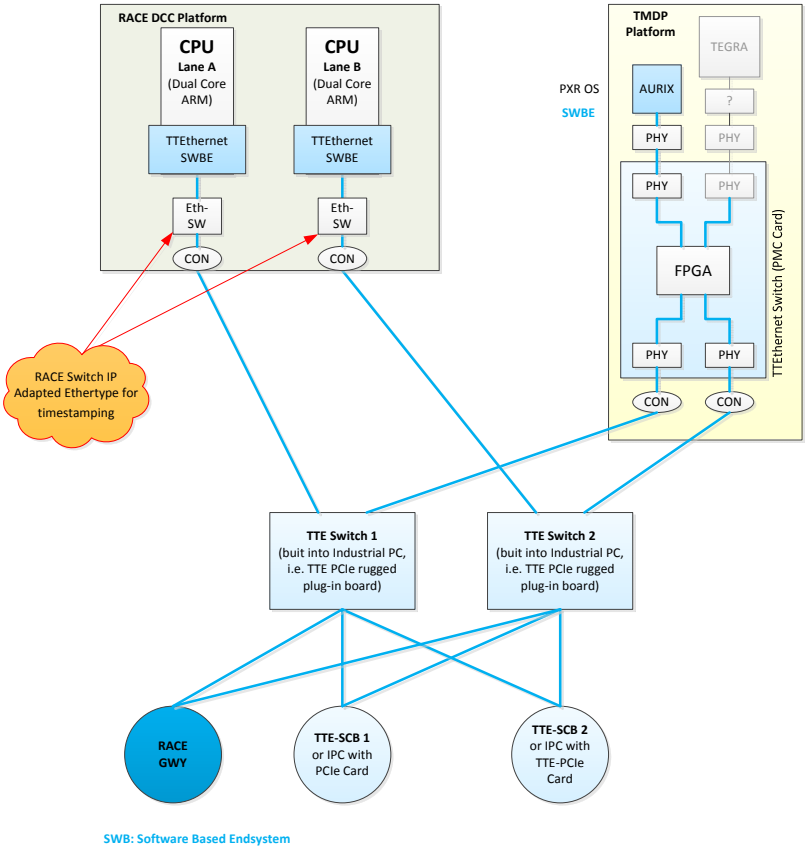

Figure 1: Block Diagram of connecting the TMDP Platform with the RACE Platform

2.2 Short description about involved platforms

2.2.1 TMDP

Delphi's **T**rusted **M**ulti **D**omain **P**latform (TMDP) is a prototyping platform to support different types of safety critical applications up to the ASIL level D (for ASIL levels, see Section 2.3.4). The general setup is shown in Figure 2.

The platform has a modular architecture consisting of main parts which are always on the platform PCB and some pluggable add-On boards.

All automotive relevant networks are supported (CAN, Flexray and LIN) and additionally a very flexible solution for the Ethernet connection. The main processing unit is the Infineon Aurix. The Aurix is a triple core processor with a maximum clocking of 300MHz each. One of the cores is running in HW lockstep mode the other two are standard.

The mainboard, including the power-supply concept, is designed to be ASIL D compliant.

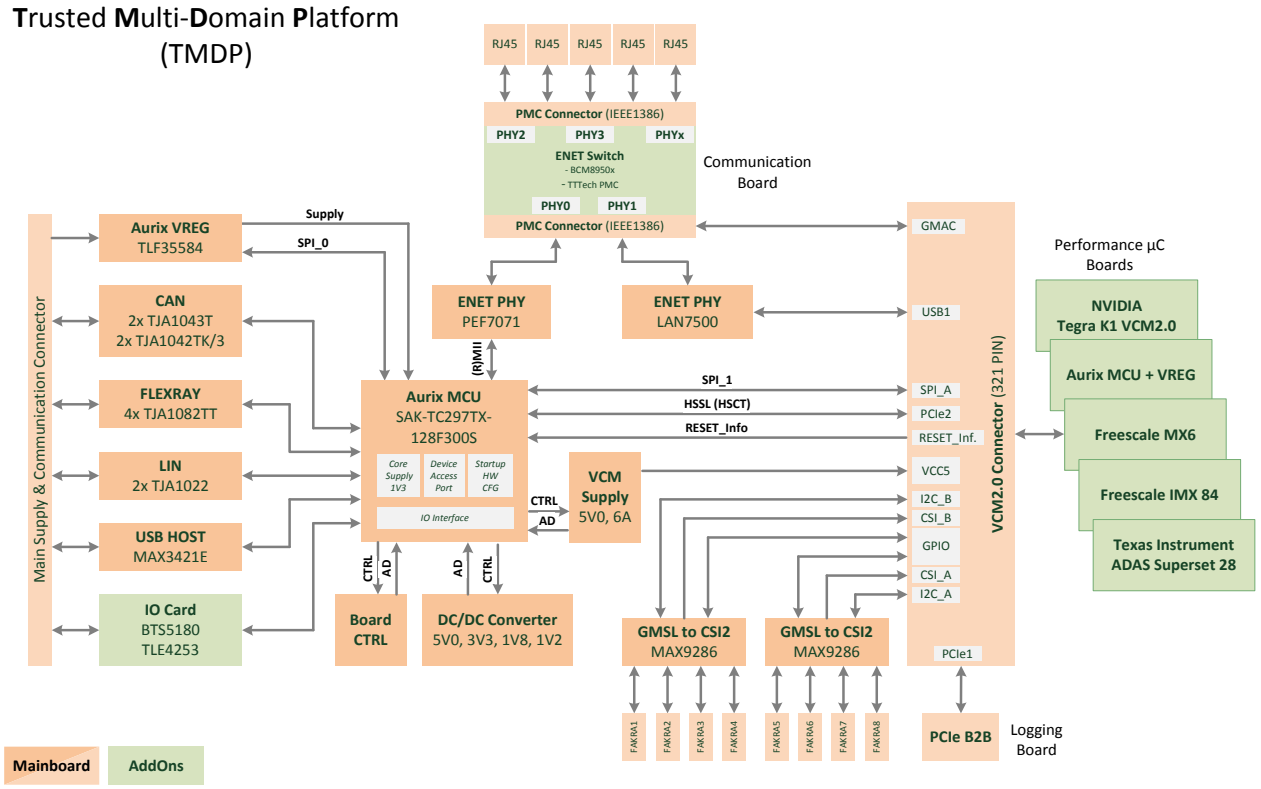

Figure 2: Hardware setup of the TMDP

The platform can be enhanced by three different types of add-on boards. One essential board, that is also mandatory for SafeAdapt, is the communication board. This board is connected to the main PCB via standardized PMC connectors. This gives us the possibility to directly plug a TTTech rugged Ethernet switch onto the mainboard. Different solutions are also possible as long as they are compliant to the PMC standard (IEEE 1386).

To increase the processing power it is possible to plug in an additional performance micro-controller board that is following the VCM2.0 specification of NVIDIA. The connector itself is a MXM 3.0 connector. The performance microcontroller can communicate to the Aurix via HSSL (ZipWire) or an Ethernet connection via the communication board.

In case those digital and/or analogue IOs are needed, the user has the option to plug an IO-board to the mainboard. On the IO-board you can place all the necessary parts to drive the peripherals.

2.2.2 RACE

The RACE platform has been developed within the German national research project “RACE – Robust and reliable Automotive Computing Environment for future eCars”, funded by the German ministry of economics and technology. The main goal of the project was to develop a uniform and open E/E platform for electrical cars, in particular for safety-critical functions up to ASIL D (like steer by wire). The approach is intended to be “revolutionary” in the sense that it completely neglects the historically grown approach taken by the automotive industry and fully “re-thinks” the way to build up the electric- and electronic (E/E) vehicle architecture.

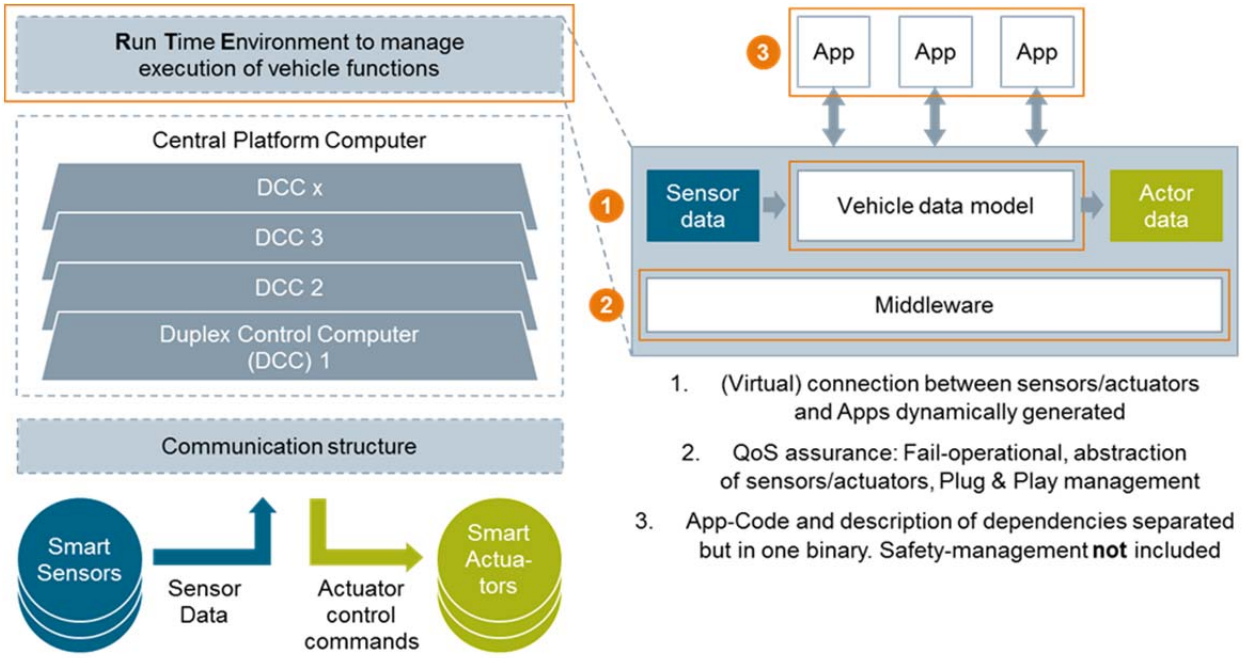

Figure 3: The RACE Platform

Figure 3 provides an overview about the RACE system architecture. The main idea is to implement a clear layering with well-defined interfaces in order to decouple sensors/actuators, the computation and communication platform and the SW functions from each other. This is the basis for “Plug&Play”, i.e. the capability to add new components (sensors, actors, computing nodes) but also new SW based functions in a flexible and modular way. The main architectural concepts of RACE are:

- *Smart sensors and actors*, which provide local intelligence to execute open-loop and closed-loop control tasks. Examples are a wheel hub motor detecting the maximum torque by itself or a video camera generating an object list. Legacy sensors and actors (e.g. with a CAN interface) can be connected to the RACE system by *gateways.*
- A suitable *communication structure*, which is physically based on 2-wire automotive Ethernet. In order to facilitate the implementation of highly safety-critical functions additional mechanisms like “Time Sensitive Networking” (TSN) and a redundant ring structure are employed, such that no single point failure will lead to loss of function. The goal is to support both, time-critical and reliable communication (e.g. for a steer-by-wire function) as well as non-critical best-effort traffic (e.g. for multimedia applications) on one network.
- A *centralized computing platform*, which is composed of one or more *DCCs (Duplex Control Computers)*. Each DCC consists of two CPUs, executing the same computations. By periodically comparing the results of these computations, sporadic failures can be detected. In such a case the DCC will be deactivated (i.e. stopped or restarted), while another DCC can take over the functions of the deactivated DCC. By deploying the same function on multiple DCC in a hot-standby manner, the required availability/safety level can be achieved.

- A *runtime environment (RTE)* which provides a virtual connection to all sensors and actors by means of a vehicle data model. The RTE provides basic mechanisms for failure detection and handling as well as for dynamic system configuration ("Plug&Play").
- *Applications ("Apps")* which are implemented on top of an API for accessing the vehicle data model. These apps do not have to take care of all aspects handled by the RTE, such as failure handling, communication or physical sensors. That way, automotive applications can be implemented pure software packages without the need to take care of car specific aspects like physical sensors, networking structure etc.

The RACE vehicle has been developed within RACE as a demonstrator car. It features innovative components like a wheel-hub motor, an electrical braking system and a full steer-by-wire system. All components and functions of the car are implemented with the RACE architecture. In particular, it is intended to demonstrate the safety features of the RACE platform. Additional demonstrator applications, such as autonomous parking and energy management, have been also implemented within this car in order to highlight the potential of the proposed approach.

2.2.3 TTEthernet backbone platform

TTEthernet (Deterministic Ethernet) is an SAE standardized[3] data communication technology supporting to transmit data according to standard Ethernet best effort traffic (BE), rate constraint (RC) data traffic and time-triggered Ethernet (TT) traffic in parallel (see Figure 4).

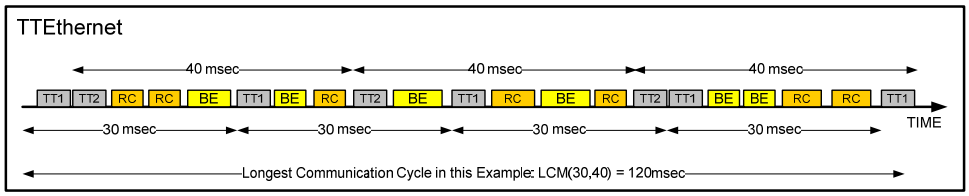

Figure 4: TTEthernet Traffic Classes supported

Time-Triggered Ethernet traffic dispatches messages according to a predefined communication schedule.

Rate-Constraint Ethernet traffic enforces minimum duration between two frames of the same stream.

Best-Effort Ethernet traffic is equal to standard Ethernet traffic and does not provide any temporal guarantees.

TTEthernet is a scalable, open real-time Ethernet platform targeted for the use within safety-related applications primarily in transportation industries and industrial automation. TTEthernet provides flexibility, modularity, scalability in Ethernet based systems. It is compatible to IEEE 802.3 Ethernet and integrates transparently with Ethernet network components.

TTEthernet has been designed for use in high safety and high reliability applications, cyber-physical systems and unified networks. TTEthernet simplifies the design of fault-tolerant and high availability solutions.

[3] SAE Standard SAE AS6802

Detailed information is available on the TTTech homepage. The specification can be requested on the TTTech home page as well[4].

2.3 Major requirement areas

2.3.1 Reconfiguration

The approach selected in SafeAdapt w.r.t. reconfiguration deviates significantly from the investigations and developments made in the FP7 project DREAMS[5]. Actually the SafeAdapt approach complements the investigations of DREAMS.

In DREAMS the concept follows a similar approach like in aerospace used today. It uses a function already resident in another ECU. This function is already operational and activated by switching to this function in case of a faulty ECU or the operational function is failing. In such case a switch over is performed. The failure is detected automatically and triggers action to perform a switch based reconfiguration using redundant hardware and preconfigured additional (back-up) functions. The approach does not work in case the other ECU is suffering from failure as well. In the SafeAdapt approach such failure can be handled even without extra hardware redundancy required.

In SafeAdapt we target to detect a failure of a safety-relevant function in the vehicle system automatically and upon such detection decide on initiating a reconfiguration action.

The reconfiguration can be split up to reactivation and reallocation. Reallocation describes the process of installing SW on an ECU. Reactivation describes the process of assigning computing and timing resources to an installed application. This means, an activated application will be executed.

In order to provide safe hardware redundancy, the system checks automatically which of the other control units executes the least required applications that are signed to tolerate a passivation. It then endeavours to clear the running functions from the new target control system and loads the gracefully degraded version of the failed function to the selected control unit[6]. Afterwards it connects the sensors and actuators to that unit via the available switches in the network and initializes and starts the unit to run the gracefully degraded version of the failed function. It also takes care of the failed function and closes it. Thus the only extra service required is suitable space on a dedicated ECU to store the degraded versions of target safety relevant functions. This approach supports to decide on a case by case basis which other control unit in the overall system would be capable to activate the gracefully degraded version of the failed safety-related function. It does not require extra redundant systems to be able to cover the full safety related requirements including tolerating the failure of a safety related function at minimum gracefully degraded function level. Instead it will be required that an ICT system with its applications can be reconfigured. Therefore, less important automotive functions/applications will be stopped, resources will be freed and (degraded) functionality reactivated. The overall process of reconfiguration shall be completed in so short time, that the operation is not influenced significantly.

[4] https://www.tttech.com/technologies/ttethernet/.
[5] http://www.dreams-project.eu/
[6] In case the function is significantly small in related code it might also be considered to load the full function. In a second step it might also be possible to replace a gracefully degraded version after the gracefully degraded version has taken the role to provide at least a minimum required service at the shortest interruption possible (i.e. a few milliseconds).

The project intends to find out how long it would take from failing of a safety related function to a completed reconfigured function being restored and being fully operational with the new configuration by using the set-up described. This time span is called fault-recovery time which must be less than the given fault-tolerant time interval (according to IOS26262) of the considered application.

It shall also be taken into consideration how much the maximum size of a gracefully degraded version of such function may be, in order to meet deadlines overall of a few milliseconds only.

This shall lead to the insight if 100Mbit/s data rate can be sufficient or if the next order of magnitude is required. It shall provide knowledge on the influence of different parameters in the duration of the entire reconfiguration process. One of the results expected would be to define the parameters really decisive for the fault-recovery time (i.e. maximum amount of code to be transferred to a new ECU as “gracefully degraded version”, data rate, amount of ECUs in the pool to use instead of the failing ECU, etc.).

2.3.2 System Verification and Evaluation

The main idea of SafeAdapt is to develop novel architecture concepts based on adaptation to address the needs of a new E/E architecture for FEVs regarding safety, reliability and cost efficiency. In order to judge a research project whether it is successful, the following elements as the cost, energy efficiency, reducing material (the weight), State of the Art should be taken into the evaluation process and if possible some of the final results can be demonstrated to the public.

In order to evaluate a system or a function or a use case effectively some specific requirements with review to the verification and evaluation should be made at the early stage of the design process and added into the run time core requirement list.

2.3.3 Universal SW component

General requirements towards the safe adaptation platform core (SAPC) are aiming at a universal SW component that shall run on all platforms that are taking part in the reconfiguration process. Therefore the SAPC shall behave like a standard AUTOSAR SW component and shall only support the standard RTE interfaces.

A simple form of the SW architecture is shown in Figure 5.

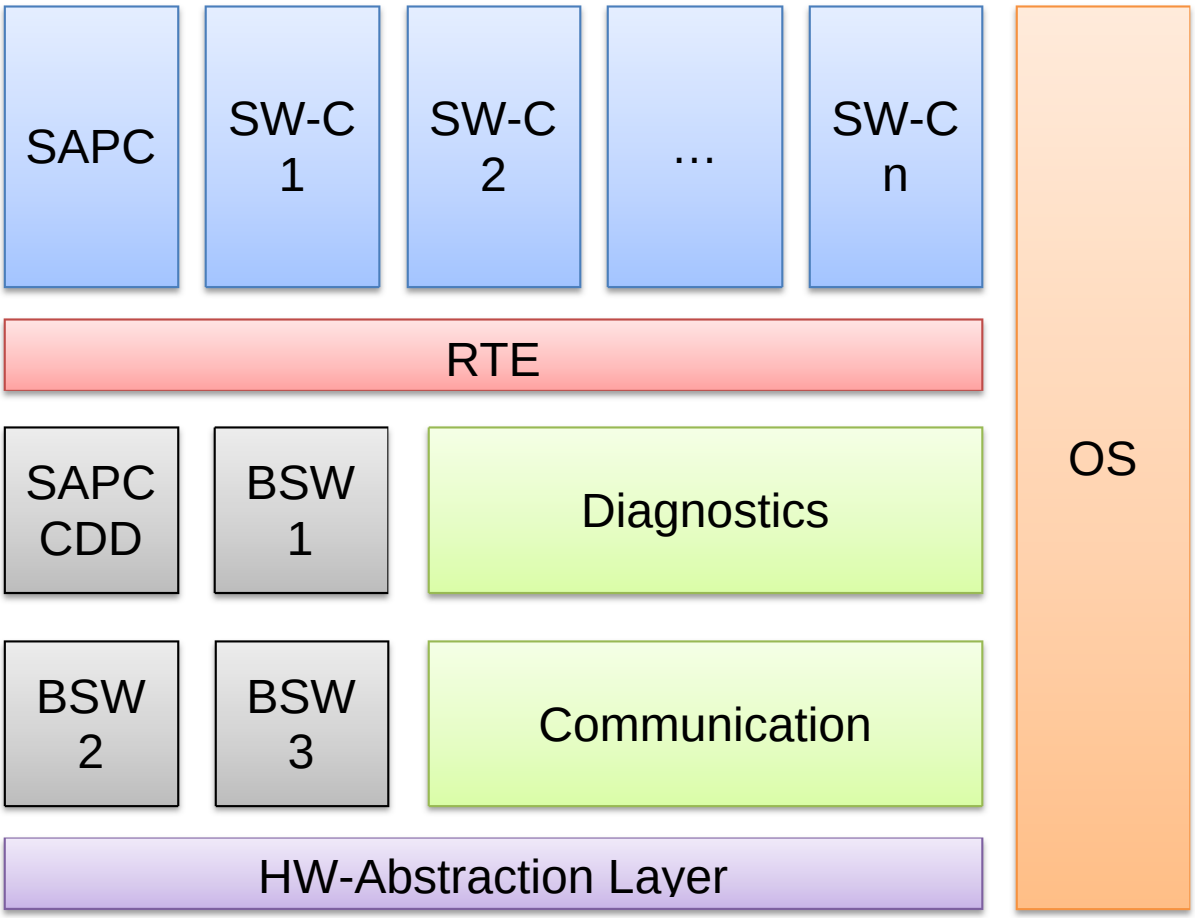

Figure 5: General SW Architecture similar to AUTOSAR

A complementary complex driver (CDD) below the RTE will serve the interfaces of the SAPC and acts as a wrapper towards the used operating system (in case of Delphi it will be the real-time OS PXROS-HR[7]). The complex driver for sure will be unique for all the different platforms in an adaptive system.

To make the SAPC work properly some essential data about the system's health status needs to be exchanged. For that reason, a System Health Vector has been designed. This vector contains, generally speaking, the information about the current status of each application that can be reconfigured during the adaptation process and some more information about the different platform operating states.

During the adaptation process it needs to be ensured that all SAPCs on the different platforms will come to the same result how the reconfigured system will look like. Therefore a database is implemented that contains all the necessary information about the applications, the restrictions for adaptation, the needed processing power and memory space. The database will be updated among the platforms in case that one platform got an additional SW-component installed.

[7] https://www.hightec-rt.com/en/downloads/pxros-hr.html

In case of Delphi's TMDP the reconfiguration process can also take place on the platform itself. Applications might be deactivated in one memory partition and then be restarted in a different memory area, or assigned processing core can be changed.

To make sure that not an unnecessary adaptation process is started, the system shall also support a rudimentary power mode management via network. The different states of the platforms need to be published in the system, and in case of sleep and wake-up organized.

Delphi is planning to introduce an ACC application in the project and therefore need to have access to front radar data. This can be either done by Delphi's own RACAM or by the Siemens radar in the front of the vehicle. Nevertheless, the radar input shall be received via the Ethernet connection to the TTTech switches and gateways. The ACC application shall be part of the number of reconfigurable applications.

For detailed information about the requirements and constraints, please have a look into the provided Excel sheet.

2.3.4 Safety

Safety is one of the most relevant issues for the SafeAdapt project. "ISO 26262 Road vehicles – Functional safety" is the standard that will lead the safety requirements for reconfiguration in SafeAdapt.

ISO 26262 provides an automotive-specific risk-based approach to determine integrity levels (Automotive Safety Integrity Levels – ASIL) of the functions to be installed in a vehicle. These ASIL levels (D, C, B, A, QM) are the basis for prioritization of the rules of the SafeAdapt Core reconfiguration.

Less critical functions (QM, A or B) will be the ones to be passivized in order to maintain the most critical ones (D and C) active. The adaptation will take into account the concept of *dependant functions*; that is, the properly functioning of some of them can depend on the well-functioning of others, so some constraints regarding this point can be applied when activating or passivating functions.

A hazard analysis has been initially done to identify the different ASIL levels of the functions. Moreover, a hazard analysis of the reconfiguration process itself is being done. This will help to identify specific safety goals for the reconfiguration process.

Fault tolerance will be achieved by means of defining an internal or external safety mechanism in order to control or mitigate failure modes, such as hot standby in case of SW safety mechanism.

Most of the safety requirements will be applied when the vehicle is running. Anyhow, installation of new components or update of functions will be taken into account. The SAPC should be properly configured and initiated on all platforms after one of these operations has been performed, just to ensure that the vehicle is ready for a safe reconfiguration process if needed.

2.3.5 Energy Efficiency

In conventional vehicles the energy is a scarce resource. In FEVs, the energy consumption becomes critical, as it is also needed for the vehicle powertrain. The SafeAdapt Platform Core is designed to be used in FEVs and therefore energy efficiency needs to be considered.

The concept of safe adaptation helps to make FEVs more energy efficient. Safe adaptation allows using redundancy mechanisms other than additional hardware. By reducing the number of ECUs, the weight of the vehicle is reduced, and thus, the energy consumption of the powertrain.

SAPC rules will take safety classification of the functions as the most relevant criteria to perform reconfiguration. Anyway, once safety issues have been considered, energy efficiency optimisation criteria can be also used in a second step to decide on how to perform reconfiguration. For example, keeping a required hot standby online will always precede switching it off for saving the energy. But if multiple configurations are available, the SAPC should consider using the configuration that is most energy efficient according to its data available. Furthermore, energy efficiency must be considered in case energy levels can produce a future foreseen safety relevant situation (as in UC_511_01).

The SAPC should be designed so it can also be used to support the smart use of CPU resources available. For example, safe adaptation can facilitate disabling not needed functions and enables other system optimisations during runtime (e.g. joint resource usage) to save additional energy.

The mechanisms used to trigger and calculate the optimisation of energy efficiency should be designed to reuse the available capabilities of the SAPC as much as possible.

Bibliography

ALCATEL * ALSTOM * ANSALDO SIGNAL * BOMBARDIER * INVENSYS RAIL * SIEMENS. (n.d.). *FIS for Man-Machine Interface.* ERTMS/ETCS Specifications.

ALCATEL * ALSTOM * ANSALDO SIGNAL * BOMBARDIER * INVENSYS RAIL * SIEMENS. (n.d.). *FIS for the Train Interface.* ERTMS/ETCS Specifications.

ALCATEL * ALSTOM * ANSALDO SIGNAL * BOMBARDIER * INVENSYS RAIL * SIEMENS. (n.d.). *System Requirement Specification.* ERTMS/ETCS Specifications.

Artisan. (2012). Retrieved June 20, 2012, from Artisan Studio - Products - Atego: http://www.atego.com/products/artisan-studio/

CENELEC. (1999). *EN 50126 - Railway applications — The specification and demonstration of reliability, availability, maintainability and safety (RAMS).*

CENELEC. (2003). *EN 50129 - Railway applications — Communication, signalling and processing systems — Safety related electronic systems for signalling.*

CENELEC. (2011). *EN 50128 - Railway applications — Communication, signalling and processing systems — Software for railway control and protection systems.*

EC. (1996). Council Directive 96/48/EC of 23 July 1996 on the interoperability of the trans-European high-speed rail system. *Official Journal L235* , pp. 6-24.

EC. (2001, april). Directive 2001/16/EC of the European Parliament and of the Council of 19 March 2001 on the interoperability of the trans-European conventional rail system . *Official Journal L110*, pp. 1-27.

EC. (2008). Directive 2008/57/EC of the European Parliament and of the Council of 17 June 2008 on the interoperability of the rail system within the Community. *Official Journal L191 vol 51*.

EC. (n.d.). *Commission Decision 2012/88/EU on the 25th January 2012 on the technical specification for interoperability relating to the control-command and signalling subsystems of the trans-European rail system.*

OMG. (2012). Retrieved June 8, 2012, from Website of the Object Management Group - Systems Modeling Language version 1.3: http://www.omgsysml.org/

OPENCOSS. (2012-06-28). *D1.1 Constraints of the certification process.*

OPENCOSS. (2012-06-28). *D1.2a Automitive Use Case.*

OPENCOSS. (2012-06-28). *D1.2a Automotive Use Case.*

OPENCOSS. (2012-06-28). *D1.2b Avionic Use Case.*

OPENCOSS. (2012-06-28). *D1.2c Railway Use Case.*

Ward, P., & Mellor, S. (1985). *Structured Development for Real-Time Systems.* New Jersey: Prence Hall.

List of Abbreviations

DOORS	Dynamic Object-Oriented Requirements System
SAE	Society of Automotive Engineers
SAPC	Safe Adaptation Platform Core
SafeAdapt	Safe Adaptive Software for Fully Electric Vehicles
TTEthernet	Time-Triggered Ethernet (SAE standard SAE AS6802)
TMDP	Trusted Multi Domain Platform
RACE	Robust and reliable Automotive Computing Environment
ECU	Electronic Control Unit

Annex

The annex of this document consists of the EXCEL file hosting the requirements captured in the EXCEL format compliant to DOORS input file in order to allow using DOORS if required. The file name of the DOORS compliant requirements sheets, which form part of this document, is:

SafeAdapt_D2-2_Requirements.xlsx

Requirement ID: <Participant>-<Req. ID>	Category <E.g. Functional / non-Functional>	Sub Category <HW, SW, ...>	Short Description: <Req. Name>	Description: <Req. Description>	Verification Method <Description how to verify>	Rationale: <The rationale behind this req.>	Dependencies: <Req. to either req.>	Conflicts: <Conflicts with other req.>	Date: <DD.MM.YYYY>	Supporting Material: <Additional information>	Object Status: <New, Changed, etc.>	Object Version <Version No.>	Review: <Comment>
DEL-001	Non-Functional	System	Network Topology - Double star architecture - Sensors/Actuators [1]	All sensors and actuators that are not related to a fail operational application (x-by-wire) shall be connected to at least one of the gateways that enable the interfacing with the TTEthernet network unless otherwise specified. The distribution (which sensor/actuator is connected to which gateway) need to be defined according to the mounting locations					16.04.2014		New	1	
DEL-002	Non-Functional	System	Network Topology - Double star architecture - Sensors/Actuators [2]	All sensors and actuators that are related to a fail operational application (x-by-wire) shall be connected to different gateways that enable the interfacing with the TTEthernet network. The distribution (which sensor/actuator is connected to which gateway) need to be defined to fulfil the necessary redundancy.					16.04.2014		New	1	
DEL-003	Non-Functional	System	Network Topology - Double star architecture - Gateways [1]	The network shall have at least two gateways for the sensors/actuators. The gateways shall enable the interfacing with the TTEthernet network.					16.04.2014		New	1	
DEL-004	Non-Functional	System	Network Topology - Double star architecture - Gateways [2]	Each gateway in the network shall be connected to each available TTEthernet switch in the network to support redundant communication lanes.					16.04.2014		New	1	
DEL-005	Non-Functional	System	Network Topology - Double star architecture - Switches	The network shall have at least two TTEthernet switches which shall be connected to each ECU.					16.04.2014		New	1	
DEL-006	Non-Functional	System	Network Topology - Double star architecture - ECUs	The network shall at least contain two ECUs (RACE from Siemens and TMDP from Delphi) that are hosting the applications.					16.04.2014		New	1	
DEL-007	Non-Functional	System	Network Topology - Double star architecture - Common requirements	The network structure shall be flexible enough to support additional ECUs, switches and gateways to be scalable for future extensions.					16.04.2014		New	1	
DEL-008	Non-Functional	Hardware	RaCam Interface - Gatewaying to TTEthernet	The interface of Delphi's RaCam is CAN. The messages will be sent out in a CAN-burst (see the following requirements for further information). At least one of the gateways shall be capable to gateway these messages to the TTEthernet network.		RaCam module specification.			16.04.2014		New	1	
DEL-009	Non-Functional	Software	RaCam Interface - Specification of Radar CAN messages	The radar completes the processing of the target data with a cycle time of 50 msec +/- 5 msec. At the completion of this processing, the radar shall transmit all its CAN messages in one group. The instrumentation buffers shall not overflow as a result of these sequential CAN messages. The spacing between the messages in the group should be minimized. The spacing between the groups of messages should be 50 msec +/- 5 msec.		RaCam module specification.			16.04.2014		New	1	
DEL-010	Non-Functional	Software	RaCam Interface - Specification of CAN burst messages [1]	1 x Status Message 1 (4E0h) 1 x Status Message 2 (4E1h) 1 x Status Message 3 (4E2h) 1 x Status Message 4 (4E3h)		RaCam module specification.			16.04.2014		New	1	
DEL-011	Non-Functional	Software	RaCam Interface - Specification of CAN burst messages [2]	1 x Track Message 1 (500h) : 1 x Track Message 64 (53Fh)		RaCam module specification.			16.04.2014		New	1	
DEL-012	Non-Functional	Software	RaCam Interface - Specification of CAN burst messages [3]	1 x Motion/Power Message (540h) with CAN_TX_TRACK_CAN_ID_GROUP = 0 1 x Motion/Power Message (540h) with CAN_TX_TRACK_CAN_ID_GROUP = 9 (time multiplexed)		RaCam module specification.			16.04.2014		New	1	
DEL-013	Non-Functional	Software	RaCam Interface - Specification of CAN burst messages [4]	1 x Validation Message 1 (5D0h) (only if Validation message transmission is enabled during development) 1 x Validation Message 2 (5D1h) (only if Validation message transmission is enabled during development)		RaCam module specification.			16.04.2014		New	1	
DEL-014	Non-Functional	Software	RaCam Interface - Specification of CAN burst messages [5]	1 x Status Message 5 (5E4h) 1 x Status Message 5 (5E5h) 1 x Status Message 5 (5E6h) 1 x Status Message 5 (5E7h) 1 x Status Message 5 (5E8h)		RaCam module specification.			16.04.2014		New	1	
DEL-015	Non-Functional	Software	RaCam Interface - Specification of CAN burst messages [6]	The FLS shall transmit these CAN messages in the following order. It is important that the Track 64 data is sent last because this will trigger the beginning of the instrumentation's processing. Start-4E0h-4E1h-4E2h-4E3h-500h...53Fh-540h(msg 1 to 10)-5E4h-5E5h-5E6h-5E7h-5E8h-End		RaCam module specification.			16.04.2014		New	1	
DEL-016	Non-Functional	Hardware	Realization of TTEthernet switch on TMDP	Integration of TTEthernet technology and the TMDP hardware					16.04.2014		New	1	
DEL-017	Non-Functional	Hardware	Cruise control switches	The vehicle shall have switches installed that support commanding the ACC functionality. The following switches shall be installed: - Cancel Switch - Set Switch		Requirement for Delphi's ACC/AEB software component.			16.04.2014		New	1	
DEL-018	Non-Functional	Hardware	CrsCntrlSwStCanSwAct sensor value for ACC application	The network shall supply the sensor status of the "Cruise Control Switch Status : Cancel Switch Active" coded in the signal CrsCntrlSwStCanSwAct. The switch can be active or in-active (1 Bit).		Requirement for Delphi's ACC/AEB software component.			16.04.2014		New	1	
DEL-019	Non-Functional	Hardware	CrsCntrlSwStSetSwAct sensor value for ACC application	The network shall supply the sensor status of the "Cruise Control Switch Status : Set Switch Active" coded in the signal CrsCntrlSwStSetSwAct. The switch can be active or in-active (1 Bit).		Requirement for Delphi's ACC/AEB software component.			16.04.2014		New	1	
DEL-020	Non-Functional	System	BPDAPS_BkPDrvApP sensor value for ACC application	The network shall supply the sensor status of the "Brake Pedal Driver Applied Pressure Status : Brake Pedal Driver Applied Pressure" coded in the signal BPDAPS_BkPDrvApP. The signal shall be coded in 8 Bits.		Requirement for Delphi's ACC/AEB software component.			16.04.2014		New	1	
DEL-021	Non-Functional	System	BrkSysAutBrkFld system status value for ACC application	The network shall supply the system status of the "Brake System Automatic Braking Failed" coded in the signal BrkSysAutBrkFld. The signal shall be coded in 1 Bit.		Requirement for Delphi's ACC/AEB software component.			16.04.2014		New	1	
DEL-022	Non-Functional	System	BrkPdlPos sensor value for ACC application	The network shall supply the sensor status of the "Brake Pedal Position" coded in the signal BrkPdlPos. The signal shall be coded in 8 Bits.		Requirement for Delphi's ACC/AEB software component.			16.04.2014		New	1	
DEL-023	Non-Functional	System	BrkPedTrvlAchvd sensor value for ACC application	The network shall supply the sensor status of the "Brake Pedal Initial Travel Achieved Status : Brake Pedal Initial Travel Achieved" coded in the signal BrkPedTrvlAchvd. The signal shall be coded in 1 Bit.		Requirement for Delphi's ACC/AEB software component.			16.04.2014		New	1	
DEL-024	Non-Functional	System	BrkPdlDrvAppPrsDetcd sensor value for ACC application	The network shall supply the sensor status of the "Brake Pedal Driver Applied Pressure Detected" coded in the signal BrkPdlDrvAppPrsDetcd. The signal shall be coded in 1 Bit.		Requirement for Delphi's ACC/AEB software component.			16.04.2014		New	1	
DEL-025	Non-Functional	System	AccPdlOvrrdAtv sensor value for ACC application	The network shall supply the sensor status of the "Accelerator Pedal Override Active" coded in the signal AccPdlOvrrdAtv. The signal shall be coded in 1 Bit.		Requirement for Delphi's ACC/AEB software component.			16.04.2014		New	1	
DEL-026	Non-Functional	System	VehSpdAvgDrvn system status value for ACC application	The network shall supply the system status of the "Vehicle Speed Average Driven" coded in the signal VehSpdAvgDrvn. The signal shall be coded in 15 Bits.		Requirement for Delphi's ACC/AEB software component.			16.04.2014		New	1	
DEL-027	Non-Functional	System	VehOdo sensor value for ACC application	The network shall supply the sensor status of the "Vehicle Odometer" coded in the signal VehOdo. The signal shall be coded in 32 Bits.		Requirement for Delphi's ACC/AEB software component.			16.04.2014		New	1	
DEL-028	Non-Functional	System	VehMovState system status value for ACC application	The network shall supply the system status of the "Vehicle Movement State" coded in the signal VehMovState. The signal shall be coded in 3 Bits.		Requirement for Delphi's ACC/AEB software component.			16.04.2014		New	1	
DEL-029	Non-Functional	System	ActVehAccel sensor value for ACC application	The network shall supply the sensor status of the "Actual Vehicle Acceleration" coded in the signal ActVehAccel. The signal shall be coded in 12 Bits.		Requirement for Delphi's ACC/AEB software component.			16.04.2014		New	1	
DEL-030	Non-Functional	System	EngTrqActExtRng system status value for ACC application	The network shall supply the system status of the "Engine Torque Actual Extended Range" coded in the signal EngTrqActExtRng. The signal shall be coded in 12 Bits.		Requirement for Delphi's ACC/AEB software component.			16.04.2014		New	1	
DEL-031	Non-Functional	System	EngTrqDrRqdExtRng system status value for ACC application	The network shall supply the system status of the "Engine Torque Driver Requested Extended Range" coded in the signal EngTrqDrRqdExtRng. The signal shall be coded in 12 Bits.		Requirement for Delphi's ACC/AEB software component.			16.04.2014		New	1	
DEL-032	Non-Functional	System	EngTrqMaxExtRng system status value for ACC application	The network shall supply the system status of the "Engine Torque Maximum Extended Range" coded in the signal EngTrqMaxExtRng. The signal shall be coded in 12 Bits.		Requirement for Delphi's ACC/AEB software component.			16.04.2014		New	1	
DEL-033	Non-Functional	System	EngRunAtv system status value for ACC application	The network shall supply the system status of the "Engine Run Active" coded in the signal EngRunAtv. The signal shall be coded in 1 Bit.		Requirement for Delphi's ACC/AEB software component.			16.04.2014		New	1	
DEL-034	Non-Functional	System	EngSpd system status value for ACC application	The network shall supply the system status of the "Engine Speed" coded in the signal EngSpd. The signal shall be coded in 16 Bits.		Requirement for Delphi's ACC/AEB software component.			16.04.2014		New	1	
DEL-035	Non-Functional	System	DrvIntndAxlTrq system status value for ACC application	The network shall supply the system status of the "Driver Intended Axle Torque" coded in the signal DrvIntndAxlTrq. The signal shall be coded in 15 Bits.		Requirement for Delphi's ACC/AEB software component.			16.04.2014		New	1	
DEL-036	Non-Functional	System	DrvIntndAxlTrqMax system status value for ACC application	The network shall supply the system status of the "Driver Intended Axle Torque Maximum" coded in the signal DrvIntndAxlTrqMax. The signal shall be coded in 15 Bits.		Requirement for Delphi's ACC/AEB software component.			16.04.2014		New	1	
DEL-037	Non-Functional	System	DrvIntndAxlTrqMin system status value for ACC application	The network shall supply the system status of the "Driver Intended Axle Torque Minimum" coded in the signal DrvIntndAxlTrqMin. The signal shall be coded in 15 Bits.		Requirement for Delphi's ACC/AEB software component.			16.04.2014		New	1	
DEL-038	Non-Functional	System	PT_BrkPedDscrtInpStat sensor value for ACC application	The network shall supply the sensor status of the "Powertrain Brake Pedal Discrete Input Status" coded in the signal PT_BrkPedDscrtInpStat. The signal shall be coded in 8 Bits.		Requirement for Delphi's ACC/AEB software component.			16.04.2014		New	1	
DEL-039	Non-Functional	System	PrkBrkSwAtv sensor value for ACC application	The network shall supply the sensor status of the "Park Brake Switch Active" coded in the signal PrkBrkSwAtv. The signal shall be coded in 1 Bit.		Requirement for Delphi's ACC/AEB software component.			16.04.2014		New	1	
DEL-040	Non-Functional	System	StrWhlAng sensor value for ACC application	The network shall supply the sensor status of the "Steering Wheel Angle" coded in the signal StrWhlAng. The signal shall be coded in 16 Bits.		Requirement for Delphi's ACC/AEB software component.			16.04.2014		New	1	
DEL-041	Non-Functional	System	StrWhlAngGrd sensor value for ACC application	The network shall supply the sensor status of the "Steering Wheel Angle Gradient" coded in the signal StrWhlAngGrd. The signal shall be coded in 12 Bits.		Requirement for Delphi's ACC/AEB software component.			16.04.2014		New	1	
DEL-042	Non-Functional	System	StrWhlAngSenCalStat sensor value for ACC application	The network shall supply the sensor status of the "Steering Wheel Angle Sensor Calibration Status" coded in the signal StrWhlAngSenCalStat. The signal shall be coded in 2 Bits.		Requirement for Delphi's ACC/AEB software component.			16.04.2014		New	1	
DEL-043	Non-Functional	System	ACCBrkngAct system status value for ACC application	The network shall supply the system status of the "Adaptive Cruise Control Braking Active" coded in the signal ACCBrkngAct. The signal shall be coded in 1 Bit.		Requirement for Delphi's ACC/AEB software component.			16.04.2014		New	1	
DEL-044	Non-Functional	System	AdptCrsGapSwAct system status value for ACC application	The network shall supply the system status of the "Adaptive Cruise Control Gap Switch Activation" coded in the signal AdptCrsGapSwAct. The signal shall be coded in 2 Bits.		Requirement for Delphi's ACC/AEB software component.			16.04.2014		New	1	
DEL-045	Non-Functional	System	SysPwrMd system status value for ACC application	The network shall supply the system status of the "System Power Mode" coded in the signal SysPwrMd. The signal shall be coded in 2 Bits.		Requirement for Delphi's ACC/AEB software component.			16.04.2014		New	1	
DEL-046	Non-Functional	System	AutoBrkngAct system status value for ACC application	The network shall supply the system status of the "Automatic Braking Active" coded in the signal AutoBrkngAct. The signal shall be coded in 1 Bit.		Requirement for Delphi's ACC/AEB software component.			16.04.2014		New	1	
DEL-047	Non-Functional	System	ABAAPS_AccPos system status value for ACC application	The network shall supply the system status of the "Autonomous Braking Accelerator Actual Position Status : Accelerator Position" coded in the signal ABAAPS_AccPos. The signal shall be coded in 8 Bits.		Requirement for Delphi's ACC/AEB software component.			16.04.2014		New	1	
DEL-048	Non-Functional	System	RdLlNomAxlTrq system status value for ACC application	The network shall supply the system status of the "Road Load Nominal Axle Torque" coded in the signal RdLlNomAxlTrq. The signal shall be coded in 19 Bits.		Requirement for Delphi's ACC/AEB software component.			16.04.2014		New	1	
DEL-049	Non-Functional	System	BrkSysAutBrkStat system status value for ACC application	The network shall supply the system status of the "Brake System Automatic Braking Status" coded in the signal BrkSysAutBrkStat. The signal shall be coded in 2 Bits.		Requirement for Delphi's ACC/AEB software component.			16.04.2014		New	1	
DEL-050	Non-Functional	System	IMULatAccPrim sensor value for ACC application	The network shall supply the sensor status of the "Inertial Measurement Unit Lateral Acceleration Primary" coded in the signal IMULatAccPrim. The signal shall be coded in 10 Bits.		Requirement for Delphi's ACC/AEB software component.			16.04.2014		New	1	

DEL-051	Non-Functional	System	IMULonAccPri sensor value for ACC application	The network shall supply the sensor status of the "Inertial Measurement Unit LongitudinalAcceleration Primary" coded in the signal IMULonAccPri. The signal shall be coded in 12 Bits.		Requirement for Delphi's ACC/AEB software component.			16.04.2014		New	1	
DEL-052	Non-Functional	System	IMURollCntSec sensor value for ACC application	The network shall supply the sensor status of the "Inertial Measurement Unit Rolling Count Secondary" coded in the signal IMURollCntSec. The signal shall be coded in 2 Bits.		Requirement for Delphi's ACC/AEB software component.			16.04.2014		New	1	
DEL-053	Non-Functional	System	IMUYawRtPri sensor value for ACC application	The network shall supply the sensor status of the "Inertial Measurement Unit Yaw Rate Primary" coded in the signal IMUYawRtPri. The signal shall be coded in 13 Bits.		Requirement for Delphi's ACC/AEB software component.			16.04.2014		New	1	
DEL-054	Functional	System	Output of ACC application signal ACCAct370	The output of the ACC application shall contain the signal ACCAct370. The signal's long name is "Adaptive Cruise Control Active" and shall be coded in 1 Bit.		Requirement for Delphi's ACC/AEB software component.			16.04.2014		New	1	
DEL-055	Functional	System	Driver information [1]	The content of the signal ACCAct370 shall be displayed to the driver in a way that needs to be defined. (to be discussed with SIE)		Requirement for Delphi's ACC/AEB software component.			16.04.2014		New	1	
DEL-056	Functional	Software	Output of ACC application signal ACCATC_AxlTrqRq	The output of the ACC application shall contain the signal ACCATC_AxlTrqRq. The signal's long name is "Adaptive Cruise Control Axle Torque Command : Axle Torque Request" and shall be coded in 10 Bits.		Requirement for Delphi's ACC/AEB software component.			16.04.2014		New	1	
DEL-057	Functional	Software	Output of ACC application signal FOAI_AlrtWrnIndRq	The output of the ACC application shall contain the signal FOAI_AlrtWrnIndRq. The signal's long name is "Forward Object Alert Indications : Alert Warning Indication Request" and shall be coded in 4 Bits.		Requirement for Delphi's ACC/AEB software component.			16.04.2014		New	1	
DEL-058	Functional	Software	Output of ACC application signal FOAI_VehAhdIndRq	The output of the ACC application shall contain the signal FOAI_VehAhdIndRq. The signal's long name is "Forward Object Alert Indications : Vehicle Ahead Indication Request" and shall be coded in 4 Bits.		Requirement for Delphi's ACC/AEB software component.			16.04.2014		New	1	
DEL-059	Functional	Software	Output of ACC application signal FOAI_AlrtChmInhbRq	The output of the ACC application shall contain the signal FOAI_AlrtChmInhbRq. The signal's long name is "Forward Object Alert Indications : Alert Warning Chime Inhibit Request" and shall be coded in 1 Bit.		Requirement for Delphi's ACC/AEB software component.			16.04.2014		New	1	
DEL-060	Functional	Software	Output of ACC application signal ACCDrvrSelItdSpd_kph	The output of the ACC application shall contain the signal ACCDrvrSelItdSpd_kph. The signal's long name is "Adaptive Cruise Control Driver Selected Speed in kph" and shall be coded in 12 Bits.		Requirement for Delphi's ACC/AEB software component.			16.04.2014		New	1	
DEL-061	Functional	Software	Driver information [2]	The content of the signal ACCDrvrSelItdSpd_kph shall be displayed to the driver in a way that needs to be defined. (to be discussed with SIE)		Requirement for Delphi's ACC/AEB software component.			16.04.2014		New	1	
DEL-062	Functional	Software	Output of ACC application signal ACCDrvrSelItdSpdIO	The output of the ACC application shall contain the signal ACCDrvrSelItdSpdIO. The signal's long name is "Adaptive Cruise Control Driver Selected Speed Indication On" and shall be coded in 1 Bit.		Requirement for Delphi's ACC/AEB software component.			16.04.2014		New	1	
DEL-063	Functional	Software	Output of ACC application signal ACCUnavlbleDTWthrIO	The output of the ACC application shall contain the signal ACCUnavlbleDTWthrIO. The signal's long name is "Adaptive Cruise Unavailable Due To Weather Indication On" and shall be coded in 1 Bit.		Requirement for Delphi's ACC/AEB software component.			16.04.2014		New	1	
DEL-064	Functional	Software	Driver information [3]	The content of the signal ACCUnavlbleDTWthrIO shall be displayed to the driver in a way that needs to be defined. (to be discussed with SIE)		Requirement for Delphi's ACC/AEB software component.			16.04.2014		New	1	
DEL-065	Functional	Software	Output of ACC application signal DrvThrtlOvrdIO	The output of the ACC application shall contain the signal DrvThrtlOvrdIO. The signal's long name is "Driver Throttle Override Indication On" and shall be coded in 1 Bit.		Requirement for Delphi's ACC/AEB software component.			16.04.2014		New	1	
DEL-066	Functional	Software	Output of ACC application signal ACCHdwaySttngIO	The output of the ACC application shall contain the signal ACCHdwaySttngIO. The signal's long name is "Adaptive Cruise Control Headway Setting Indication On" and shall be coded in 1 Bit.		Requirement for Delphi's ACC/AEB software component.			16.04.2014		New	1	
DEL-067	Functional	Software	Output of ACC application signal ACCTmpUnavlbleIO	The output of the ACC application shall contain the signal ACCTmpUnavlbleIO. The signal's long name is "Adaptive Cruise Control Temporarily Unavailable Indication On" and shall be coded in 1 Bit.		Requirement for Delphi's ACC/AEB software component.			16.04.2014		New	1	
DEL-068	Functional	Software	Driver information [4]	The content of the signal ACCTmpUnavlbleIO shall be displayed to the driver in a way that needs to be defined. (to be discussed with SIE)		Requirement for Delphi's ACC/AEB software component.			16.04.2014		New	1	
DEL-069	Functional	Software	Output of ACC application signal ACCSnsClnRqdIO	The output of the ACC application shall contain the signal ACCSnsClnRqdIO. The signal's long name is "Adaptive Cruise Control Sensor Cleaning Requested Indication On" and shall be coded in 1 Bit.		Requirement for Delphi's ACC/AEB software component.			16.04.2014		New	1	
DEL-070	Functional	Software	Driver information [5]	The content of the signal ACCSnsClnRqdIO shall be displayed to the driver in a way that needs to be defined. (to be discussed with SIE)		Requirement for Delphi's ACC/AEB software component.			16.04.2014		New	1	
DEL-071	Functional	Software	Output of ACC application signal ServAdpCrsCtrlIndOn	The output of the ACC application shall contain the signal ServAdpCrsCtrlIndOn. The signal's long name is "Service Adaptive Cruise Control Indication On" and shall be coded in 1 Bit.		Requirement for Delphi's ACC/AEB software component.			16.04.2014		New	1	
DEL-072	Functional	Software	Driver information [6]	The content of the signal ServAdpCrsCtrlIndOn shall be displayed to the driver in a way that needs to be defined. (to be discussed with SIE)		Requirement for Delphi's ACC/AEB software component.			16.04.2014		New	1	
DEL-073	Functional	Software	Output of ACC application signal FrtRdrBlckdIO	The output of the ACC application shall contain the signal FrtRdrBlckdIO. The signal's long name is "Front Radar Blocked Indication On" and shall be coded in 1 Bit.		Requirement for Delphi's ACC/AEB software component.			16.04.2014		New	1	
DEL-074	Functional	Software	Output of ACC application signal VADIR_FlwDst	The output of the ACC application shall contain the signal VADIR_FlwDst. The signal's long name is "Vehicle Ahead Distance Indication Request : Following Distance" and shall be coded in 8 Bit.		Requirement for Delphi's ACC/AEB software component.			16.04.2014		New	1	
DEL-075	Functional	Software	Driver information [7]	The content of the signal VADIR_FlwDst shall be displayed to the driver in a way that needs to be defined. (to be discussed with SIE)		Requirement for Delphi's ACC/AEB software component.			16.04.2014		New	1	
DEL-076	Functional	Software	Graceful degradation for steer-by-wire	In case of an application or HW error on the platform that is originally executing the steer-by-wire (SBW) functionality, a degraded SBW function shall be deployed on another operative ECU in the system. The TMDP shall be able to execute the degraded SBW function.					16.04.2014		New	1	
DEL-077	Functional	Software	Graceful degradation for steer-by-wire - Force feedback	A degraded steer-by-wire (SBW) function shall not support force feedback.					16.04.2014		New	1	
DEL-078	Functional	Software	Graceful degradation for steer-by-wire - Steering dynamics	A degraded steer-by-wire (SBW) function shall not support speed dependant power aided steering.					16.04.2014		New	1	
DEL-079	Functional	Software	Health vector - content description	A health vector will be used for exchanging adaptation-dependent information between platforms					30.07.2014		Changed	2	
DEL-080	Non-Functional	Hardware	Steer-by-wire sensor integration into network	The steer-by-wire sensors that a absolutely necessary for operation shall be available 3 times to support voting on the inputs. This also implies that each sensor shall be connected to a different network path (if only two paths are available then 2 sensors need to be connected to one path, one to the other).		Neccessary to allow detection of wrong but plausible sensor values.			16.04.2014		New	1	
DEL-081	Functional	Software	Voting on ECU level - in case of triplex sensor layout	If three redundant sensor inputs for the steer-by-wire function are available the voting for the correct sensor value shall be done on ECU level.		Centralized voting does not make much sense in an adaptive system.			16.04.2014		New	1	
DEL-082	Functional	Software	Voting on ECU level - in case of triplex actuator layout	If three redundant actuators for the steer-by-wire function are available the voting for the correct actuator to be used shall be done on ECU level.		Centralized voting does not make much sense in an adaptive system.			16.04.2014		New	1	
DEL-083	Functional	Software	ON/OFF requirements for ACC [1]	In case that the signal CrsCntrlSwStSetSwAct ("Cruise Control Switch Status : Set Switch Active") is set to 1 and the vehicle speed is higher than 30 kph the ACC shall be activated.					16.04.2014		New	1	
DEL-084	Functional	Software	ON/OFF requirements for ACC [2]	In case that the signal CrsCntrlSwStCanSwAct ("Cruise Control Switch Status : Cancel Switch Active") is set to 1 the ACC shall be de-activated.					16.04.2014		New	1	
DEL-085	Functional	Software	ON/OFF requirements for ACC [3]	In case that the vehicle speed drops to 30 kph or below the ACC shall be de-activated.					16.04.2014		New	1	
DEL-086	Functional	Software	ON/OFF requirements for ACC [4]	In case that the driver presses the brake-pedal the ACC shall be de-activated.					16.04.2014		New	1	
DEL-087	Functional	Software	ON/OFF requirements for AEB [1]	In case that the vehicle speed drops below 30 kph the AEB shall become activated.					16.04.2014		New	1	
DEL-088	Functional	Software	ON/OFF requirements for AEB [2]	In case that the vehicle speed is 30 kph or higher the AEB shall be de-activated.					16.04.2014		New	1	
DEL-089	Functional	Software	Definition of plausability check - CrsCntrlSwStSetSwAct ("Cruise Control Switch Status : Set Switch Active") signal for ACC [1]	In case that the signal CrsCntrlSwStSetSwAct ("Cruise Control Switch Status : Set Switch Active") becomes stuck at 1 (after 500 ms permanently at 1) the ACC shall only be activated once during the actual ignition cycle.		To avoid unwanted activation / de-activation of the function.			16.04.2014		New	1	
DEL-090	Functional	Software	Definition of plausability check - CrsCntrlSwStSetSwAct ("Cruise Control Switch Status : Set Switch Active") signal for ACC [2]	In case that the signal CrsCntrlSwStSetSwAct ("Cruise Control Switch Status : Set Switch Active") is already stuck at 1 (after 500 ms permanently at 1) after ignition on, the ACC shall not be activated during the actual ignition cycle.		To avoid unwanted activation / de-activation of the function.			16.04.2014		New	1	
DEL-091	Functional	Software	Definition of plausability check - CrsCntrlSwStCanSwAct ("Cruise Control Switch Status : Cancel Switch Active") signal for ACC [1]	In case that the signal CrsCntrlSwStCanSwAct ("Cruise Control Switch Status : Cancel Switch Active") becomes stuck at 1 (after 500 ms permanently at 1) the ACC shall be deactivated immediately and it shall not be possible to activate the ACC again during the actual ignition cycle.		To avoid unwanted activation / de-activation of the function.			16.04.2014		New	1	
DEL-092	Functional	Software	Definition of plausability check - CrsCntrlSwStCanSwAct ("Cruise Control Switch Status : Cancel Switch Active") signal for ACC [2]	In case that the signalCrsCntrlSwStCanSwAct ("Cruise Control Switch Status : Cancel Switch Active") is already stuck at 1 (after 500 ms permanently at 1) after ignition on, it shall not be possible to activate the ACC during the actual ignition cycle.		To avoid unwanted activation / de-activation of the function.			16.04.2014		New	1	
DEL-093	Functional	Software	Passivation of comfort functions [1]	In case of an adaptation scenario the passivation of not needed comfort functions to free up memory and processing power shall be supported by the TMDP.					16.04.2014		New	1	
DEL-094	Functional	Software	Passivation of comfort functions [2]	In case of a state of charge below 20% the passivation of not needed comfort functions to reduce the overall power consumption shall be supported by the TMDP.					16.04.2014		New	1	
DEL-095	Functional	Software	Adaptation of ACC/AEB functionality	In case of a detected failure on the TMDP platform that lets the ACC and AEB functionality become unsafe, the ACC/AEB functionality shall be switched off immediately and shall be executed on a different platform.					16.04.2014		New	1	
DEL-096	Non-Functional	Hardware	Power supply [1]	TMDP shall be supplied by two clamp 30 wires (cl. 30a & cl. 30b) that are independently supplied by a power distribution box.		Redundant power supply			17.04.2014 / 29.04.2014		Changed	2	
DEL-097	Non-Functional	Hardware	Power supply [2]	The power supply lines shall support at least 5 A current.					29.04.2014		New	1	
DEL-098	Non-Functional	Hardware	Power supply [3]	The TMDP shall be protected against over-voltage and reverse polarity.					29.04.2014		New	1	
DEL-099	Non-Functional	Hardware	Power supply [4]	The AURIX processor on TMDP shall be able to cut the power supply for the secondary processor board.		AURIX is master of power mode			29.04.2014		New	1	
DEL-100	Non-Functional	Hardware	Startup condition [1] - Clamp 15	The TMDP shall start up if the clamp 15 switches to high.					29.04.2014		New	1	
DEL-101	Non-Functional	Hardware	Startup condition [2] - Networking	The TMDP shall start up if a wake up over Ethernet and/or CAN is received.					29.04.2014		New	1	
DEL-101	Non-Functional	Hardware	Startup condition [3]	The TMDP shall NOT be able to be woken up via the second processor board.					29.04.2014		New	1	
DEL-102	Functional	Software	Stay awake condition	The TMDP shall stay awake as long as at least one ECU is NOT signalizing ready to sleep conditions.					29.04.2014		New	1	
DEL-103	Functional	Software	Sleep condition [1]	TMDP shall only go to sleep if all ECUs are signalizing ready to sleep AND the clamp 15 signal is low.					29.04.2014		New	1	
DEL-104	Functional	Software	Sleep condition [2]	An ECU shall only signalize ready to sleep if all applications running on this ECU are de-initialized and ready to terminate.					29.04.2014		New	1	
DEL-105	Non-Functional	Hardware	Diagnostic capabilities [1]	The TMDP power supply lines cl. 30a & cl. 30b shall be fully diagnosable.		To fulfil the ISO 26262 requirements for ASIL-D			29.04.2014		New	1	
DEL-106	Non-Functional	Hardware	General HW Requirements	The TMDP shall support the following general automotive HW requirements: - temperature of operation -40 to +85 centigrade - fully operational from 7V to 16V					29.04.2014		New	1	
DEL-107	Functional	Software	Safe Adaptation Runtime Core [1] - TMDP requirement referring to feature F1	TMDP will not use pre-configured memory partitions.					TMDP using PXROS is capable to dynamically allocate new protected memory regions for the faulty FCR (only referring to data storage in RAM, program code is stored in NVRAM due to AURIX architecture).		30.07.2014		New	1
DEL-108	Functional	System	Safe Adaptation Runtime Core [2] - System restart	The restart of the system by an ignition-cycle shall lead to a BIST (core check, walk pattern test, etc.) to check the sanity of the system. In case of no error found, the original run-schedule shall be started. This requirement only applies to the hardware, not to the applications!					30.07.2014		New	1	

DEL-109	Functional	Software	Safe Adaptation Runtime Core [3] - Diagnostics	The detection of a safety relevant error shall lead to a non self-healing error entry in the diagnostic application and only degraded functionality shall be made available. The user shall be asked to have a maintenance (-> driver information).		This is to avaiod that intermittent errors highly safety critical functions can be "cured" by just doing an ignition cycle.			30.07.2014		New	1	
DEL-110	Functional	Software	Safe Adaptation Runtime Core [4] - TMDP requirement referring to feature F5	Worst case execution time for each application shall be known to the safe adaptation runtime core.		Important to check if the activation of additional functions are within the time-budget.			30.07.2014		New	1	
DEL-111	Functional	Software	Safe Adaptation Runtime Core [5] - Handling of hot-standby applications	Hot standby applications shall have a flag to signal the safe adaptation runtime core that only a passivation in case of an error is allowed. To avoid multiple application execution in different CCCs it shall not be allowed to reactivate a hot standby application on the faulty CCC.		An error in a highly safety critical application that needs to be covered by a hot-standby application to fulfill timing requirements shall lead to it's passivation on the CCC. The standby application shall only be run on a different CCC.			30.07.2014		New	1	
CEA-001	Functional	Tools	Safe-Adaptation Core (SAC) has to be able to evaluate adaptation information generated by modeling tools	Model provides various information about SW system so for instance the information about which component is connected with which component and the resource requirements of the components and the timing constraints of the components. These information (model at run-time) needs to be accessable in order to be able to make adaptation at run-time.	By assuring that the information in model is respected.		CEA-005, CEA-006		01.09.2014	D4.1 (intermediate versions need to be provided)	New	1	
CEA-002	Functional	Software	SAC must be to instantiate, dispose and reconnect components	In order to make adaptation, SAC must instantiate, dispose and/or reconnect components. Depending on the criticality of the components, instantiation and disposal might correspond to activation and deactivation of pre-allocated components. In this sense. SAC must be capable of doing these tasks.	CEA-004				01.09.2014		New	1	
CEA-003	Functional	Software	Execution of adaptation scenarios must be possible	In particular, we must be able to inject events (via fake sensor data) that trigger adaptations and we can have subsequent related requirements that says that we must be able to verify adaptation scenarios.	By running scenarios, We use CEA-004 to examine that adaptation is actually executed.		CEA-004		01.09.2014		New	1	
CEA-004	Functional	Software	Observation of current SW configuration must be possible	Observation of current SW configuration (component allocations, activation status, etc.) must be possible (at least in simulations).	By running scenarios and obtaining observation data.	We want to be able to verify whether the adaptations are executed as planned or not.	CEA-003		01.09.2014		New	1	
CEA-005	Functional	Software	We must be able to execute state-machines	SAC must be able to execute state-machines that describe system modes.	By running adaptation scenarios based on state-machines, i.e. we run a scenario and typically expect it to change its state.	We want to be able to model system modes. System modes are modeled in forms of state-machines, and we want to assure that these models are respected. Note that it needs to be discussed if we really want to model system modes due to the possible combinatorial explosions.			01.09.2014		New	1	
TEC-001	Non-Functional	System	Reconfiguration of Failed Cruise Control(Cold-Standby)	Overall time response for safety-critical functionality should be less than 50ms (fault detection: 10ms; passivation: 10ms). In detail, fault-recovery takes less than: - 50ms for safety-critical functionality with required fail-operational behaviour. - 1000 ms for functionality with required fail-passive behaviour with regards to ISO26262 related requirements and required recovery to increase operational availability.					20.08.2014		New	1	
TEC-002	Non-Functional	System	Reconfiguration of Failed Cruise Control(Cold-Standby)	CSCC should register hardware partitions status					20.08.2014		New	1	
TEC-003	Non-Functional	System	Reconfiguration of Failed Cruise Control(Cold-Standby)	When HW partitions are available in the current CSCC duplex control computer, item functions (SW part) should be reconfigured using empty pre-configured partitions.					20.08.2014		New	1	
TEC-004	Non-Functional	System	Steer-By-Wire Adaptation after ECU-Failure (CSCC-Failover)	Overall time response should be less than 50ms, see TEC-001					20.08.2014		New	1	
TEC-004-01	Non-Functional	System	Steer-By-Wire Adaptation after ECU-Failure (CSCC-Failover)	The CCC supports fault-tolerance for applications that areexecuted. This fault-tolerance includes the possibility of having hot standby applications. The switchover-criteria as a performance-level of the considered appliciations. The performance-level can represent graceful degradation.									
TEC-005	Non-Functional	System	Adaptation after Break-by-Wire Malfunction (Dependable Function)	Overall time response should be less than 50ms (fault detection: 10ms; passivation: 10ms), see TEC-001					20.08.2014		New	1	
TEC-005-01	Non-Functional	System	Adaptation after Break-by-Wire Malfunction (Dependable Function)	The priority-based support for fault-tolerance can also consider additional constraints besides safety (such as data-transport delays) to select 1ooN reconfiguration possibilities.									
TEC-006	Non-Functional	System	Adaptation after Break-by-Wire Malfunction (Dependable Function)	CSCC should register hardware partitions status, see TEC-003					20.08.2014		New	1	
TEC-007	Non-Functional	System	Adaptation after Break-by-Wire Malfunction (Dependable Function)	CSCC should register applications with its priorities and interdependencies A CCC configurator can configure the priorities of applications based on a given set of rules. Furthermore, interdepenencies in between applications can be defined. Those interdependencies can be used to determine degradation-levels of applications.					20.08.2014		New	1	
TEC-008	Non-Functional	System	Adaptation after Break-by-Wire Malfunction (Dependable Function)	Mirrimising of applications passivation and reconfiguration should be accomplished					20.08.2014		New	1	
TEC-009	Non-Functional	System	Adaptation after Break-by-Wire Malfunction (Dependable Function)	Overall time response should be less than 50ms (fault detection: 10ms; passivation: 10ms), see TEC-001					20.08.2014		New	1	
TEC-010	Non-Functional	System	Adaptation after Break-by-Wire Malfunction (Dependable Function)	CSCC should registar hardware partitions status, see TEC-006.					20.08.2014		New	1	
TEC-011	Non-Functional	System	Adaptation after Break-by-Wire Malfunction (Dependable Function)	CSCC should register applications with its priorities and interdependencies, ee TEC-007.					20.08.2014		New	1	
TEC-012	Non-Functional	System	Adaptation after Break-by-Wire Malfunction (Dependable Function)	Mirrimising of applications passivation and reconfiguration should be accomplished, see TEC-008.					20.08.2014		New	1	
TEC-013	Non-Functional	System	Communication Failure with External Aggregate (Hot-Standby)	Overall time response should be less than 50ms (fault detection: 10ms; passivation: 10ms), see TEC-001.									
TEC-013-01	Non-Functional	System	Communication Failure with External Aggregate (Hot-Standby)	The fault-tolerance supports graceful degradation (and thus follows the strategie to support the best possible performance-level to the driver even in case of inconsistent failures).									
TEC-014	Non-Functional	System	Communication Failure with External Aggregate (Hot-Standby)	CSCC should manage master/hot-standby mode of applications, see TEC-001.									
TEC-015	Non-Functional	System	Communication Failure with External Aggregate (Hot-Standby)	When having hot-standby configuration, item functions (SW part) can be reconfigured to performing a switch from a CSCC hosting a master instance to CSCC hosting a hot-standby instance, see TEC-001.									
TEC-016	Non-Functional	Software	Installation of New Component	An application on the central ICT computing core CCC, which has the clearance to implement new software, has to face common security requirements (authentication, integrity, ...) to avoid installations, e.g., of illegal / harmful software or software that is installed by third parties to without the awareness of the owner.					20.08.2014		New	1	
TEC-017	Non-Functional	Software	Installation of New Component	The safe adaptation core SAPC is properly configured and initiated on all platforms (RACE, TMDP) after installing new components The CCC supports the installation of new applications. The installation-proocess that will be initiated by maintenance personnel ensures the proper operation of the CCC with regard to a given set of functional and non-functional requirements.									
TEC-018	Non-Functional	Software	Update of Function (Update)	Application on the central ICT computing core CCC which has the clearance to install new software. Such application has to face common security requirements (authentication, integrity, ...) to avoid installations of illegal or harmful software, or software that is installed by third parties to without the awareness of the owner, see TEC-016.					20.08.2014		New	1	
TEC-019	Non-Functional	Software	Update of Function (Update)	The safe adaptation core SAPC is properly configured and initiated on all platforms (RACE, TMDP) after installing new components, see TEC-017.					20.08.2014		New	1	
TEC-019-01	Non-Functional	Software	Update of Function (Update)	The CCC supports the extension of ist configuration and the inclusion of an additional aggregate (sensor-actuator-unit) to the ICT system.									
TEC-020	Non-Functional	System	Degradation of Steer-By-Wire Application (Internal)	Overall time response should be less than 50ms (fault detection: 10ms; passivation: 10ms), see TEC-001.					20.08.2014		New	1	
TEC-021	Non-Functional	System	Degradation of Steer-By-Wire Application (Internal)	CSCC should register hardware partition status, see TEC-003.									
TEC-022	Non-Functional	System	Degradation of Steer-By-Wire Application (Internal)	If a degraded mode function is available and hardware partitions are available, when an item function SW failure is detected the function should be activated in a degraded mode instead of failing entirely The CCC shall support graceful degradation on application and on CSCC-level. Gracefull degradation can be represented using a scalar value.					20.08.2014		New	1	
TEC-023	Non-Functional	System	Adaptation for Range Extension (Energy Efficiency)	CSCC should register energy efficiency level for all applications The CCC supports energy consumption optimisation with regard to given constraints and the given route.					20.08.2014		New	1	
TEC-024	Non-Functional	System	Adaptation for Range Extension (Energy Efficiency)	If several item functions (SW part) have the same priority, they should be reconfigured according to energy efficiency optimisation criteria.					20.08.2014		New	1	
TEC-025	Non-Functional	System	Adaptation for Range Extension (Energy Efficiency)	If several item functions (SW part) have the same ASIL, they should be reconfigured according to energy efficiency optimisation criteria. 12.09.2014, Armbruster, SIE: See TEC-024.					20.08.2014		New	1	
DUR-001	Functional	System	Minimum Computing power requirements (CSCC); compatabilities and the extendabilityies of the Switches and Gateways	Need to have some overspec (i.e. 50%) to cover the foreseen circumstances and have wide compatibilities with the current network and data communication standards.	extra cost in Safeadapt Platform core configuration in comparison of another state of the art approachings: cost versus functionalities								
DUR-002	Non-Functional	System	Use Case reliablity evualation	Registration of the frequency of the passivation (adaptation) of a use case.	with failure injection to show the frequency the adaptation	To prove the saving of the material cost and thus the weight as well. To prove the detection rate and to prove the enhancement of reliability of the use case by using the safadapt core; for maintenance purpose to know the failure rate.							
DUR-003	Non-Functional	Process	Adaptation paths	Visualisation of adaptation path at final prototype demonstration.	By visualisation with LEDs	Show the adaptation visually at demonstration.							
DUR-004	Non-Functional	Efficiency	Energy saving at low battery level < 35% to extend the remaining range	Registration of the energy consumption and the priority of an application.	longer range with shutting down of some applications	Eenergy efficiently driving in order to reach home or the next charging station.							
DUR-005	Non-Functional	System	Maintenace mode: failure analysis	To view a list of history of errors.		Check with the frequent users of the EV.							
TTT-001	Functional	System	TTEthernet back-bone architecture	The set-up to interconnect the TMDP and teh RACE platforms shall be TTEthernet.	analysis	Ethernet interfaces are easy and cheap to implement. Standard Ethernet trafic may be difficult to relay on in highly safety-relevant applications where delays and jitter must be known. Deterministic behaviour is also considdered of value in the reconfiguration process the back-bone archutecture shall be capable of. TTEthernet unites all these features.			20.10.2014		New		
TTT-002	Functional	System	Back-bone data communication	The backbone data communication shall allow standard Ethernet and deterministic, Time-Triggered Ethernet data traffic	analysis	Using standard Ethernet interfaces is feasible and in case of highly safety-related traffic needed for control or reconfiguration functions, determinism and hard real time features are required.			20.10.2014		New		
TTT-003	Functional	Hardware	Switch Form Factor for implementation on the tMDP platform	The TTEthernet Switch used on the TMDP Platform shall have standard PMC card form factor.	analysis	The TMDP platform foresees a PMC interface to mount the TTEthernet PMC card.			20.10.2014		New		
TTT-004	Functional	Hardware	Switch to connect TMDP and RACE platforms exterenal of TMDP platform	The TMDP external switch shall be implemented in an industrial standard PC.	analysis	Standard industrial PC can easily be integated into the Platforms, they also offer the opportunity to host a PCIe switch card.			21.10.2014		New		
TTT-005	Functional	System	Data transmission speed of the backbone	The data rate shall be 100Mbit/s.	analysis	Standard TTEthernet hardware equipment is availalbel for data rates of 100 Mbit/s and 1 Gbit/s. The TMDP & RACE platform CPUs support 100 Mbit/s, the switch only will need to run at 100 Mbit/s data rate.			22.10.2014		New		
TTT-006	Functional	System	TTEthernet End Systems	TTEthernet software based endsystems shall be operational on the TMDP and RACE platforms in order to suitably connect the two platforms by TTEthernet traffic.	analysis	No dedicated extra hardware needs to be installed on the TMDP and RACE platforms as an end system. CPU capacity avialable will be usad only.			22.10.2014		New		
TTT-007	Functional	System	Operating System TMDP Platform and TTEthenet	The TMDP Platform will use PXROS. The backbone data communication shall be capable of using the PXROS operating system.	analysis	No extra OS shall be required to use the TTEthernet backbone data communication.			22.10.2014		New		
TTT-008	Functional	System	Operating System RACE Platform and TTEthenet	The RACE Platform will use PikeOS. The backbone data communication hall be capable of using the PikeOS operating system.	analysis	No extra OS shall be required to use the TTEthernet backbone data communication.			22.10.2014		New		
TTT-009	Functional	System	Reconfiguration	The system using TTEthernet backbone architecture shall allow remote reconfiguration by loading gracefully degraded version of the related function to a platform different to the one hosting the non gracefully degraded version. The gracefully degraded function shall be taken from a repository hosted on an ECU independent from the 2 platforms (RACE, TMDP).	analysis	By using such approach the need to install hot redundant hardware shall be reduced satisfying the same safety standards compared to redundancy installed. The idea is to use ECUs which can be used by deleting other non safety critical function in such emergency situationssituation where a safety-relevant function fails.			22.10.2014		New		
TTT-010	Functional	System	Failure detection	The TTETHernet platform shall support remote failure detection.	analysis	In case remote automatic reconfiguration shall be conducte a system is required to detect a failure of a safty-related function on ist own in order to make the decision to reconfigure automatically.			22.10.2014		New		

TTT-011	Functional	System	Sensors and actuators	Sensors and actuators shall be connected to the TTEThernet switches by means of a dedicted ECU.	analysis	In order to safe funding budget, the sensors and actuators shall be connected to a dedicated ECU that can be connected to ether of the platforms just in order to demonstarte reconfiguration capability. In safety-relevant applications a dedicated redundancy concept needs to implemented including sensors and actuators.			22.10.2014		New		
TTT-012	Functional	System	Dual Switch Concept	The connection between the TMDP and the RACE platforms shall make use of a redundant architecture concerning switches in order to also demonstrate the capapbilities of a switched network in case of failure and switching requirement between redundantly built system parts.	analysis	In case of a failure of one switch the other one needs to be able to connect the different platforms reliably in order to tolerant one switch failing.			22.10.2014		New		