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 D2.3

Requirements for the Design Process and Tools for Safe Adaptation

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 Design Process and Tools for Safe Adaptation
Creator	Andreas Eckel
Description	A detailed collection of requirements from a tool perspective and a description of the governing requirements engineering process
Publisher	TTTech Computertechnik AG
Contributors	CEA: Ansgar Rademacher Delphi: Timo Feismann Duracar: Ken Lam Fraunhofer: Annette Paulic, Dulcineia Penha, Philipp Schleiss Ficosa: Daniel Bande, Andrea Saccagno 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

List of Tables

Executive Summary

This document summarises the requirements for the development processes and tools for the safe adaptation approach followed in the course of the SafeAdapt project.

Please note that this document is dedicated to the requirements of the SafeAdapt tools and processes and only contains a limited description besides the requirements tables.

More detailed information on the SafeAdapt processes and tools can be found in D4.2.

1 About this document

This document contains the requirements for the SafeAdapt project with respect to the design process and the tools within SafeAdapt.

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 optimising requirements communication, collaboration and verification.

The DOORS software for requirements processing supports:

1. Requirements Management in a centralized 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 sheets 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”) – optional tool short name – 3 digit number XXX: <Company short name-Tool short name-XXX> thus resulting in an identifier for a requirement for example like: “*TTT-001*” (first requirement by TTTech) or “*TEC-DYN-001*” (first requirement by Tecnalia concerning toll Dynacar).

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

The Excel sheet then identifies the following data per requirement:

a) Column A: Requirement Identifier
b) Column B: Category (functional/non-functional, could be extended if needed)
c) Column C: Sub Category (Efficiency/Hardware/Process/Software/System/Tools)
d) Column D: Short Description
e) Column E: Description
f) Column F: Verification Method
g) Column G: Rationale
h) Column H: Dependencies
i) Column I: Conflicts
j) Column J: Date (of issue)
k) Column K: Supporting material
l) Column L: Object Status (changed/new/, could be extended if needed)
m) Column M: Object Version
n) Column N: Review

2 SafeAdapt Design Process

SafeAdapt provides tool support and a methodology to ensure that innovative architecture solutions are equally supported in the design process. The SafeAdapt tool chain includes modelling, design and validation support. This tool uses a model-based design flow, which is complemented by pre-existing AUTOSAR tool chains, to design adaptivity. Moreover, the SafeAdapt approach enables early verification and validation of the systems non-functional requirements such as adaptability.

In brief, the SafeAdapt Tool Chain is composed of the following tools presented in alphabetical order:

Tool	Purpose
Arctic Studio (ARCCORE)	AutoSAR modelling & code generation.
composeR (SIE)	composeR is a safety analysis tool compliant to FTA/FMEA analyses as defined by various standards such as IEC61508.
Dynacar (TEC)	Help during SW and HW testing phase. Configurable vehicle model running in a real-time system. Models from third parts (Simulink, Dymola) can be integrated on the same platform.
ERNEST (ESK)	Verification and validation of the timing behaviour of networked embedded systems at early design stages of the system.
FMEDAexpress (SIE)	FMEDAexpress is a safety analysis tool for FMEDA analysis according to IEC61508 or ISO26262.
Papyrus (CEA)	Papyrus is a general purpose UML modelling tool that supports SysML (including SysML specific diagrams), MARTE and EAST-ADL profiles. Moreover, it offers several possibilities to customize the user interface.
Prossurance (TEC)	Safety assurance management system. It supports compliance assessment and certification of safety-critical products. Construction of safety cases.

Tool	Purpose
Qompass (CEA)	Qompass is a design tool for model transformation and code generation. Qompass helps to deploy component-based systems taking into account SW and HW architecture. The tool has a support for realizing arbitrary interactions between software components. Qompass also supports a separation of concerns by enabling containers that embed the original component and intercept its communication with the environment as well as offering additional service.
TTEthernet-Tools (TTT)	Generating a valid network configuration for end systems and switches for time-triggered, rate-constrained and best-effort Ethernet traffic.
UNISIM-VP (CEA)	UNISIM-VP is a cross-platform open source simulation environment. Its purpose is to be used during co-design, integration and validation of hardware/software systems. The simulation environment comprises a set of tools and services such as program loaders, OS ABI translators, instrumentation and graphical debugger.
XMT (SIE)	Model oriented system design.

Table 1 SafeAdapt Tools

3 SafeAdapt Tools

Within the following a short overview on the used tools is provided for completeness. A more detailed description is provided in D4.2.

3.1 The following tools are respected:

1. Arctic Studio
2. composeR
3. Dynacar
4. ERNEST
5. FMEDAexpress
6. Papyrus
7. Prossurance
8. Qompass
9. TTEthernet Tools
10. UNISIM-VP
11. XMT

3.2 Arctic Studio

The Arctic Studio tool chain provides a complete software development environment for automotive embedded software solutions based on the open industry-leading standard AUTOSAR. The tool chain supports all stages of an automotive ICT project and provides tools for different types of tasks, such as application development, embedded platform development, and system integration.

As an input the Arctic Studio tool requires AUTOSAR configuration files that where either imported using arxml files or created inside of Arctic Studio. Furthermore, Arctic Studio supports the import of "Software Component Description" files (ARText) and provides importers for communication matrices in form of AUTOSAR ECU extract and CanDB files. The end result of the Arctic Studio tool chain is a configuration dependent RTE in form of C-code and a compiled, linked, and executable binary image (ELF) for the target platform.

Furthermore the tool features:

- Full access to AUTOSAR arxml files through the Artop open source project
- Wizards for creating AUTOSAR projects and AUTOSAR files
- Full support for handling configurations split into multiple files
- AUTOSAR viewer with possibility to walk through the AUTOSAR configuration in a tree view

- Support of AUTOSAR standard version 4.0.2, 4.0.3 and 4.1.1

3.3 composeR

composeR is a safety analysis tool compliant to FTA/FMEA analyses as defined by various standards such as IEC61508. As already described in the previous section, classical safety analyses like FTA and FMEA do not aim at adaptations during runtime. Both, top-down approaches like FTA and bottom up approaches like FMEA are still required for a sound safety analysis. A generic failure type system and a component-based approach facilitate a safety framework for verification at runtime. A component integrated safety model allows analysis at design time. An integrated safety analysis leverages the benefits of model-based development for certification efforts and the composeR tool allows a cost-efficient safety analysis by systematic reuse of safety analysis models. Furthermore, it allows making the right global and detailed design decisions in early phases and break with delaying try and error project cultures. It supports agile approaches by immediate reliability and safety analysis of detailed designs. By that, developers gain higher confidence in system test by verifying tests against safety models. We can reduce expensive diagnosis effort by using the safety analysis models for root cause analysis aiming at a shorter time-to-market.

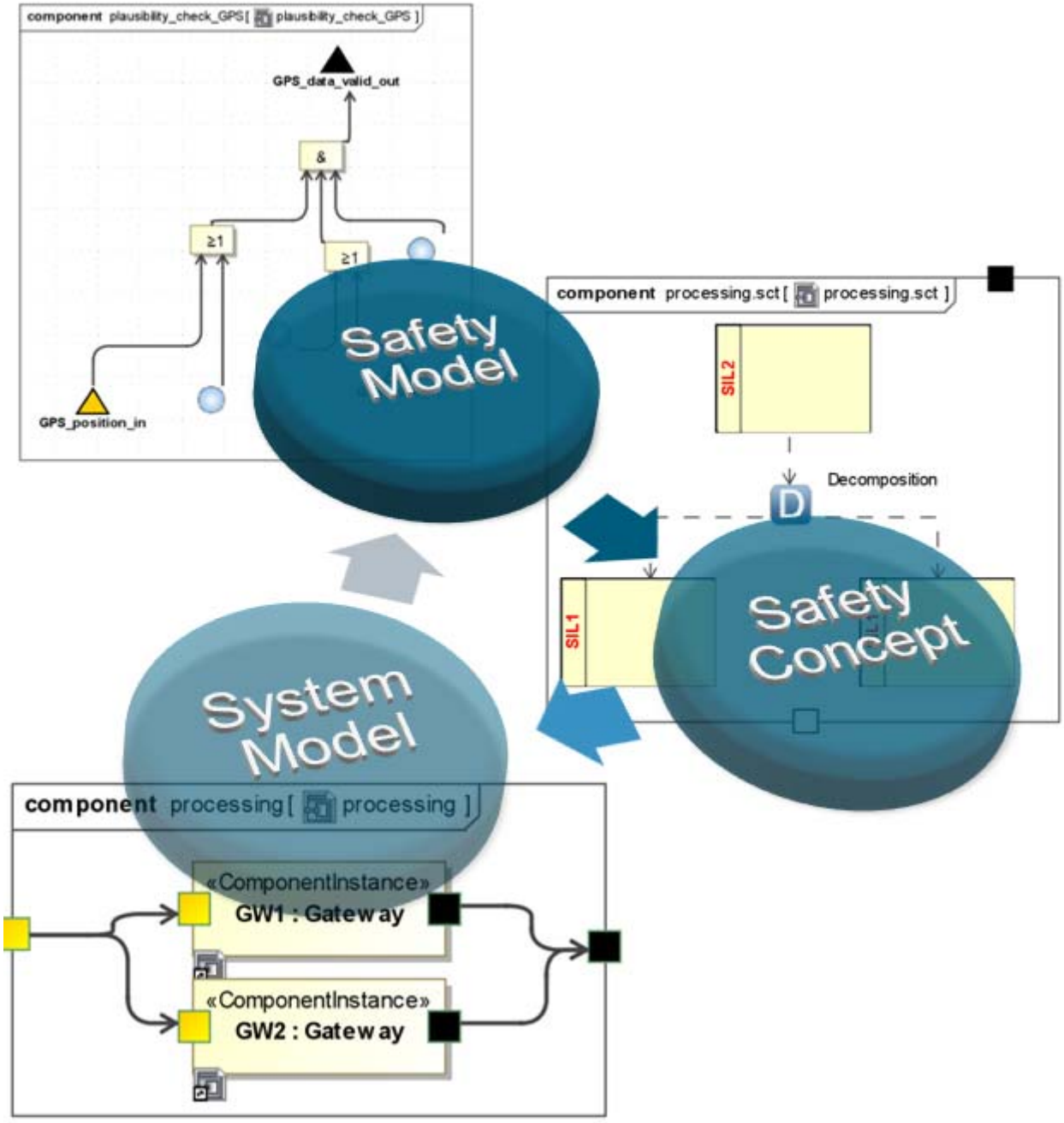

Figure 1: composeR model structure

Figure 1 shows the overall structure of the composeR tool. Here, Component Integrated Fault Trees are combined with the system design. In this tool, currently the SysML methodology is used to model the inner system design. Additionally to the safety and system views, composeR provides a third view, the safety concept, to allow argumentation structures using an extension of the goal structuring notation. In the project, the tool is extended by additional functionality aiming at an automated certification. It is assumed, that every step towards an automated certification is also a step towards a certification at runtime. The goal of a certification at runtime is achieved, if the automation is fast enough to allow a certification decision in real-time.

3.4 Dynacar Tool

Dynacar is a road vehicle model fully developed by TECNALIA in LabVIEW RT. It can be used through the whole powertrain design process, allowing the rapid prototyping, implementation and real-time testing of electronic control units and powertrain components. When combined with Veristand™, custom control algorithms and simulation models generated with other languages can be easily integrated into the vehicle model. Dynacar RT allows powertrain engineers to quickly generate their own vehicle model, using the graphic user interface with an advanced virtual environment, and to supervise the real time testing with all the capabilities of Veristand™.

Next, the main characteristics of the tool are described in more detail:

1. Fully **configurable Real Time Vehicle model embedded in a PXI controller**, valid for **conceptual or model-based design of vehicles.**
2. Capable of **integrating customer real time models and controls**, working as a **"Virtual Rolling Chassis"** concept (or test mule virtual car). This can be carried out in the same PXI controller, or implemented in a series of networked and Synchronized PXI´s, depending on the computational requirements of the models.
3. Designed for **model-based development and testing equipment** in applications such as ICE and hybrid powertrains dynamometers, eMotor powertrain dynamometers, transmission dynamometers, battery testing benches and fuel cell testing benches.
4. Development of control algorithms, starting with Model in the Loop (MiL) down to HiL **Hardware in the Loop (using ECUS),** for vehicle controllers related to **Chassis and Powertrain domains**, such as anti-lock brakes, traction control algorithms, electronic stability control, regenerative braking for electric vehicles, energy management systems for hybrid vehicles and others.
5. Same model applicable throughout all design stages in the "V" development diagram.
6. Capable of either ***Driver in the Loop (DiL)*** or autonomous cycles (path follower) from "out of the box".
7. Validated with different experimental and analytical evidences.
8. Good compromise between number of parameters and accuracy of the results.
9. Basic set of vehicle 3D skins, ranging from a segment to large buses, and different circuits, from proving grounds to racing and road circuits. Vehicle 3D model and specific circuits can be included on demand

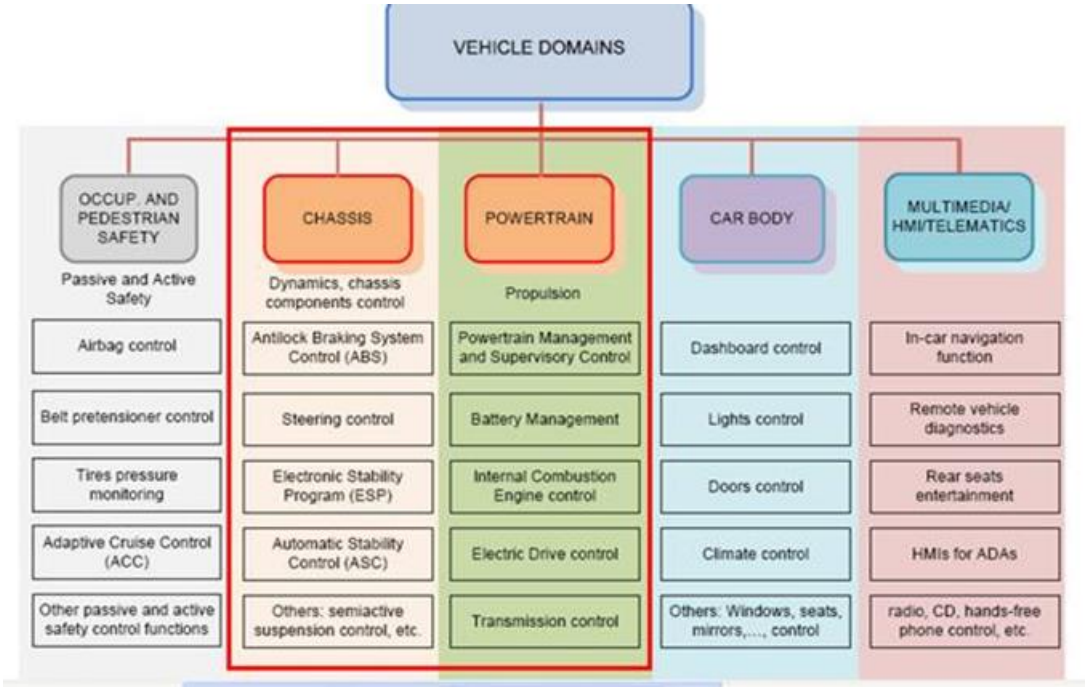

Figure 2: Vehicle Domains

For more information on Dynacar tool, refer to deliverable D2.1 "*Definition of Use Cases and Scenarios for Safe Adaptation*".

3.5 ERNEST

ERNEST is a framework for the **E**a**R**ly verification and validation of **N**etworked **E**mbedded **S**ys**T**ems. The open platform supports the early analysis of component-based software systems with a focus on automotive networked embedded systems. The complexity of networked embedded systems is continuously increasing, because the requirements and the set of provided functionalities of these systems are growing, as well. Early verification of embedded systems is necessary to prevent failures and to save costs during the design. It is insufficient to solely consider functional properties of the software for networked embedded systems to satisfy the quality requirements in most of their application domains. An early verification of non-functional properties is inevitable to produce robust software-based embedded systems in a cost-efficient way.

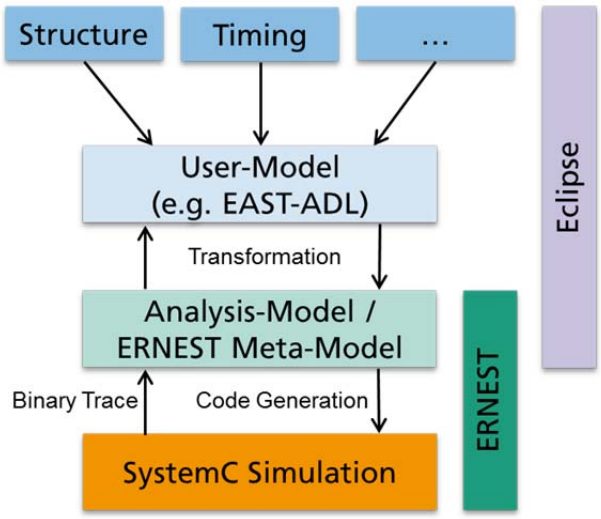

Figure 3: ERNEST Framework user-model integration

ERNEST provides flexible mechanisms to verify non-functional properties in early design stages, based on a specific simulation framework that is written in SystemC. ERNEST can be integrated easily into a model-based design flow and is based on the open-source development platform Eclipse. It can easily be enhanced by various analysis techniques, as ERNEST is built as an extensible tool platform for verifying non-functional properties. The modeled hardware, software and communication behavior is simulated as accurate as needed to analyze and verify a networked embedded system. The results of these simulations can be used in third-party tools or re-integrated into the initial model. This enables an iterative model-driven development process exploiting early prototyping.

3.6 FMEDAexpress

FMEDAexpress is a safety analysis tool for FMEDA analysis according to IEC61508 or ISO26262. Classic safety analyses like FTA and FMEA do not aim at adaptations during runtime. Both, top-down approaches like FTA and bottom up approaches like FMEA are still required for a sound safety analysis. A generic failure type system and a component-based approach facilitate a safety framework for verification at runtime. A component integrated safety model allows analysis at design time. FMEDAexpress provides basic functionality for local and generic effects and is flexible and extendable. Figure 4 shows the central interface of this tool.

Characteristics:

1. Handles .xml input and output.
2. .xlst file allows customized view.
3. SQL Database makes it easy to extend, e.g. to store additional information or to adapt different analyses.
4. .NET 4 Framework application written in C Sharp.
5. Currently provides full FMEDA analyses with quantifications according to IEC61508.
6. Handles local and generic effects.

7. Implements a component-based approach for hardware components.
8. Implements routines that solve specific problems during FMEDA analysis which are under restrictions by SIEMENS.

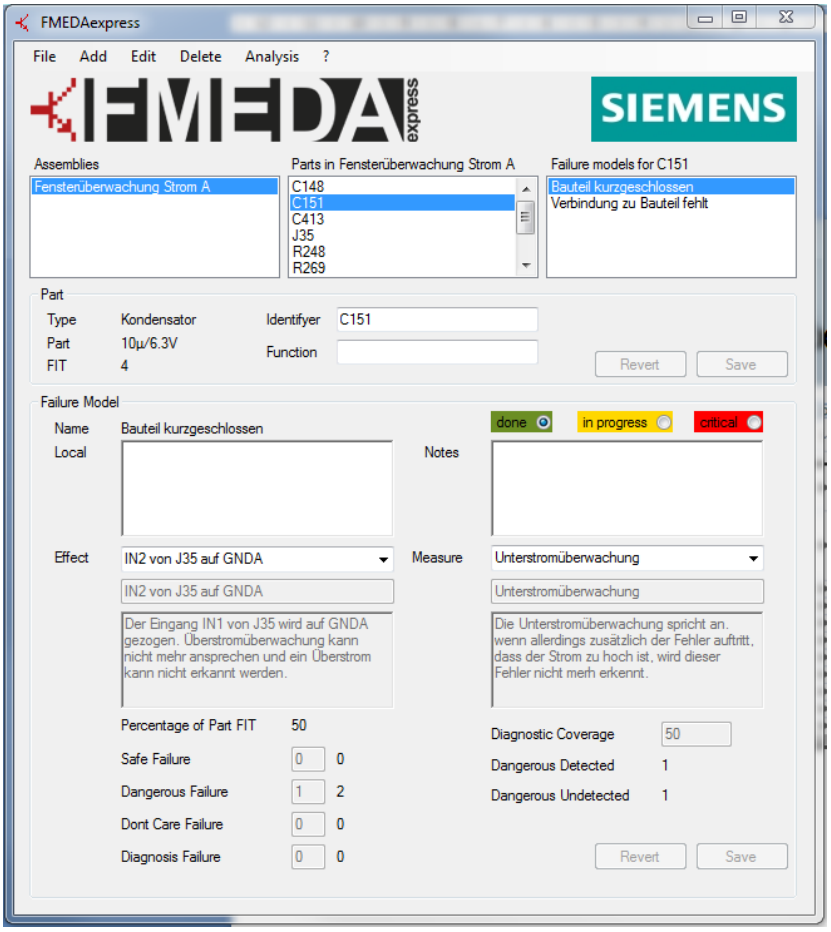

Figure 4: Screenshot of the central FMEDAexpress user interface

Using a tool supported model-based approach overcomes the drawbacks of an Excel-based analysis. Adding a new evaluation method to an Excel sheet is a time intensive task and adding automations to existing analyses (reuse) is error prone. Furthermore, the visualization in Excel is constrained to one view. With .xml, multiple views can coexist at the same time.

Adding a new failure mode can result in complex inconsistencies in an Excel-based FMEDA, e.g. if the analysis is comparatively large and has to be reviewed entirely. Reoccurring effects or diagnostic measures can result in a complex network of links in your Excel-based FMEDA. Using Database structures eases the process. Due to the database structure FMEDAexpress overcomes those drawbacks and fault trees can be generated out of FMEDA analyses. Figure 5 shows the meta-model that allows the previously described benefits over an Excel-based solution.

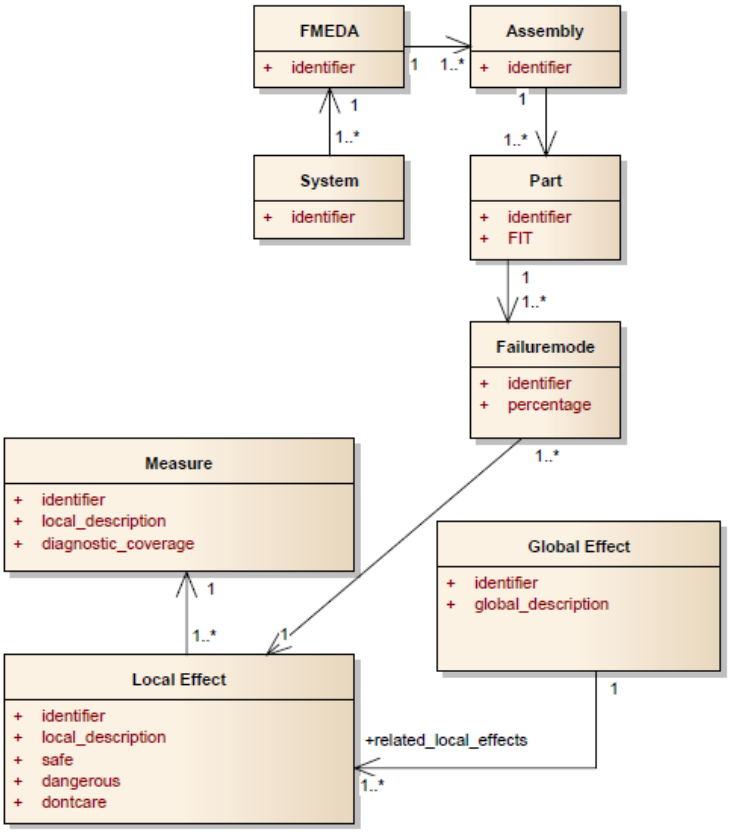

Figure 5: Meta-model of FMEDAexpress

3.7 Papyrus

Papyrus is a general purpose UML modelling tool. It consists of a set of Eclipse plug-ins. It supports the UML extension mechanisms in form of profiles and offers several possibilities to customize the user interface. In particular, it supports the profiles SysML (including SysML specific diagrams), MARTE and EAST-ADL.

Papyrus is an official Eclipse project and is available within the Eclipse modelling bundle. More information can be found on eclipse.org/papyrus. It is the base for the modelling tool Qompass.

3.8 Prossurance Tool

Prossurance is a product and process assurance management system to support the compliance assessment and certification of safety-critical systems in sectors such as aerospace, railway and automotive.

Prossurance helps to create a transparent view of the process and product quality against a set of harmonized compliance requirements derived from standards and regulations.

Through the use of knowledge-based systems, quantitative methods and modular reuse techniques, Prossurance reduces compliance management and (re-)certification costs.

Given the numerous and complex regulatory challenges, organizations working in the development and operation of safety-critical systems are always seeking ways to improve compliance

management and certification processes. This is crucial to reduce product costs, avoid legal issues, penalties, incoming reductions and, more important, the exclusion from key markets and regions.

Comply with standards/regulations requires more than simply applying them “as they come”. Differing interpretations of regulations, from different perspectives (regulatory entities, manufacturers, suppliers, assessors) create difficulties in the context of specific projects.

Regulations come from many sources, often heterogeneous in detail and vocabulary. Furthermore, major problems arise when evolutions to a safety-critical system entail reconstruction of certification arguments and evidence, or when trying to reuse products from one application domain in another, because they are constrained by different standards. The full safety assurance and certification process is applied as for a new product, thus reducing the return on investment of such a reuse decision.

Following modules compose the Prossurance tool:

- Knowledge Management: Capture information from reference framework. Specify company specific reference framework. Map knowledge from different reference frameworks.
- Assurance Project Management: Create Safety Assurance project. Define Safety Assurance project baseline. Define access permission for users.
- Evidence Management: Determine the evidence to provide. Collect and characterise information about evidence items. Specify traceability between evidence items. Perform evidence change impact analysis.
- Argumentation Management: Define modular assurance structure. Develop claims and links to evidence. Specify argumentation module assumptions. Validate argumentation module assumptions.
- Process Management: Check process compliance against reference framework measure and estimate safety metrics. Specify traceability between process items. Perform process change impact analysis.

Prossurance will be the basis in SafeAdapt to define a vertical solution for the automotive sector. Its Knowledge Management module will include the reference framework for the automotive industry according to ISO 26262.

3.9 Qompass

Qompass is a design tool for model transformation and code generation. The Qompass tool helps designers to deploy component-based systems. This means that designers take into account not only the SW architecture but also the HW architecture and allocation of SW to HW. The tool has a support for realizing arbitrary interactions between software components. These interactions are defined in a model library. Thus, it is possible to target multiple middleware technologies, e.g. interaction styles used in automotive domain, e.g. communication via the AUTOSAR virtual function bus (though not realized yet).

3.10 TTEthernet Tools

TTEthernet (SAE AS6802)[2] is a scalable, open real-time Ethernet platform used for safety-related applications primarily in transportation industries and industrial automation. TTEthernet extends classic Ethernet functionalities to provide more flexibility, modularity and scalability in Ethernet-based systems. It is compatible to IEEE 802.3 Ethernet and integrates transparently with Ethernet network components.

TTEthernet based networks enable the seamless communication of all kinds of applications via Ethernet. Conventional PCs, web and office devices, multimedia systems, real-time systems and safety-critical systems are to use the same network. One single network that is completely compatible with the IEEE Ethernet 802.3 standards is suited for data transmission among different applications with various requirements, e.g. satisfying different criticality requirements and fail-safe or even fail-operational behavior. **Fehler! Verweisquelle konnte nicht gefunden werden.** gives an overview of the different communication types of TTEthernet. For SafeAdapt, the time-triggered traffic in particular is of most relevance since it best supports the requirements of the automotive domain.

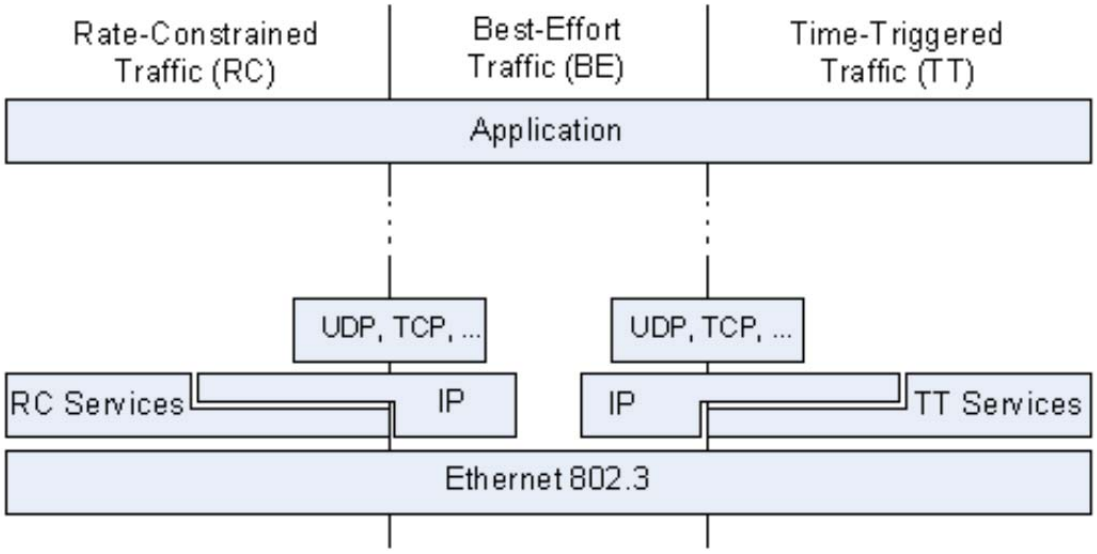

Figure 6: TTEthernet traffic types and relation to other protocols

Time-triggered (TT) traffic has two important pre- requisites: the need for a global notion of time in the network and the availability of schedules that organize the communication in the time domain, i.e. providing time partitioning on the network. For these reasons, switches in TTEthernet take over the central role of organizing the data communication. TT messages are routed in the switch according to a predefined schedule with as little delay as possible. Precise planning at the time of system design precludes resource conflicts at runtime. TT messages have the highest priority level. If the planned transmission time of one of these messages arrives, this message is immediately transmitted. Due to the predefined transmission of the message the switch ensures that the medium is free at the time of transmission and delays are precluded.

[2] See SAE Standard AS6802: *Time-Triggered Ethernet*, http://standards.sae.org/as6802/

Schedules in TTEthernet are generated using a dedicated tool chain, where each tool solves a particular task of the configuration. The overall TTEthernet configuration tool chain is depicted in Figure 8. It consists of the following main parts:

- **TTEPlan:** TTEPlan is the TTEthernet network planning tool. Based on input provided to the tool, TTEPlan creates the whole network configuration databases.
- **TTEBuild:** TTEBuild allows converting XML-based device configuration database files into binary configuration images required by the TTE Switches and the TTE End Systems.
- **TTELoad:** TTELoad is an application suitable to configure a TTE Switch based on TTEthernet switch IP that also supports bootstrap configurations of TTE Switches.
- **TTEView:** This TTEthernet frame dissector for Wireshark[3] 1.x is a plug-in to Wireshark which supports the recording and analysis of over 300 Ethernet and internet protocols including TTEthernet.

An overview of this tool chain showing input and output files is shown in Figure 7 and Figure 8. TTEPlan can be used to configure a network from scratch, or to migrate an existing configuration to a network description file. The configuration output of the tool chain is a schedule that can be downloaded or otherwise communicated to the TTEthernet network components. It defines the time-slots during which communication on the network will occur including a separation along the different communication types co-existing in the network. An example configuration output is shown in Figure 7.

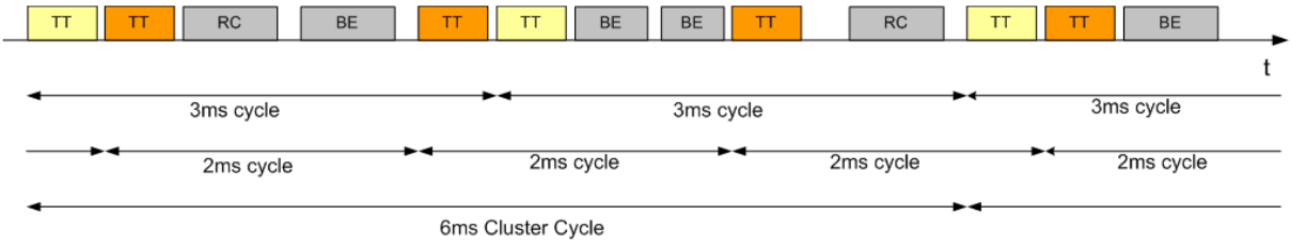

Figure 7: Example schedule as output of the TTE tool chain

[3] http://www.wireshark.org

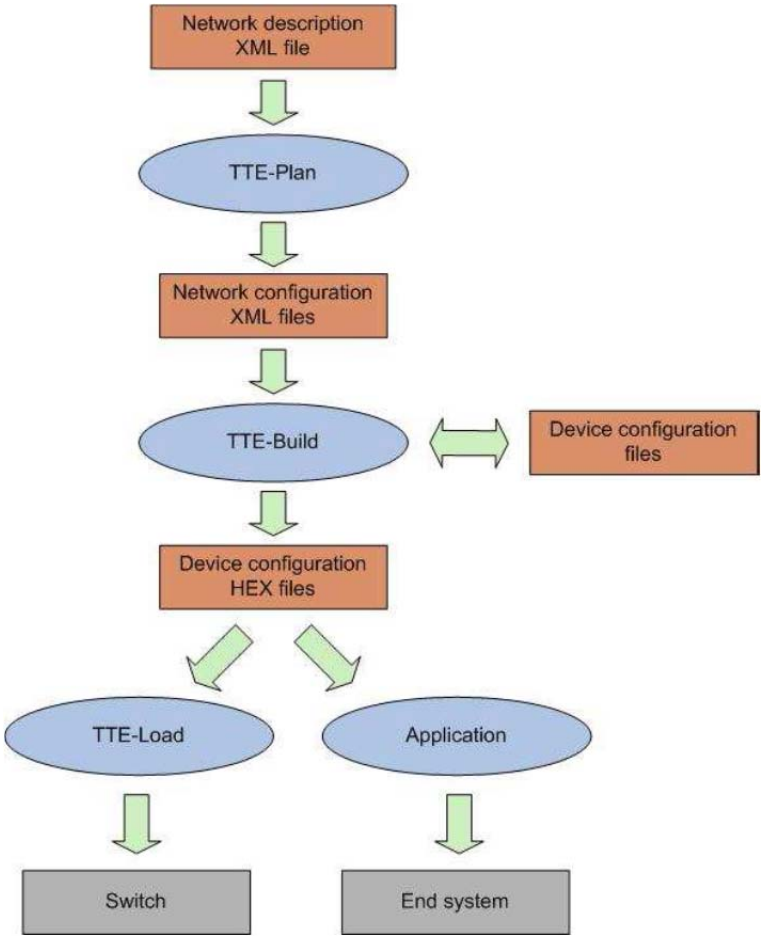

Figure 8: TTEthernet configuration tool chain

3.11 UNISIM VP Simulator

UNISIM-VP is a cross-platform open source simulation environment based on industry standard SystemC. Its purpose is to be used during co-design, integration and validation of hardware/software systems.

The simulation environment comprises a set of tools and services such as program loaders, OS ABI translators, instrumentation and graphical debugger. Supported hosts are Windows, Linux and Mac OS X.

The UNISIM-VP simulation environment will be used to emulate the targeted hardware platform and hence to execute the embedded software.

3.12 XMT

In the past many embedded systems have been implemented as standalone systems. They have been developed in isolation and only provide well-defined, but quite static interfaces to other systems. However, recently the trend to integrate a larger number of embedded systems into larger “systems of systems” has got momentum. Examples are larger networks of wireless sensors or, as in SafeAdapt, networks of ECUs.

The XMT modeling tool (http://www.fortiss.org/forschung/projekte/chromosome/) has been developed by fortiss as a modeling frontend for networked embedded systems based on the CHROMOSOME middleware. It has been used in RACE as a tool to model all vehicle data (i.e. all information to be exchanged between sensors, functions and actors), to model functions and their interdependencies as well as to model the physical setup of a system. The model generated by XMT tool is used to generate various configuration information like data structures for network communication, configuration of RTE models and deployment descriptors. Figure 9 and Figure 10 show an example for the representation of data object classes and components in XMT tool. In SafeAdapt the goal is to use the configuration information from XMT as a basis for the automated assessments.

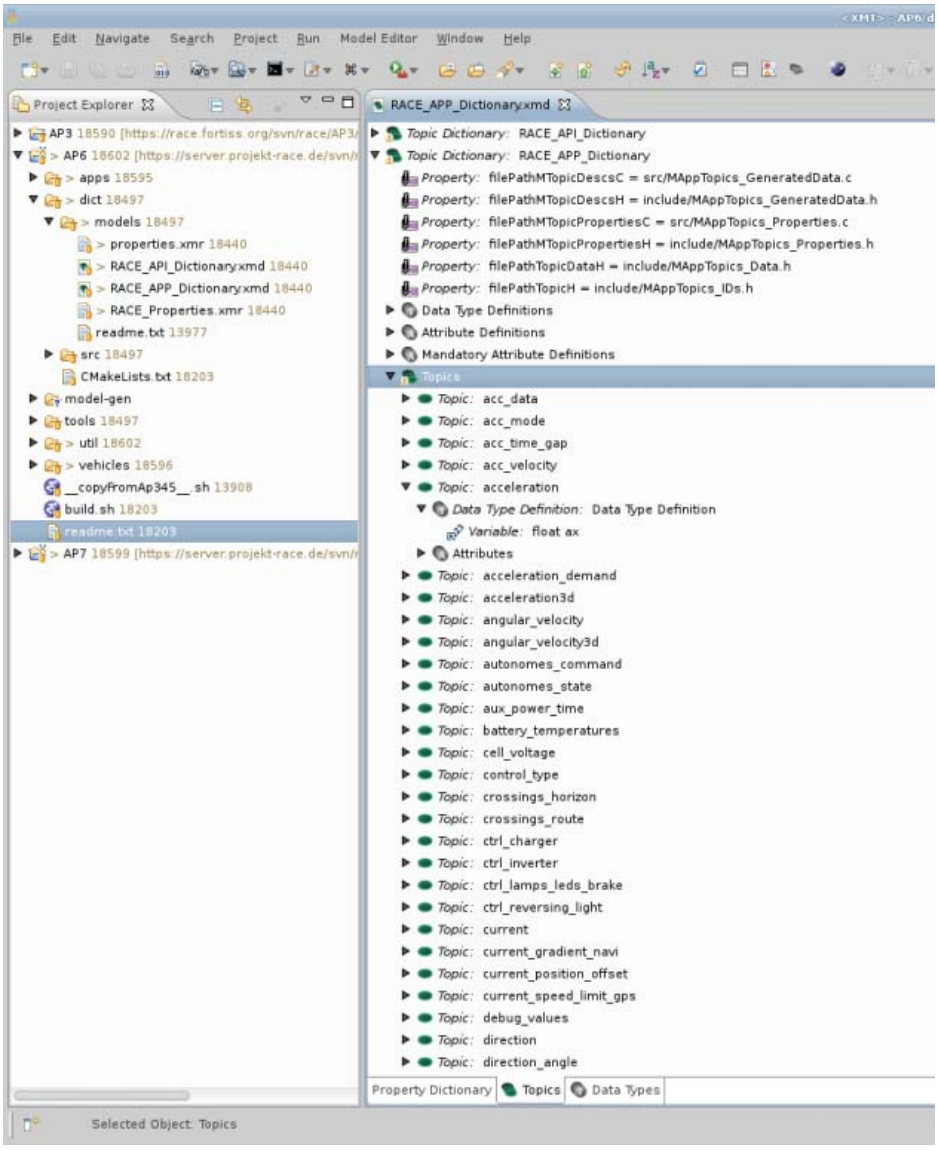

Figure 9: Example for the representation of data object classes (topics)

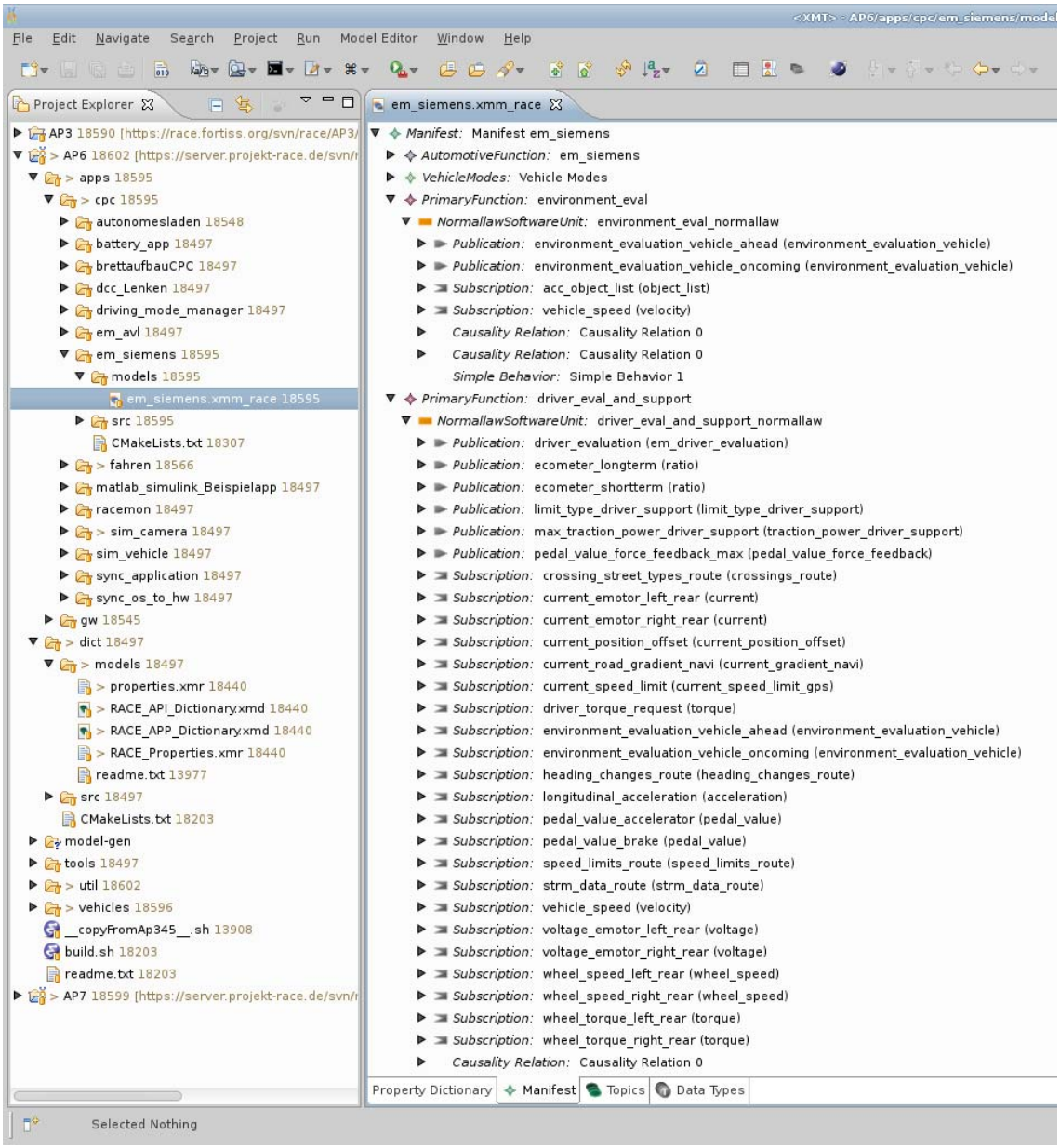

Figure 10: Representation of components (functions, software units, inputs, outputs)

Bibliography

SAE Standard AS6802: *Time-Triggered Ethernet*, http://standards.sae.org/as6802/

List of Abbreviations

Abbreviation	Definition
DiL	Driver in the Loop
DOORS	Dynamic Object-Oriented Requirements System
ERNEST	EaRly verification and validation of Networked Embedded SysTems
RACE	Robust Reliable Automotive Computing Environment
SAE	Society of Automotive Engineers
SafeAdapt	Safe Adaptive Software for Fully Electric Vehicles
TMDP	Trusted Multi Domain Platform
TTEthernet	Time-Triggered Ethernet (SAE standard SAE AS6802)

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 that form part of this document is:

SafeAdapt_D2-3_Requirements.xlsx

Requirement ID: <Participant><Req. ID>	Category <F, 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: <Dep. to other 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>
CEA-001	Functional	System	Modeling of system architecture in EAST-ADL				none						
CEA-002	Non-Functional	System	Modeling of non-functional requirements using UML/MARTE				none						
CEA-003	Functional	System	Modeling of global modes, but respect CEA-004				none	CEA-004					
CEA-004	Non-Functional	Efficiency	Avoid combinatorial explosion	Avoid combinatorial explosion, i.e. avoid enumerating all possible system configurations in detail. (Possible solutions: configuration spaces, hiearchical configurations, etc.)			none						
CEA-005	Non-Functional	Tools	Depending on criticality, we must be able to assure schedulability even during reconfiguration at all times (offline-analysis)	Possible conflict: doing schedulability analysis requires detailed configuration information and verification of ALL possible transition.			none	CEA-004					
CEA-006	None	Process	Respect tool flow picture from D4.2				none			D4.2			
CEA-007	Functional	Process	Link with simulation tools (can either generate code for target execution or for simulation)				none						
CEA-008	Functional	Process	Export from UML/EAST-ADL to artop (AUTOSAR)				none						
ESK-001	Functional	Tools	Timing-Analysis shall support AUTOSAR	The tool for the V&V of the timing behaviour shall support AUTOSAR exchange formats like ARXML.	Test with an Use-Case modelled in AUTOSAR using ARTOP that delivers the models in the ARXML-format.	AUTOSAR is a standard in the automotive domain. The support allows the use of the SafeAdapt approach by several different companies.	none		01.06.2014	AUTOSAR Specification, AUTOSAR Tooling Platform (ARTiP)	New	1	
ESK-002	Functional	Tools	Timing-Analysis shall support EAST-ADL	The tool for the V&V of the timing behaviour shall support system description in EAST-ADL.	Test with an Use-Case modelled in EAST-ATL using Papyrus and additional EAST-ADL profile plug-in.	EAST-ADL is an adequate domain-specific language for automotive that supports the descriptions of timing constraints.	none		01.06.2014	EAST-ADL Specification, EAST-ADL Plug-in provided by CEA	New	1	
ESK-003	Non-Functional	Tools	Ease of integration for timing-analysis	Tool for the Timing-Analysis shall be easily integrated in the overall tool-chain of SafeAdapt.	Integration with other tools in the tool chain.	In order to provide a seamless V&V process the analysis tool needs to support different exchange formats.	ESK-001, ESK-002		01.06.2014	OSGi, Eclipse Plug-in API	New	1	
ESK-004	Functional	Tools	Timing-Analysis shall provide feedback to the input model	The analysis results of the timing-analysis shall be back-propagated into the input model highlighting violations in the timing-behaviour.	Test with timing constraints modelled in an existing use-case.	It eases the verification of the timing-behaviour by visualizing violated constraints.	ESK-002		01.06.2014	TADL Specification	New	1	
ESK-005	Functional	Tools	The SafeAdapt Development & Modelling Process shall support AUTOSAR	The process shall support AUTOSAR components and exchange formats like ARXML.	Test with a AUTOSAR component in ARXML format.	AUTOSAR is a standard in the automotive domain. The support allows software to be reused on different ECUs provided by different companies.	none		01.06.2014	AUTOSAR Specification	New	1	
ESK-006	Functional	Tools	The SafeAdapt Development & Modelling Process shall support dynamic linking	The process shall support individual compiling, linking, and packaging of components.	A component is built without OS code and can be loaded by any safe adapt core at runtime.	As component can be executed on any ECU with SafeAdapt Core, the local storage of all potential application is not feasible due to storage limitations, therefore application must be loaded from remote locations on demand.	none	Does not comply with current AUTOSAR standard	23.06.2014		New	1	
ESK-007	Functional	Tools	The OS source code shall be modifiable	The modules of the operating system can be edited to support features of the SafeAdapt Core.	Change source code and compile it.	The SafeAdapt Core must be in control of low level functions such as scheduling, memory management, and code loading, which are not directly supported by OS APIs.	none		23.06.2014		New	1	
ESK-008	Functional	Tools	The OS shall support loading of closed source AUTOSAR components	The operating system can load binary images of single AUTORSAR compatible applications at runtime.	Load a binary image of a AUTOSAR component.	Suppliers do not want to provide source code, therefore binary images shall be loadable without changes.	none		14.08.2014	AUTOSAR Specification	New	1	
ESK-009	Functional	Tools	The SafeAdapt Development & Modelling Process shall provide a component self-description	Every component provides a formalised description of runtime & adaptation requirements that are needed to calculate a new system state.	Test with a adaption use case.	The SafeAdapt Core must be aware of the requirements of each software component to plan new system state after adaptation.	none	Does not comply with current AUTOSAR standard	14.08.2014		New	1	
TEC-DYN-001	Non-Functional	Hardware	Recommended Dynacar RT Hardware Requirements	The Recommended Dynacar RT Hardware Requirements are: 1) Host PC 2) Visuals PC 3) Ethernet Switch 4) Steering Wheel 5) NI PXI Real Time Target	Visual check	In order to run Dynacar simulaton software the following hardware is needed.	none		26.06.2014		New	1	
TEC-DYN-002	Non-Functional	Hardware	Host PC specifications	CPU: Core i5 or similar Memory: 2 GB minimum for 32 bit and 4 GB minimum for 64 bit version Graphics: Integrated graphic card Storage: The minimum total hard disk space will be 30GB for the OS and the complete software package. OS: Windows 7 32-64 bit OS	Visual check	Host PC minimun specifications to run Dynacar.	TEC-DYN-001		26.06.2014		New	1	
TEC-DYN-003	Non-Functional	Hardware	Visuals PC specifications	CPU: Core i7 or similar Memory: 2 GB minimum for 32 bit and 4 GB minimum for 64 bit version Graphics: Graphic card with minimum NVidia GEFORCE GTX chipset, SLI ready. Two identical graphic cards will with three monitor configurations is recommended for more immersive driving experience, but one monitor configuration can be used with only one graphic card. Microsoft DirectX 11 is required. Storage: The minimum total hard disk space will be 30GB for the OS and the complete software package. OS: Windows 7 32-64 bit OS	Visual check	Visual PC minimun specifications to run Dynacar.	TEC-DYN-001		26.06.2014		New	1	
TEC-DYN-004	Non-Functional	Hardware	Ethernet Switch specifications	A standard gigabit Ethernet switch with at least 3 Ethernet connections.	Visual check	Ethernet switch specifications to run Dynacar.	TEC-DYN-001		26.06.2014		New	1	
TEC-DYN-005	Non-Functional	Hardware	Steering Wheel specifications	Logitech G27 steering wheel is required for driver in the loop simulations.	Visual check	A Steering wheel is needed to drive the virtual vehicle. Supported Wheel is Logitech G27.	TEC-DYN-001		26.06.2014		New	1	
TEC-DYN-006	Non-Functional	Hardware	NI PXI specifications	As the RT Platform a NI PXI Real time controller is required. The minimum hardware configuration is NI PXI 8110 Core 2 Quad 2.26 GHz with Real Time Embedded software, installed in a PXI 1031 4-slot chassis.	Visual check	Minimum requierements for the PXI to be able to run Dynacar.	TEC-DYN-001		26.06.2014		New	1	
TEC-DYN-007	Non-Functional	Hardware	Connections for Hardware in the Loop HIL	The PXI minimum configuration should be upgraded with different I/O communication cards (CAN, LIN, FlexRay, Ethernet, Digital, Analog, etc.), depending of the final user requirements.	Visual check	PXI can be upgraded with different comunication expansion card hardware.	none		26.06.2014		New	1	
TEC-DYN-008	Non-Functional	Hardware	SafeAdapt Core System HIL	The ECUs and gateways hardware from SafeAdapt core must be available as the HIL system to connect with Dynacar.	Visual check	The unit under test must be aballable to be connected with Dynacar in order to create a HIL simulation.	none		26.06.2014		New	1	
TEC-DYN-009	Non-Functional	Software	All Dynacar PC and PXI in a Local IP Address network	Host PC, Visual PC and PXI must have an IP address between the same IP range. By default PXI has 193.0.0.2 so the rest of the PC's must have an IP in the 193.0.0.XXX range.	Compilation	All Dynacar components work together connected via an ethernet conexciotn through the same ip range.	none		26.06.2014		New	1	
TEC-DYN-010	Non-Functional	Software	NI Veristand 2012	Host PC must have a valid licensed Veristand 2012 installation	Compilation	Working license of Veristand 2012 is a must have, since Dynacar works with the Veristand interface as a plugin.	none		26.06.2014		New	1	
TEC-DYN-011	Non-Functional	Software	NI PXI with Veristand 2012 engine installed	The NI PXI real time target must have the Veristand engine and all the required components properly installed.	Compilation	Veristand engine must be installed inside the PXI to run veristand project files.	none		26.06.2014		New	1	
TEC-DYN-012	Non-Functional	Software	SIL Software in the loop dll generation Requirements	To create external model dll files for Software in the loop (SIL), a valid Matlab, Simulink and Realtime Workshop installation is needed.	Compilation	External model dll files compatible with Veristand are created with third party software like Matlab Simulink.	none		26.06.2014		New	1	
TEC-DYN-013	Functional	Tools	Vehicle Parameters to feed Dynacar GUI	Dynacar vehicle configuration GUI must be filled with representative vehicle parameters that define the vehicle under test. (Mass, Inertias, Suspension, Steering, Powertrain, etc.)	Analysis	To simulate a vehicle in Dynacar some vehicle data needs to be available to configure vehicle physics paramenters	none		26.06.2014		New	1	
TEC-DYN-014	Non-Functional	Software	Dynacar RT simulation Time Step	Dynacar RT Vehicle Simulation Runs on a Time Step of 1ms. Every 1ms the physics calculations are executed and new output variables are calculated.	Compilation	Dynacar RT physics calculations are made every 1ms in order to have an acurate real time simulation	none		26.06.2014		New	1	
TEC-DYN-015	Non-Functional	Software	SIL models Time Step	SIL models created with Simulink must have the same or greater time step than Dynacar (1ms or more). If you choose to have higher time step than Dynacar, you must configure decimation in Veristand when importing external dll models.	Compilation	Externals dll models can be configured to run at the same rate as Dynacar or at a lower rate always a multiple of 1 ms.	none		26.06.2014		New	1	
TEC-DYN-016	Non-Functional	Software	Veristand Data Logging in TDMS Format	Veristand can log any variable from Dynacar, external models and HIL systems that is defined inside Veristand Engine. The log is written in TDMS format. The logging rate can be specified in Veristand	Compilation	To log simulation variables the embedded data logger plugin fron NI is used. This plugin writes log files in TDMS format.	none		26.06.2014		New	1	
TEC-DYN-017	Non-Functional	Tools	Dynacar License	Dynacar License is associated with the host id identifier of the Host PC. Dynacar GUI can only be executed in the Host PC where is installed. NI PXI real time target must be permanently connected to the same network as Host PC and simulation will only execute when Dynacar GUI is open in the HOST PC and connected to the PXI.		Valid Dynacar license is needed to run all Dynacar software.	none		26.06.2014		New	1	
TEC-DYN-018	Functional	Tools	Required Dynacar Inputs for HIL	SafeAdapt ECU system must provide at least the following signals to control Dynacar virtual vehicle: -Throttle -Brake -Clutch (if applicable) -Steering wheel Angle	Analysis	Minimum input variables requiered by dynacar to control the virtual vehicle.	none		26.06.2014		New	1	
TEC-DYN-019	Functional	Tools	Required Dynacar Outputs for HIL	Dynacar has available several vehicle variables as outputs. A definition of the output variables that SafeAdapt ECU System must receive from Dynacar virtual vehicle needs to be done.	Analysis	Safe Adapt ECU Platform will need sensor variables from the virtual vehicle. Dynacar has a list of output variables.	none		26.06.2014		New	1	
TEC-DYN-020	Functional	Tools	Generate Required Dynacar Outputs for HIL	Modify Dynacar source code to include variables that are not available in the standard version of Dynacar but are needed by SafeAdapt system.	Analysis	The variables that are not already included in the Dynacar outputs must be generated modifying the Dynacar source code.	none		26.06.2014		New	1	
TEC-DYN-021	Functional	Tools	Virtual Sensors Generation	If needed, generate required virtual sensor with Simulink as SIL or by modifying Dynacar source code.	Analysis	Virtual sensors can be generated if needed by Safe Adapt Core.	none		26.06.2014		New	1	
TEC-DYN-022	Functional	Tools	Dynacar I/O available connection Types	Available type of connection between Dynacar and Safe Adapt hardware are: - CAN Network - Analog/Digital Signals	Analysis	To connect Safe Adapt Core with Dynacar, a CAN network or analog/ digital signals can be used.	none		26.06.2014		New	1	
TEC-DYN-023	Functional	Tools	Dynacar I/O Ethernet connection	If an Ethernet connection with SafeAdapt hardware is a must have, the TTTech communication protocol must be implemented in Dynacar. TTTech will provide the protocol and if it is possible code it in Veristand.	Analysis	If the connection is made via ethernet, the comunication protocol must be implemented in Dyancar.	TEC-DYN-022		26.06.2014		New	1	
TEC-DYN-024	Functional	Tools	Use Case recreation in Virtual Dynacar World	A detailed description of the use case is a must have. Dynacar source code will be modificated to generate use case scenario.	Analysis	The use case scenarios must be designed and developed inside Dyncar framework to simulate each use case.	none		26.06.2014		New	1	
TEC-DYN-025	Functional	Tools	Failure Trigger definition	A way to trigger the malfunction must be defined in every use case. This trigger must be synchronised with Dynacar and SafeAdapt hardware.	Analysis	To trigger the failure and readaptation, a trigger method must be defined.	none		26.06.2014		New	1	
TEC-PRO-001	Functional	Tools	Standards' requirements	Prossurance should include the requirements established by fuctional safety standards.	Visual check	SafeAdapt is oriented to ISO26262 "Road Vehicles - Functional Safety" so this is the standard to be considered by Prossurance.	none		26.06.2014	ISO 26262	New	1	
TEC-PRO-002	Functional	Tools	Manage safety/assurance case	The user will be able to manage assurance cases, and safety cases in particular. Managing means creating, reading, updating, and deleting the assurance or safety case.	Visual check	Safety cases present the argument that a system will be acceptably safe in a given context. A Safety Case is a structured argument, supported by a body of evidence, that provides a compelling, comprehensible and valid case that a system is safe for a given application in a given operating environment. Safety cases are often required as part of a regulatory process.	none		26.06.2014		New	1	
TEC-PRO-003	Functional	Tools	Compose Compliance Demonstration	The user will be able to prepare the assessment for the assessor by creating an assurance process project baseline which includes all necessary items for assurance or safety demonstration.	Visual check	The baseline needs to be composed referring to the relevant artefacts in the repository.	TEC-PRO-002		26.06.2014		New	1	
TEC-PRO-004	Functional	Tools	Artifact Properties Management	Prossurance should support the management of configurable properties for artefacts.	Visual check	This is useful in case of future reuse of specific artefacts.	TEC-PRO-005		26.06.2014		New	1	
TEC-PRO-005	Functional	Tools	Manage artefacts	The user will be able to add, change, and delete artefacts to, in, or from an associated repository, preferably through interfacing systems.	Visual check	Some artefatcs can be initially generated by external systems, so to save time and money we need interfaces to other specialized tools.	none		26.06.2014		New	1	
TEC-PRO-006	Functional	Tools	Import artefact into repository	The user will be able to import one or more artefacts into the repository from other sources. These other sources include other repositories, development or test environments, or safety dossiers. The functionality is preferably offered by a tool interfacing with the Prossurance tool.	Visual check	Some artefatcs can be initially generated by external systems, so to save time and money we need interfaces to other specialized tools.	none		26.06.2014		New	1	
TEC-PRO-007	Functional	Tools	View traceability to assurance process requirements	The user will be able to view the relation between any product artefact to the assurance process requirement using traceability links.	Visual check	Traceability between different elements coming from the different activities along the system development lifecycle are needed.	none		26.06.2014		New	1	

TEC-PRO-008	Functional	Tools	Manage evidence	The user wil be able to manage evidences (add, remove, edit, etc.).	Visual check	The user will be able to view all the inventory of every piece of evidence, like evidence characterization (by mean of properties), but also the more mundane information like name, time stamp of creation, etc.	none		26.06.2014		New	1	
TEC-PRO-009	Functional	Tools	Navigate repository	The user will be able to navigate through the different views within a repository.	Visual check	Different users may need to see information in different ways for different purposes. System engineers need to see information in a different way than safety engineers.	none		26.06.2014		New	1	
TEC-PRO-010	Functional	Tools	Maintain history	Prossurance should keep track of the state of the repository in order to offer a historical record of the chain of events during the development of the repository.	Visual check	This information can be used to give more insight in evolutionary chain of certification evidence, of the impact of decisions, cost estimations, etc.	none		26.06.2014		New	1	
TEC-PRO-011	Functional	Tools	Version support	Prossurance should keep track of the versions of all items stored in the repository, like artefacts, safety cases, etc.	Visual check	Different versions of the items can be in different steps of the ther life cycle (development, production, maintenance, etc.). So Prossurance need to store information of the different versions of the system and all the related information.	none		26.06.2014		New	1	
TEC-PRO-012	Functional	Tools	Navigate safety / assurance case	The user will be able to navigate through an assurance case (which can be a safety case as well) according to the cases hierarchical structure.	Visual check	Assurance case will encomprise all the information to argument that the system is safe in an strusutured way.	TEC-PRO-009		26.06.2014		New	1	
TEC-PRO-013	Functional	Tools	Support artefact viewing	Prossurance should provide a means for suitably authorized users to view an artefact stored in the repository.	Visual check	Based on an standard security policy, diferent user profiles should be managed in order to ensure that the right information is shown to the right person.	TEC-PRO-012		26.06.2014		New	1	
TEC-PRO-014	Functional	Tools	View process information	The user will be able to navigate through process information (e.g. the author of a document, the number of tests failed in a test activity, etc.).	Visual check	Different users may need to see information in different ways for different purposes. System engineers need to see information in a different way than safety engineers.	TEC-PRO-013		26.06.2014		New	1	
TEC-PRO-015	Functional	Tools	Safety Case Status View	Prossurance should support a view of the current state of the Safety Case.	Visual check	It is important to know the status of the safety case in order to know how far we are from the overall goal of having a safe system.	TEC-PRO-012		26.06.2014		New	1	
TEC-PRO-016	Functional	Tools	View missing evidence	Prossurance should provide a view that shows the missing evidence based on unsatisfied assurance process requiremerns.	Visual check	It is important to know the status of the safety case in order to know how far we are of the overall goal of having a safe system.	TEC-PRO-015		26.06.2014		New	1	
TEC-PRO-017	Functional	Tools	Manage tool configuration	The user will be able to manage the configuration of Prossurance in his/her own platform environment.	Visual check	Prossurance can be used to manage information by different companies so there must be a way to personalize the tool for each case.	none		26.06.2014		New	1	
TTT-TOOL-001	Functional	Tools	TTEthernet data loading tool - TTE-Load	The TTE-Load tool will be used to configure the TTEthernet devices by downloading the configuration images geinerated by the TTE-Build Device Configuration tool.	Analysis	This tool simplifies the download process during development time and for updates late in the product life cycle ensuring that all neosesary software modules are loaded to the TTEthernet devices in the correct format.	none		24.10.2014		New	1	
TTT-TOOL-002	Functional	Tools	TTEthernet network design tool - TTE-Plan	The TTE-Plan tool will be used to generate the schedule for the network. It will use a special network description file.	Analysis	Although scheduling sometimes is seen as a simple task, it becomes quite complex in case structures and communication architectures implemented in a system rise in complexity. The TTE-Plan tool ensures that all parameters needed are set properly and also checks automatically if the parameters desired are possible. The tool also ensures that a time saving, powerful process is provided supporting an efficient way to conduct optimization and modification tasks which can be reproduced and provide a reliable design process.	none		24.10.2014		New	1	
TTT-TOOL-003	Functional	Tools	TTEthernet data loading tool - TTE-Load download procedure	The TTE-Load tool will perform a safe unlocking procedure prior to start reprogramming the static configuration memory. It also will support bootstrap configurations of TTEthernet switches.	Analysis	Whether in a set-up during design time or in a system in the filed, it is essential that the downloading program ensures that the system components are unlocked and are so prepared to receive a new configuration set.	none		24.10.2014		New	1	
TTT-TOOL-004	Functional	Tools	TTEthernet network configuration generation tool - TTE-Build	The TTE-Build Network Configuration tool will be used to extract the data from the network configuration files and calculates the parameters for the individual devices. The results are stored in device configuration files.	Analysis	After designing the schedule, the next step is preparing and calculating the configuration files for the devices based on the input from the scheduling tool TTE-Plan. This is a complex work that involves conducting the same steps over and over again which qualifies the work perfectly for automation by a tool. It also ensures that the files generated are available always in the same identical templates and provides significantly more easy integration. The results are produced in XMl code supporting readbility.	none		24.10.2014		New	1	
TTT-TOOL-005	Functional	Tools	TTEthernet network device Configuration tool - TTE-Build	The TTE- Build Device Configuration tool will be used to convert the XML representation into device compatible image format. Such format can either be binary or hex coded. The result is ready to be loaded to the devices such as switches via TTE-Load.	Analysis	The XML representation requires conversion into device compatible image format. Again, such work follows simple rules that must be performed over and over again and thus is best suited for automation by means of a tool.	none		24.10.2014		New	1	
TTT-TOOL-006	Functional	Tools	TTEthernet network viewing tool - TTE-View	The TTE-View tool is used to monitor, record and dissect any kind of Ethernet traffic.	Analysis	Efficient design also needs to provide a viewing in order to perform any debugging and/or optimization work in an efficient manner. The viewing tool also helps to apply modifications and enhancements for converting existing designs for other products in a similar application (re-use, enhancments in a product family etc.).	none		24.10.2014		New	1	
TTT-TOOL-007	Functional	Tools	TTEthernet network viewing tool - TTE-View -Packet Analyzer	The TTE-View tool is based on the Wireshark Packet Analyzer. It is implemented as a plug-in to Wireshark. The Analyzer is enhanced to dissect and display TTEthernet specific frames.	Analysis	Using Wireshark reduced development effort in the viewing tool and ensuries that the user will be confrinted with a known interface.	none		24.10.2014		New	1	
TTT-TOOL-008	Functional	Tools	TTEthernet Tools databases - TTE-Plan Network Description Database	a) It describes the high level communication requitrements for the system, i.e. physical & logical topology. b) It describes the virtual links (VLs) including their Ids timing requirements and possible frame sizes. c) It describes the synchronization parameters and requirements i.e SAE AS6802 clock	Analysis	A unique data base concept is put behind the entire TTEthernet development tool chain in order to follow a traceable development and design concept. Each tool in the TTEthernet tool chain is covering a defined share of the complete design work to develop a TTEtherent based data communication network.	TTT-TOOL-002		24.10.2014		New	1	
TTT-TOOL-009	Functional	Tools	TTEthernet Tools databases - TTE-Build Network Configuration Data Base	a) It provides the implementation as a set of XML files. b) It contains the network schedule as calculated by TTE-Plan. c) The network configuration is hardware independent. It describes all details necessary to configure a network including schedules, port assignments and the buffer allocation for all devices in the network. d) It supports that parts of the entire network configuration may be created and/or modified by third party tools.	Analysis	A unique data base concept is put behind the entire TTEthernet development tool chain in order to follow a traceable development and design concept. Each tool in the TTEthernet tool chain is covering a defined share of the complete design work to develop a TTEtherent based data communication network.	TTT-TOOL-004		24.10.2014		New	1	
TTT-TOOL-010	Functional	Tools	TTEthernet Tools databases - TTE-Build Device configuration Data Base	a) It generates one configuration per device (switch or end system). b) It generates the device configuration in both formats, XML (for human readability) and different image formats (for direct download to the device). c) The device configuration is device specific and describes every configuration parameter at bit level. Fine tuning of configuration parameters is possible at this level.	Analysis	A unique data base concept is put behind the entire TTEthernet development tool chain in order to follow a traceable development and design concept. Each tool in the TTEthernet tool chain is covering a defined share of the complete design work to develop a TTEtherent based data communication network.	TTT-TOOL-005		24.10.2014		New	1	
TTT-TOOL-011	Functional	Tools	TTEthernet Verification support	a) Several reports show the result of the internal checking functions of the TTE tools. B) The reports are generated i HTML format for easyreadability.	Analysis	Verification is a neccessary step within a design and development process. This feature supports this step to be conducted by the designer.	none		24.10.2014		New	1	
TTT-TOOL-012	Functional	Tools	TTEthernet Tools GUI	The TTE tools come with Eclipse based GUI editors. There are Eclipse plug-ins provided for each tool. Schedule visualization is available. Data base valiadtion including validation reports is supportedvia Eclipse.	Analysis	The TTEthernet tool chain will be used in a complete ensemble for each development process and for modifications, enhancements and improvements of a design. Thus a common GUI is an essential feature to use the tool chain efficiently. Eclipse is understood to be a suitable basis for this.	TTT-TOOL-001 TTT-TOOL-002 TTT-TOOL-004 TTT-TOOL-005 TTT-TOOL-006		24.10.2014		New	1	