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 D2.1.b:

Needs description to apply ISO26262 with architecture and component modeling

Due date of deliverable: 31/12/2012

Actual submission date: 31/12/2012

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: Stefan Voget

Contributors: All project partners

Revision chart and history log

Version	Date	Reason
0.1	28.10.11	Initialization of document
0.2	08.11.11	Inclusion of comments from Philippe Cuenot
0.3	15.11.11	Add outcome of WP2 Meeting (10.-11.11.2011) and discussion with Philippe Cuenot
0.4	21.11.11	More details on scope definition due to mails by E. Andrianarison and J. Lucas
0.5	22.11.11	Add comments from H.-L. Ross
1.0	01.12.11	Description of new procedure (chapter 7) due to discussion in WT2.1
		Finalization of deliverable D2.1.a
1.1	29.11.12	Update due to several change request
2.0	17.12.12	Finalization of deliverable D2.1.b

1 Table of contents

2 Executive Summary

This document acts as a frame document for the work done on requirements identification, analysis and elicitation in the SAFE project. It lists all derived project requirements as well as it gives basic rules, definitions, guidelines and procedures for the work on requirements.

The requirements work has been done in three work tasks, each analyzing another source of possible requirements:

- WT2.1 analyzed the ISO26262,
- WT2.2 analyzed the State of the art and
- WT2.3 analyzed industrial Use case scenarios.

In addition, the document concretizes the work on needs to apply ISO26262 requirements within the project. The requirements itself are documented in a couple of Excel based tables that are added as appendixes to this document.

The collection of this Word document together with the tables represents the complete deliverable D2.1 (Needs description to apply ISO26262 with architecture and component modeling).

3 Introduction

The goal of work-package WP2 is the elicitation of the project requirements for the work-packages of SAFE. This is done by the analysis of three different sources: the ISO26262 standard, a state of the Art analysis and an industrial use case analysis.

- ISO 26262 related Requirements
  - elicited in WT 2.1
  - Documented in tables that are structured according to ISO Structure (Part 2, 3, 4, 5, 6, 7, 8, 9)
- Requirements related to State of the Art analysis
  - elicited in WT 2.2
  - Joint table template with requirements identified by WT 2.3
- Requirements related to model based development
  - elicited in WT 2.3
  - Joint table template with requirements identified by WT 2.2

This document handles with the ISO 26262 related requirements identified, analyzed and elicited in work-task WT2.1 but also gives definitions valid for all three requirements related activities.

The purpose of WT2.1 is to analyze the ISO26262 document, in context of model based development techniques, considering the system design flow, starting from abstract functional representation getting down to concrete technical representations of the hardware and software (sub-) systems.

If it is necessary to comply with ISO26262, architecture and component modeling must fulfill the requirements of the standard, and need to be able to justify the relation to the technical constraints of the hardware and software design (firmware and application software). The recommended methods and measures have to be specified in terms of design activities and work products, and define the verification criteria that must be used to perform the safety analysis. All the selected tools used for development must accomplish the recommended qualification criteria.

The requirements elicited in WT2.1 are not complete with respect to all requirements listed in the ISO26262. The collection of chosen requirements is a first separation of project relevant requirements. This statement is important due to two aspects

1. A project has always limited resources. We have to select a subset of all requirements such that we can guarantee to create a substantial contribution at least for the selected subset of requirements.
2. The ISO26262 does not request specific methods but requests the existence of activities that have to produce a specific result. The norm leaves it open which methods are chosen. In case the norm lists a specific method like FMEA, FTA, test methods, inspections, verifications, etc. these methods are mentioned as examples only. The job of SAFE is to develop methods regarding ISO26262 and regarding model based development. Therefore, the scope does not contain all requirements from the norm.

The detailed analysis and synthesis of requirements derived from the ISO standard are documented with respect to:

- Terms and definitions (to facilitate common understanding)
- Relation with model based development and mapping to WP3 tasks for safety goal definition, safety requirements definition and tracking, architecture definition at different levels, i.e. system, hardware, software
- Identification of needs for methods in WP3 and tools from WP4 to analyze, assess and verify safety criteria and properties of the different assets and reduction of the systematic error rate
- Identification of needs for application tools in WP6

As this task was the first task within the SAFE project, the facilitation of common understanding between the partners had been a major work and result. Therefore, not only the WT2.1 partners were involved. All project partners took part – the ones involved in WT2.1 using the view of WT2.1 and the ones not involved in WT2.1 using the view of WP3, WP4 and WP6. Why the incorporation of these different views had been essential for the progress in the task is described in the next chapters.

4 Is a requirement in or out of scope for SAFE?

The work in WP2 resulted in a detailed scope of the SAFE project with respect to the existing work-packages. A key role takes the differentiation of project artifacts into “product model” and “process model”. The definitions of these wordings are presented in the first section.

Based on these definitions inclusion-, exclusion-lists are created that give examples for the scope definition. These lists are presented in the second section of this chapter.

Both together helped to assign the requirements to the project (having a look to the inclusion-, exclusion-lists) and to the work-packages (having a look to the definitions).

4.1 Definition of “Product Model” and “Process Model”

As there were several different wordings in first versions of the requirements tables we defined two artifacts that shall be used in the requirements definition.

Product Model

A product model is an identifier of a product given by its manufacturer. This contains requirements on information / data that have to be attached to the item or sub item under design. These data may be implemented as properties/attributes which are attached to some artifact of the model.

Once the project is finished and if the product is reused in another project all these information will be reused as well.

Examples that are contained in a product model are

- Requirements
- Analysis Models (FMEA, FTA, Dysfunctional Model, ...)
- Functional Model
- Structure Models (Interfaces, ...)
- Behavior Model (Information flow, Input- / Output-Relation, ...)

To be safe, an item shall demonstrate at the end some specific properties:

- minimal cut sets order (computation of the dysfunctional model)
- architecture metrics (computation of basis failure rates ...)
- diagnosis ability (coverage)
- satisfaction of initial expectations (computation of requirements traceability)

Process Model

Process models are processes of the same nature that are classified together into a model. Thus, a process model is a description of a process at the type level. Since the process model is at the type level, a process is an instantiation of it. The same process model is used repeatedly for the development of many applications and thus, has many instantiations. One possible use of a process model is to prescribe how things must/should/could be done in contrast to the process itself which is really what happens (www.wikipedia.org).

Examples that are contained in a process model are

- System / SW / EE Development process
- B2B Processes
- Safety Planning
- Design activities
- V&V activities
- Resources: experts, reviewers, assessors, test benches, fault injection

A “product model” artifact defines how one describes the product (i.e. the means to do model based development) and a “process model” artifact defines the activities you run to get the description of the product.

The “product model” related artifacts are mainly addressed to WP3. The “process model” related artifacts are mainly addressed to WP6. As requirements are sometimes not atomic this relation is not 1 to one but one requirement may address several aspects to be solved in different work-packages.

4.2 Inclusion / Exclusion List

To decide if an ISO26262 requirement is part of the project we introduced an Inclusion/Exclusion list that eases the decisions of in/out in the Excel sheet.

Inclusion List

1st level: a requirement directly addresses one of the main objectives of SAFE

- An extension of the AUTOSAR architecture model, in order to effectively integrate artifacts associated with the application of the ISO26262 will be provided. The extended model will be implemented in a technology reference platform.
- Methods, e.g. for efficient capturing of safety goals and requirements as well as for safety evaluation or conformance testing, will be enhanced, in order to benefit from the integrated model. To allow evaluation of the methods within significant industrial case studies, the technology reference platform will be extended with a set of appropriate plug-Ins.
- An ISO26262 compliant process will be defined on top of model-based development using AUTOSAR, and evaluated in realistic and measurable industrial case studies, involving the complete automotive supply chain.

2nd level: examples that detail the scope

- Software Engineering out of Context
  - Application of sufficient qualification of previous developed components
  - Representation of assumed requirements
- Application of Proven in Use argument
- Artifacts for product model relevant supporting processes / reuse / variants etc.
- ...

Exclusion List

- Tool Qualification

    - SAFE develops a tool platform to enable the operation of a process for safety critical systems. But it does not handle with the qualification of the tools itself, neither of the SAFE technology platform nor any other tool.
- Project Management
- Management of Infrastructure (Change and Configuration Management)
    - SAFE concentrates on the model based development related activities only. Therefore, such process parts are out of scope. These are taken from state of the art and not created new.
- Processes for Variant Management
    - With the same reason as was given before. Nevertheless, as variant management is essential for the safety lifecycle, one has to note that this does not include the artifacts needed for handling variants – see inclusion list.
- ...

The example of the mentioned aspects of variant handling shows that it is not sufficient to include or exclude high level keywords but to detail the aspects associated with it. In this case for example variant management in the sense that one needs means to express variability points in architecture is included. This is meant by "Artifacts for product model relevant ... variants ...". Variant management in the sense how one runs the process to come from a variability model to a decision model is excluded. This is meant by "no processes for variant management ...".

5 Regulations to optimize requirements documentation

In the preparation to the project and in the first period words like "model", "product", "process" were used by different authors in different ways. Especially the word "model" has been used very often in the FPP as a generic placeholder for a so far unknown project artifact. To be with the requirement description more close to the assignment to work-packages we introduced boiler plates

5.1 Boiler Plates used in requirement description

To describe a requirement close to the categorization defined in the previous subsection the following boiler plates were introduced:

- Safe Meta Model (Modeling level M2)
  - Safe < > shall support ...
  - … is related to < >
- < > ::=
  - Product artifacts
  - Process artifacts (this also includes methods)
  - Tool artifacts
  - Miscellaneous

The category “Miscellaneous” should be used carefully. In such a case further explanation in the requirements description should be given to clarify why none of the other categories is applicable.

5.2 Version of ISO26262

During first months of the project the FDIS version has been used. In November 2011 the ISO26262 were published as international standard. From that point of time the first IS version has been used.

5.3 Examples for clarification of the definitions

Example 1 (taken out of Product_Development_System_Level)

- OLD: For safety mechanisms that prevent dual point faults from being latent, the respective ASIL definition shall be automatically calculated based on the rules defined in requirement 6.4.4.4 (ASIL B for technical safety requirement ASILD, ASILA for technical safety requirement ASILB/C, engineering judgment for technical safety requirement ASIL A)
  - Not clear stated what the project shall do
- IMPROVED: Safe process shall support automatic assignment of ASIL for safety mechanisms that prevent dual point faults from being latent. (See 6.4.4.4 for the dedicated rules; example watchdog build-in safe test)
  - Boiler plates have been used; this enables a clear assignment to WP6.

Example 2 (taken out of Supporting_Processes)

- OLD: The Safe model shall support traceability by allowing for the allocation of safety requirements to elements or items, in a manner that allows for impact analyses.

  - The word “model” has been used in several interpretations. Therefore we decided to introduce a unique categorization.
- IMPROVED: Safe product shall support traceability between safety requirements and allow the allocation of safety requirements to elements or items, in a manner that allows impact analysis.

Example 3 (taken out of Product_Development_HW_Level)

- OLD: The Safe model shall allow to define traceability between hardware safety requirements and hardware components (no trace necessary to low level implementation)
  - The word “model” has been used in several interpretations. Therefore we decided to introduce a unique categorization.
- Improved: Safe product shall allow to define traceability between hardware safety requirements and hardware components (no trace necessary to low level implementation)
  - Boiler plates have been used; this enables a clear assignment to WP3 in this case.

6 Requirements analysis procedure

The work in WT2.1 has shown that it is not sufficient for the SAFE project to collect the requirements out of the ISO26262 and to document them in a deliverable only. Furthermore, the SAFE requirements have to be used to ensure a close link between the work-packages. Especially, between the work-packages that develop project results and the work-packages that evaluate the project outcome.

To ensure, that the requirements are available at an early state and that the fulfillment by the work-packages can be checked later, WT2.1 introduced two iterations for the deliverable D2.1. They are named D2.1.a and D2.1.b.

The degree of maturity of D2.1 and the activities to reach them are as follows:

1. D2.1.a

   In a first step the WP2 completed requirements collection, documentation, sorting and a first, preliminary allocation to the SAFE work-packages. This requirements allocation is preliminary as the work-packages had not been started at this point of time in the project.

   The project is structured in three realization loops. The deliverable D2.1.a is the starting document for the first project iteration.

2. D2.1.b

   Based on D2.1.a and the enclosed preliminary allocation to work-packages, the work-tasks start their work. It is in responsibility of the work-tasks to refine the contribution of the task to the allocated requirement.

   As the work in WP3, 4 and 6 is organized in three loops, this second step is iterated, too. With each work loop it gets clearer how the outcome looks like and which aspect of a requirement is fulfilled and which cannot be fulfilled. Therefore, after loop 1 and loop 2 a review and iteration of the requirements tables is done. This activity is moderated by WT2.1 members.

   D2.1.b summarizes the final set of requirements after the first loop and includes the intermediate traceability matrix, i.e. in the remaining time of the project the refinement within the other work-packages will be continued.

The WP's refined requirements will be documented and maintained among project duration using traceability mechanism from the initial WT2.1 requirement, to demonstrate the conformance of the proposed methods with the ISO26262 standard, and facilitates process aspects and assessment definition.

6.1 Overview about requirements traceability in SAFE project

The following picture shows the relationship between the work-packages with respect to the requirements traceability in the SAFE project.

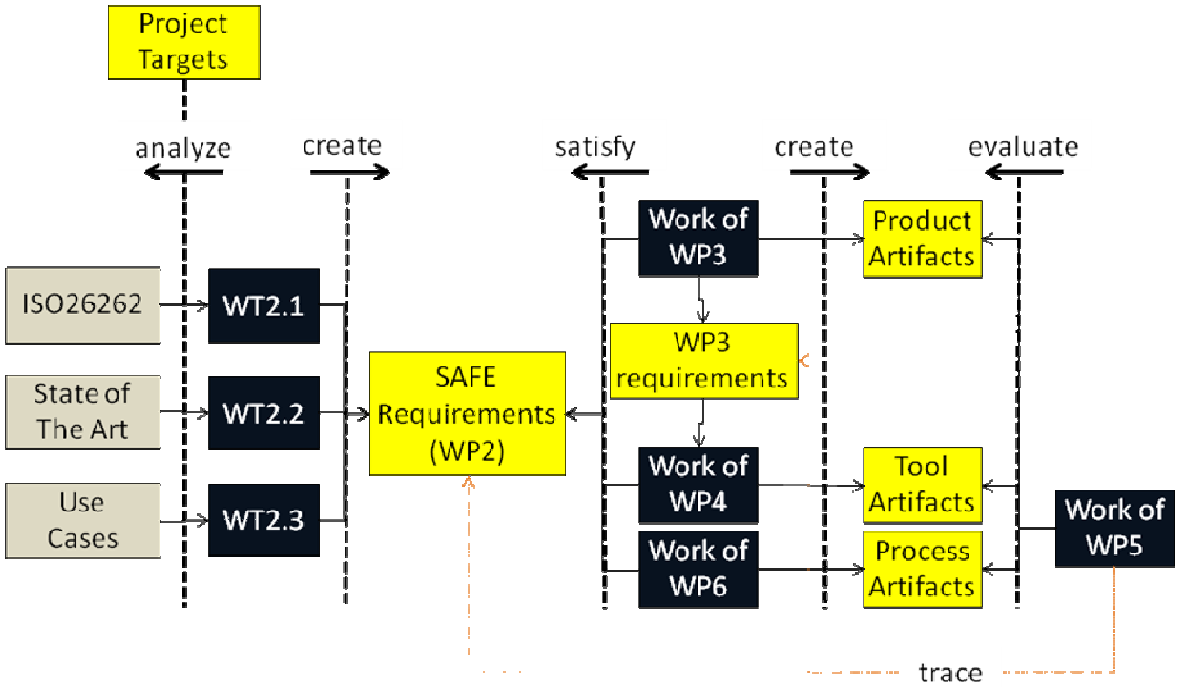

Figure 1: Requirements tracing throughout the project work-packages

WT2.1 analyses the ISO26262 norm, extracts the requirements that are compliant with the project targets and maps these requirements to the work-packages WP3, WP4 or WP6.

The source for WT2.2 is the state of the art analysis and the source for WT2.3 is the use case analysis. But, with respect to creation or extraction and mapping of requirements all three tasks do the same. At the end the SAFE requirements consists of a collection of requirements to be satisfied by WP3, WP4 and WP6.

With respect to requirement satisfaction the product artifacts created by WP3 are used in WP4 for the technology platform.

WP5 has the job to evaluate the results of the project, i.e. the outcome of WP3, WP4 and WP6. This is done in comparison with the traced requirements from WP2 and WP3.

6.1.1 Requirement Traceability Report

All derived project requirements are traced to the work-packages. To manage the traces the tool “Reqtify” is used. Figure 2 presents a report from Reqtify showing the actual coverage of the requirements by the work-packages.

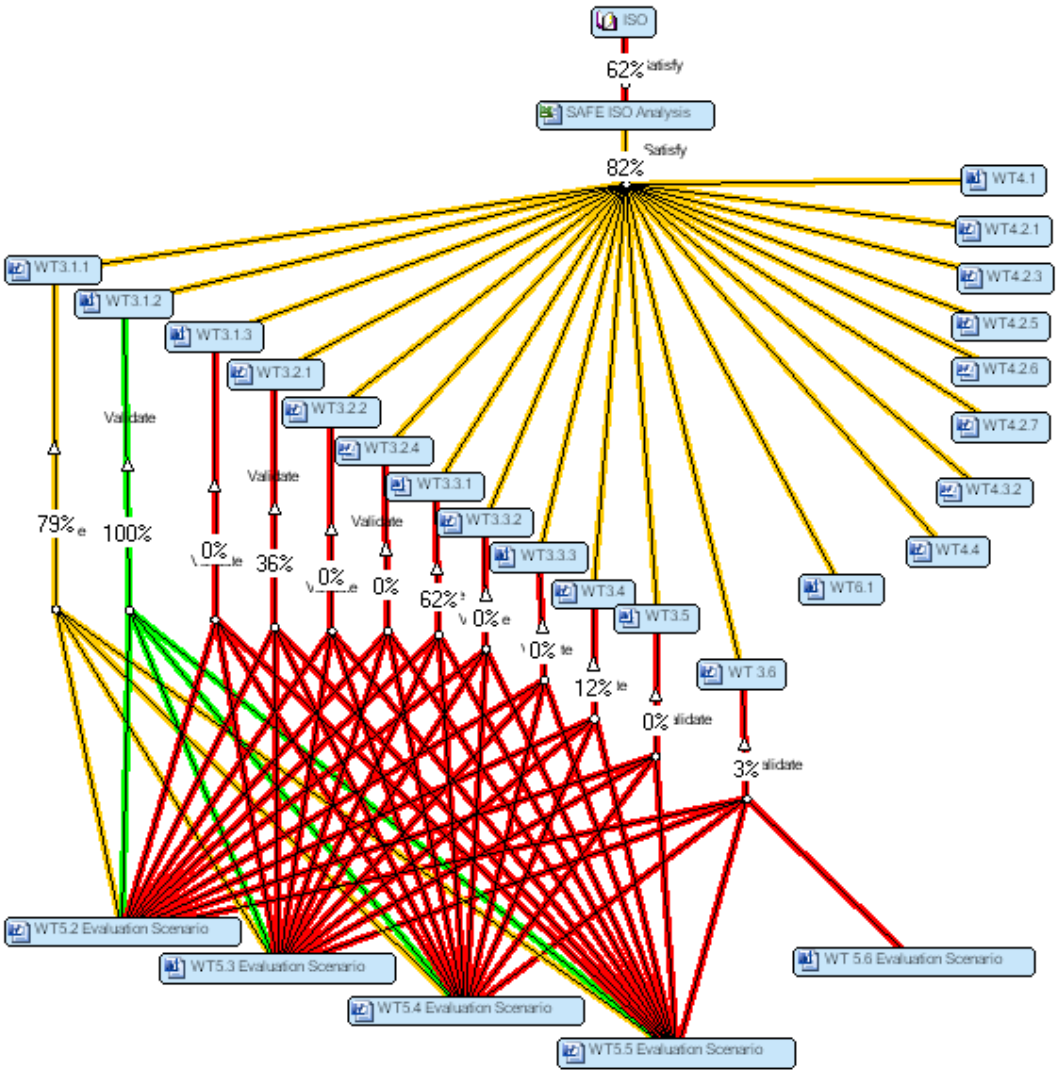

Figure 2: Tracability report from Reqtify

It shows that 62% of the ISO requirements have been selected as requirements target product design and early verification, and then traced by the project. These were taken as the initial project requirements and were allocated to work-packages. In a second iteration each work-package analyzed if they are able to cover the allocated requirements or not. As an outcome all analyzed project requirements were identified as “Included”, i.e. they will be covered by a work-package, or “Excluded”, i.e. if the concern is capable to be managed using model based technology or there is no resource in the project that is able to fulfill the requirement. From all project requirements that have been identified to be in scope of the project (Included) **82%** are already covered by the work-packages. Of course, the goal is to reach 100%. But a high amount is already reached at halftime of the project. Still around 80 requirements have to be taken into account during the second half of the project.

The evaluation of work-package 3 derived requirements by work-package five is also seen in the picture. Actually in most cases the percentage of coverage is low depicted by the red color of the connection link. This reflects the fact that the evaluation scenario has not been fully detailed at this point of time in the project, and that each use case cannot validation all project requirement as the overall scope is too large.

7 Requirements template

In this section, the template to collect and allocate the SAFE requirements documents is described.

7.1 The Excel Sheets

Column	Description	
A	ISO 26262 Part # Title Area of product lifecycle management handled in the table.	
B	ISO 26262 Part # Subtitle In case of product development the discipline is mentioned here	
C	Project internal ID of requirement ID numeration: REQ [ISO Part]_[Sub Nr.]. For split requirements add alphanumerical info and WT info REQ [ISO Part]_[Sub Nr.]_[a-z]_[WT#] Examples: REQ 04_001, Split: REQ 04_001_a_WT#	
D	Requirement Description of the requirement based on the regulations for description as introduced above.	
E	Requirement on One of the categories from the following list: – Product – Process – Tool – Miscellaneous This attribute categorizes the requirement into the working areas handled within the SAFE project. If one is able to decide on a unique category here, the assignment to a work-package is more obvious.	
F	Included / In Work / Excluded Decides if a requirement is handled within the SAFE project or not. During the iterations after each work-loop such a decision may be changed. • Included	

	This requirement is relevant for SAFE and should be handled by at least one work-package. • In Work It may happen that it takes longer to come to a clear and agreed requirement formulation. Due to the need for refinement or due to no common decision, consensus the requirement may not be finalized. As resources are limited this may still exist until the end of the process. In such a case – which shall not be the rule but an exceptional case only – the requirement is taken for succeeding activities. • Excluded This requirement is not handled in SAFE. It may happen that a requirement is redundant or is out of scope. This does not mean that it is of no relevance for the SAFE project but due to resource limitation it may be taken out for succeeding activities.	
G	Reference to ISO Reference to ID in ISO26262 where the origin of the requirement is located.	
H-K	Normative for ASIL … For each ASIL level (A, B, C, D) it is mentioned if the requirement is - Recommended (+) - Highly recommended (++) The notation of the ISO norm is used here.	
L-Z	Relevant for SAFE WT(s) For each work-task it is stated if the requirement is allocated to this work-task. One has to note, that the allocation has been done on task level as some work-packages are split in several tasks that may handle different aspects of the dedicated requirement. The cells in this area are detailed step by step. In a first iteration the WP2 members made a coarse allocation by typing an “X”. In a detailing step each task identified the specific aspect it will handle to satisfy the dedicated requirement. In such a case a description of the aspect has been done within the cell.	
AA-AN	! These columns are hidden in the deliverable but part of the internal work organization ! Relevant for Safe-E WT(s) Due to different funding authorities the subproject SAFE-E had to define deviating work-package numberings. The SAFE-E partners allocated the requirements to	

	their specific numbering in these columns. As SAFE-E is a subproject of SAFE, an entry in the SAFE-E columns always has a correspondent entry in the SAFE e columns. The other way around is not valid as SAFE-E does not handle all topics which are addressed in the SAFE project.	
AO	! This column is hidden in the deliverable but is part of the internal work organization ! Questions / Remarks Any additional remarks, comments, discussions, questions; this column is for project internal work only and is hidden in the final, official deliverable.	

8 Conclusion and Discussion

In the first phase of the SAFE project the work on requirements in WT2.1 helped to create a common understanding on the scope of the project. As the scope of the individual work-tasks still had to be refined at that level of the project, the allocation listed in D2.1.a was still preliminary. In a second iteration the work-packages analyzed if they are able to cover the allocated requirements or not. As an outcome all analyzed project requirements were identified as “Included”, i.e. they will be covered by a work-package, or “Excluded”, i.e. there is no resource in the project that is able to fulfill the requirement.

The following table informs about the number of analyzed (first iteration) and “Included” (second iteration) requirements.

	Number of … requirements	
	Analyzed	**Included**
Management of functional safety	41	5
Concept Phase	105	98
Product development at the system level	145	113
Product development at the hardware level	115	95
Product development at the software level	52	28
Production and operation	4	4
Supporting Processes	44	23
ASIL-oriented safety-oriented analysis	61	54
WT2.3 requirements	63	54
Sum	**630**	**474**

474 requirements to be handled is a big number but the continuously maintained requirements management helps to ensure their fulfillment and strengthens the overall project scope. Nevertheless, 156 requirements that would be necessary to reach a compliancy of the ISO26262 norm are not in the scope of the SAFE project. They are mainly in scope verification and process area, than can be correlated to others or new initiatives.

9 References

[1] SAFE_D2.1.a-ISO-Part_2.pdf (Management of functional safety)

[2] SAFE_D2.1.a-ISO-Part_3.pdf (Concept Phase)

[3] SAFE_D2.1.a-ISO-Part_4.pdf (Product development at the system level)

[4] SAFE_D2.1.a-ISO-Part_5.pdf (Product development at the hardware level)

[5] SAFE_D2.1.a-ISO-Part_6.pdf (Product development at the software level)

[6] SAFE_D2.1.a-ISO-Part_7.pdf (Production and operation)

[7] SAFE_D2.1.a-ISO-Part_8.pdf (Supporting Processes)

[8] SAFE_D2.1.a-ISO-Part_9.pdf (Automotive Safety Integrity Level (ASIL)-oriented safety-oriented analysis

[9] ISO/FDIS 26262 parts 2-9: 2011.

10 Acknowledgments

This document is based on the SAFE and SAFE-E projects. SAFE is 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). SAFE-E is part of the Eurostars program, which is powered by EUREKA and the European Community. The work has been funded by the German Ministry of Education and Research (BMBF) and the Austrian research association (FFG) under the funding ID E!6095. The responsibility for the content rests with the authors.