Contract number: ITEA2 – 10039

Safe Automotive soFtware architEcture (SAFE)

ITEA Roadmap application domains:

Major: Services, Systems & Software Creation

Minor: Society

ITEA Roadmap technology categories:

Major: Systems Engineering & Software Engineering

Minor 1: Engineering Process Support

WP2, WT2.1

Deliverable D2.3.1.b:

Project Glossary

Due date of deliverable: 30/11/2014

Actual submission date: 30/09/2014

Start date of the project: 01/07/2011 **Duration:** 36 months

Project coordinator name: Stefan Voget

Organization name of lead contractor for this deliverable: Continental Automotive

Editor: Jörg Kemmerich

Contributors: SAFE members during SAFE project

Revision chart and history log

Version	Date	Reason
0.1	16.04.12	Initialization of document
0.2	19.04.12	First version for internal Review
1.0	26.04.12	Integration of review comments, finalization of Draft Deliverable for D6.a
1.1	15.03.2014	Creation of framework for glossary items out of working document
1.2	June 2014	Content for deliverable D6.b inserted by Editor
1.3	30.09.2014	Finalisation, published version

1 Table of contents

2 List of figures

3 Executive Summary

The document at hand (Deliverable 2.3.1) comprises a set of terms and definitions to be used within the SAFE project. The establishment of such a document is mandatory especially for projects where

- A large number of companies from various cultures and techniques are collaborating
- Complex interdisciplinary topics have to be dealt in common

The goal has been to develop a common understanding and naming within the consortium und to provide a public document to be used also for other and/or future activities.

As several glossaries are already defined in adjacent areas and projects, these glossaries are checked first in order to prevent a redundant definition. The remaining entries that need definition in SAFE glossary are separated into a list of acronyms and the glossary itself.

4 Introduction and overview of document

The aim of this document is to provide a SAFE project wide glossary and a list of acronyms.

Both glossary list and list of acronyms have been created at the beginning of the project and have been updated on demand during the project. The results of the glossary activity have been discussed.

5 Glossary Objectives

The objective of this Glossary is to provide a common naming and understanding to the SAFE participants.

The **glossary** contains items that need explanations, explicit definitions, examples, references and/or a discussion.

The **list of acronyms** is meant to simply give the link between an acronym and the corresponding full term.

The definitions provided concentrate on terms in the following fields:

- Automotive electrical/electronic-systems
- Dependability, functional and technical safety, and
- Safety related process steps.

Within SAFE project, the items should be used as given in the glossary. Alternative definitions that are in use within the SAFE project and new definitions that come into being should be added to the document.

5.1 Reference Glossaries

As several glossaries are already defined in adjacent areas and projects, these glossaries are checked first in order to prevent a redundant definition. First and most relevant reference glossary is the glossary of the ISO26262 [1].

In case that an item is defined in another glossary, but the item is refined in SAFE, the refinement is also given in the SAFE glossary together with a reference to the original definition.

6 Deliverable Contents

6.1 Glossary

6.1.1 Integrated Safety System (ISS)

Definition	An Integrated Safety System is a composition of functions and/or components that enhance the level of safety for human beings, inside and outside of the vehicle.
Reference	EASIS-Glossary

6.1.2 System Variant

Definition	A system variant is a specific combination of configuration and calibration parameters that do not change at runtime.
Reference	no reference available
Example	A specific pre-processor instruction and country code is one specific combination of configuration and calibration data

6.1.3 Validation Scenario

Definition	A validation scenario describes operating situations and failure mode where the controllability of the vehicle and the effectiveness of safety measures, external measures and elements of other technologies shall be demonstrated.
Reference	no reference available
Example	A specific pre-processor instruction and country code is one specific combination of configuration and calibration data

6.1.4 Model Based Engineering

Definition	Frontloading of development activities. Definition of functional behavior and Code Generation based on a formal description in form of a model.
Reference	ISO26262 Glossary, Item 1.74 CESAR Glossary
Example	Development steps are structured in a one of the following ways Matlab / Simulink / Stateflow --> Targetlink --> ECU UML/SAFE Model --> Code
Alternative	Configuration based code generation (e.g. CAN interface Generation, BIOS-

Definitions	Generation)

6.1.5 Test Case

Definition	A test case defines a procedure to assure that a system element correctly implements a requirement. A test case consists of a sequence of stimuli and expected system element responses.
Reference	no reference available

6.1.6 SAFE Process Model

Definition	Portion of the SAFE model that describes process steps and activities that are necessary to develop a ASILx system according to ISO 26262
Reference	no reference available

6.1.7 SAFE Product Model

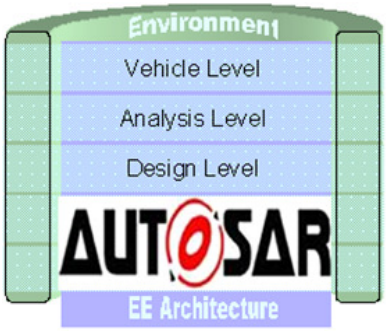
Definition	Portion of the SAFE model that describes product related attributes / properties of any work product that are necessary to develop a ASILx system according to ISO 26262
Reference	no reference available

6.1.8 Dysfunctional Behavior

Definition	Unexpected Behavior w.r.t. specification. Dysfunctional Behavior. Can be modeled
Reference	no reference available

6.1.9 Abstraction Level

Definition	An abstraction level provides a specific level of description and analysis of a design item (e.g. component). On the next lower abstraction level the granularity of the design item is refined. Such a lower abstraction level is realized by specific decomposition techniques. Models on different abstraction levels may differ in both their granularity and their viewpoints. Realized links between models allow tracing those refinements. The use of abstraction layers may support both the reuse of solutions and the management of the supply chain. Note that components of models on lower abstraction levels still have to respect the aspect specifications defined for their higher level counterparts.
Reference	Architecture Modeling; research report from project SPES2020

	(www.spes2020.de)
Example	The automotive domain specific language "EAST-ADL" is organized in four abstraction levels. - Vehicle level defines features and requirements - Analysis level defines the abstract functional architecture - Design level defines concrete functional and course hardware architecture - Implementation level defines software and detailed hardware architecture 

6.1.10 Model Based Safety Analysis

Definition	An approach for automating portions of the safety analysis process using executable formal models of the system.
Reference	A Proposal for model-based safety analysis; A. Joshi et. Al.; Presented at the 24th Digital Avionics Systems Conference, Washington, D.C., October, 2005.

6.1.11 Perspective

Definition	A perspective combines views of different abstraction levels which are related to similar viewpoints. Perspectives can be used to group and to structure views of different disciplines in order to cope with the complex task of developing a system.
Reference	Architecture Modeling; research report from project SPES2020 (www.spes2020.de)

6.1.12 Realization

Definition	A realization describes a mapping between component parts of different abstraction layers.
Reference	Architecture Modeling; research report from project SPES2020 (www.spes2020.de)

6.1.13 Refinement

Definition	Refinement defines the derivation of a concrete description of the design item from an abstract description. The derivation thereby conserves the characteristics of the abstract description. In the case of a contract specification, the derived

	component has to fulfill all contracts of the more abstract component.
Reference	Architecture Modeling; research report from project SPES2020 (www.spes2020.de)

6.1.14 View

Definition	A view is a set of models of the system under development or of a part of this within an abstraction layer with respect to a specific viewpoint. A view addresses one or more concerns.
Reference	ISO/IEC/IEEE 42010 System and software engineering – Architecture description; Architecture Modeling; research report from project SPES2020 (www.spes2020.de)

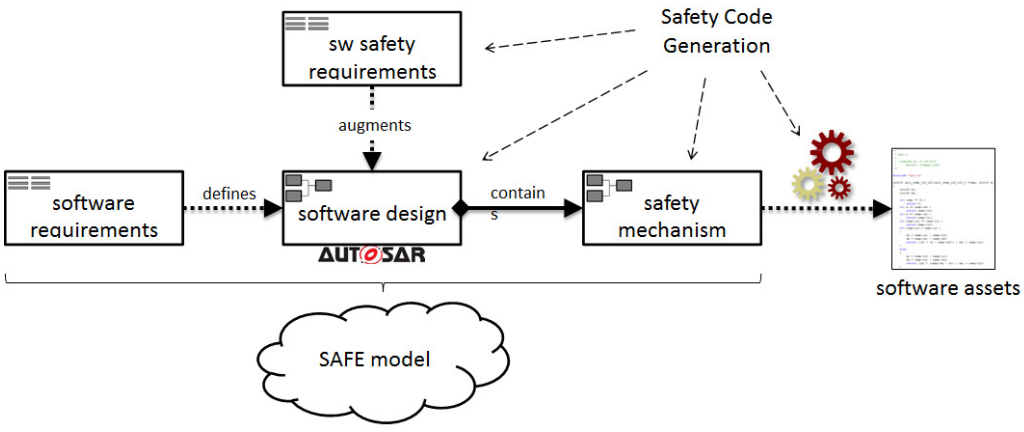
6.1.15 Viewpoint

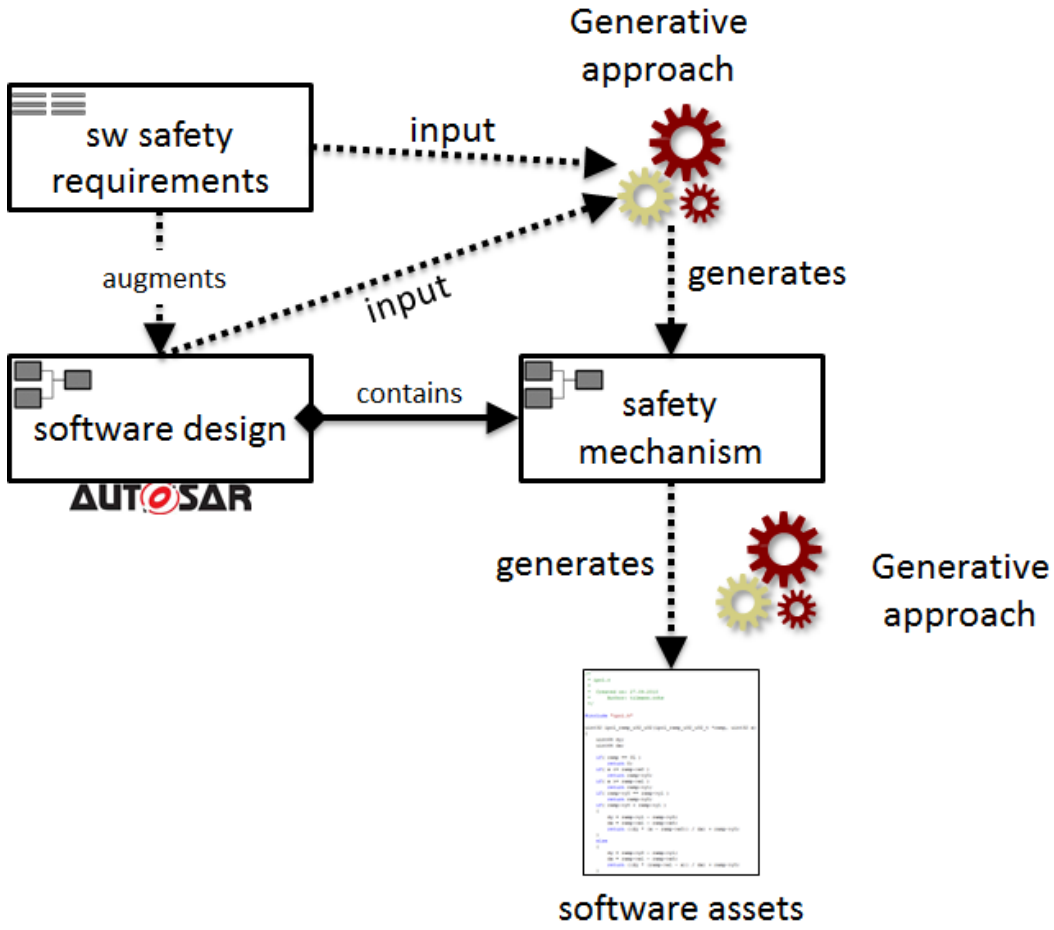
Definition	A viewpoint defines a specific form of abstraction in order to focus on particular concerns within a system. For each viewpoint a selected set of architectural constructs and structuring rules is defined in order to design and use a viewpoint specific view. Thereby, a viewpoint is not constraint to a specific abstraction layer.
Reference	ISO/IEC/IEEE 42010 System and software engineering – Architecture description; Architecture Modeling; research report from project SPES2020 (www.spes2020.de)

6.1.16 Safety extension

Definition	An extension of the system model regarding safety information
Reference	no reference available

6.1.17 Generative approach

Definition	Methodology through which software is obtained via code generators 

	
Reference	no reference available

6.1.18 Fault Containment

Definition	Mechanism to prevent the manifestation of faults in the system
Reference	no reference available

6.1.19 Error Detection

Definition	Mechanism to detect the ocurrence of errors
Reference	no reference available

6.1.20 Error Handling

Definition	Mechanism to handle errors and prevent such from interfering with system operation

Reference	no reference available

6.1.21 Malfunction

Definition	Malfunction is a failure or unintended behavior of the item or element of the item that has the potential to propagate.
Reference	Definition used for D3.2.1, agreed in Oldenburg PTC meeting

6.1.22 Horizontal Error Propagation

Definition	Propagation of errors inside a same architectural level.
Reference	Definition used for D3.3.1

6.1.23 Vertical Error Propagation

Definition	Propagation of errors through different architectural levels
Reference	Definition used for D3.3.1

6.1.24 Hazardous Event

Definition	A hazardous event is a combination of a hazard and an operational situation.
Reference	Definition used for D3.2.1

6.1.25 Safety Relevant Failure

Definition	Safety relevant failures are failures that are identified during safety analyses to have the potential to lead to a violation of a safety goal
Reference	Definition used for D3.2.1

6.1.26 Domain Model

Definition	result of modelling activities that are neccessary for a topic
Reference	no reference available

6.1.27 Horizontal Layer

Definition	collection of properties within the same architectural level
Reference	Definition used for D6.x, no reference available

6.1.28 Vertical Layer

Definition	collection of properties within different architectural level
Reference	Definition used for D6.x, no reference available

6.2 Acronyms

Item	Full Name
RTP	Reference Technology Platform
PMHF	Probabilistic Metric for random Hardware Failures
RF	Residual Fault
SPF	Single Point Fault
SPFM	Single Point Fault Metric
LF	Latent Fault
LFM	Latent Fault Metric
ETC	Electronic Throttle Control
Soc	System On Chip
SEooC	Safety Element out of Context
MTBF	Mean time between failures
TRL	Technology Readiness Level
ETC	Electronic Throttle Control
SSR	Software Safety Requirement
SSM	Software Safety Mechanism
ASIL	Automotive Safety Integrity Level
ATTEST	Advancing Traffic Efficiency and Safety through Software Technology
AUTOSAR	AUTomotive Open System ARchitecture
BCM	Body Control Management
BDD	Binary Decision Diagram
CAE	Computer Aided Engineering
CAN	Controller Area Network
CCF	Common Cause of Failure
CESAR	Cost-Efficient methods and processes for SAfety Relevant embedded systems
COTS	Component Off the Shelf
CPU	Central Processing Unit
DM	Degradation Mode
DRIS	Distributed, Reliable and Intelligent control and cognitive Systems
E/E	Electronic and Electrical
EAST-ADL	Electronic Architecture and Software Tools- Architecture Description Language
ECU	Electronic Control Unit
EMC	Electro Magnetic Compatibility
ETA	Event Tree Analysis
FDA	Function(al) Design Architecture
FIT	Failure In Time
FME(D)A	Failure Mode Effect and Diagnostic Analysis
FMEA	Failure Mode and Effect Analysis
FTA	Fault Tree Analysis
GUI	Graphical User Interface
HAZOP	HAZard and OPerability study
HDA	Hardware Design Architecture
HiP-HOPS	Hierarchically Performed Hazard Origin & Propagation Studies
HRC	Heterogeneous Rich Components
HW	Hardware

IP	Intellectual Property
LFM	Latent Fault Metric
LH	Limp Home
MAENAD	Model-based Analysis & Engineering of Novel Architectures for Dependable electric vehicles
MCU	Microcontroller Unit
OEM	Original Equipment Manufacturer
Open-PSA	Open Probabilistic Safety Assessment
RAM	Random Access Memory
RBD	Reliability Block Diagram
RSL	Requirements Specification Language
RTE	Real Time Environment
SAFE	Safe Automotive soFtware architEcture
SM	Safety Mechanism
SPEEDS	Speculative and Exploratory Design in Systems Engineering
SPFM	Single Point Fault Metric
SW	Software
SWC	Software Component
TCM	Top Column Module
WT	Work Task
XML	Extensible Markup Language
EAST-ADL	Electronics Architecture and Software Technology - Architecture Description Language
FAA	Function Analysis Architecture

7 Conclusions and Discussion

In the first phase of the SAFE project a first set of items for the glossary list and the acronym list has been created.

Later until the end of the SAFE project these lists have been finalized in order to provide a public document.

8 References

[1] ISO 26262(1) Vocabulary

[2] AUTOSAR Glossary

[3] CESAR Glossary

9 Acknowledgments

This document is based on the SAFE project in the framework of the ITEA2, EUREKA cluster program Σ! 3674. The work has been funded by the German Ministry for Education and Research (BMBF) under the funding ID 01IS11019, and by the French Ministry of the Economy and Finance (DGCIS). The responsibility for the content rests with the authors.