Graz University of Technology

Master Thesis

Efficient Requirements Management Considering Automotive Standards: Best Practice Sharing of Mechatronic Engineering within MAGNA Groups

Gunther Spork

Institute of

Production Science and Management

Member of Frank Stronach Institute

Graz, 2011

Statement of Independence

Hiermit erkläre ich, dass diese Diplomarbeit mein eigenes Werk ist und auf meinen eigenständigen Untersuchungen basiert. Beiträge Anderer (beispielsweise Ideen, Abbildungen, Textpassagen, Tabellen) sind an den entsprechenden Stellen durch einen Quellverweis gekennzeichnet. Weiters erkläre ich, dass diese Arbeit von mir noch nie als Abschlussarbeit oder zur Erlangung eines akademischen Grades eingereicht wurde.

I declare that this diploma thesis is my own work, based on my original research and expressed in my own words. Any use made within it of works of others in any form (e.g. ideas, figures, text, tables) is properly acknowledged at the point of use. I have not submitted this thesis for any other course or degree.

Graz, 26.2.2011 *Gunther Spork*

Abstract

The aim of this thesis is the creation of a requirements management business model for an automotive supplier. It should not only fulfil the requirements of currently relevant standards in an efficient way but also support strategic aspects like protection of intellectual property and re-use of product features. After a short introduction in chapter 1 the context for requirements management is set in chapter 2. It includes the history of quality management in automotive industry with special focus on engineering. In addition the concepts of Automotive SPICE, ISO/DIS 26262 and the V-model are introduced. Chapter 3 is dedicated to identify best practices for requirements engineering at MAGNA. It includes 5 case studies of three different MAGNA groups. In chapter 4 an exemplar requirements management business model is described in detail. It is based on inputs of personal experience, literature research and the identified best practices of the case studies. The suggested model could be used for the establishment of a new system or selected aspects may be integrated in an already established development process. The final chapter offers a conclusion and outlook.

Acknowledgement

Für Petra, Christina und Silvia.

Table of Contents

STATEMENT OF INDEPENDENCE III

ABSTRACT IV

ACKNOWLEDGEMENT V

TABLE OF CONTENTS VI

1 INTRODUCTION 1

1.1 OBJECTIVES 1
1.2 ABOUT MAGNA 3

2 QUALITY MANAGEMENT IN AUTOMOTIVE BUSINESS 5

2.1 HISTORY OF QUALITY MANAGEMENT 5
2.2 QUALITY MANAGEMENT IN AUTOMOTIVE INDUSTRY 7
2.3 NEW REQUIREMENTS REGARDING QUALITY MANAGEMENT IN AUTOMOTIVE ENGINEERING 9
2.3.1 AUTOMOTIVE SPICE 14
2.3.2 ISO/DIS 26262: ROAD VEHICLES – FUNCTIONAL SAFETY 30
2.4 V-MODEL IN REQUIREMENTS ENGINEERING 38
2.5 MAIN TOOLS IN AUTOMOTIVE REQUIREMENTS ENGINEERING 43

3 BEST PRACTICES FOR REQUIREMENTS ENGINEERING AT MAGNA 46

3.1 CASE STUDY MAGNA POWERTRAIN D&CCS 47
3.1.1 GENERAL OVERVIEW 47
3.1.2 REQUIREMENTS MANAGEMENT STRUCTURE 48
3.1.3 LINK BETWEEN REQUIREMENTS MANAGEMENT AND TEST MANAGEMENT 54
3.1.4 SUMMARY 57
3.2 CASE STUDY MAGNA E-CAR, THONDORF 58
3.2.1 GENERAL OVERVIEW 58
3.2.2 REQUIREMENTS STRUCTURE 61
3.2.3 LINK BETWEEN REQUIREMENTS MANAGEMENT AND TEST MANAGEMENT 65
3.2.4 SUMMARY 65

3.3 CASE STUDY MAGNA POWERTRAIN ELECTRONICS, SAILAUF ... 67
3.3.1 GENERAL OVERVIEW ... 67
3.3.2 REQUIREMENTS MANAGEMENT STRUCTURE ... 70
3.3.3 LINK BETWEEN REQUIREMENTS MANAGEMENT AND TEST MANAGEMENT ... 75
3.3.4 SUMMARY ... 76
3.4 CASE STUDY MAGNA MIRRORS & CLOSURES, SAILAUF ... 77
3.4.1 GENERAL OVERVIEW ... 77
3.4.2 REQUIREMENTS MANAGEMENT STRUCTURE ... 77
3.4.3 LINK BETWEEN REQUIREMENTS MANAGEMENT AND TEST MANAGEMENT ... 80
3.4.4 SUMMARY ... 80
3.5 CASE STUDY MAGNA POWERTRAIN ELECTRONICS, ROCHESTER HILLS ... 82
3.5.1 GENERAL OVERVIEW ... 82
3.5.2 REQUIREMENTS MANAGEMENT STRUCTURE ... 84
3.5.3 LINK BETWEEN REQUIREMENTS MANAGEMENT AND TEST MANAGEMENT ... 88
3.5.4 SUMMARY ... 88
3.6 CONCLUSION OF CASE STUDIES ... 89

4 REQUIREMENTS MANAGEMENT BUSINESS MODEL ... 90

4.1 ASSUMPTIONS AND PRECONDITIONS ... 90
4.2 PROJECT LIFECYCLE ... 90
4.3 STRUCTURE OF REQUIREMENTS MANAGEMENT ... 94
4.4 FUNCTIONAL RE-USE ... 98
4.5 RELATIONSHIP BETWEEN REQUIREMENTS MANAGEMENT SYSTEM AND OTHER PROCESSES ... 100
4.6 SUMMARY ... 101

5 CONCLUSION AND OUTLOOK ... 103

LIST OF ABBREVIATIONS ... 107

LIST OF REFERENCES ... 108

WEBLINKS ... 112

LIST OF FIGURES ... 113

LIST OF TABLES ... 116

1 Introduction

'Quality has become an increasingly important means of competition on the world market. A strategy based management commitment to continuous quality improvement has therefore to be applied more generally and systematically in any organization to enable it to keep its position on the market. Otherwise, large shares of the market will be lost to those competitors who are more aware of the importance of quality.'[1]

This general statement may apply even more for system suppliers in automotive business facing the challenge to handle conflicting targets like shorter development time, innovation, increasing functionality, cost and weight reduction or reliability. One, if not the predominant lever to achieve those targets is product development and there especially mechatronic engineering. This was recognized by automotive OEMs (Original Equipment Manufacturer) and led to creation of new, automotive specific standards in the last 5 years.

1.1 Objectives

The objective of this thesis is to create a business model for requirements engineering considering the new standards. This model will be based on the experience of several years of personal experience in this field, supplier assessments, literature research and most import on the analysis of case studies of the main mechatronic engineering sites of MAGNA. In many companies you will hear criticism regarding the effort related to the fulfilment of new standards like Automotive SPICE and ISO/DIS 26262. Although there is no doubt that these standards support the quality of product development the implementation may be combined with extensive paper work. The situation can be compared with the early days of the ISO9000-series when some companies improved their processes and others mainly increased documentation. It is therefore necessary to identify the area of Best Practices, where quality and profit is high and costs are low. This is also shown in Figure 1.

[1] Bergmann/Klefsjö (1994), p.15 in:Leitner/Valastiak (2009), p.1

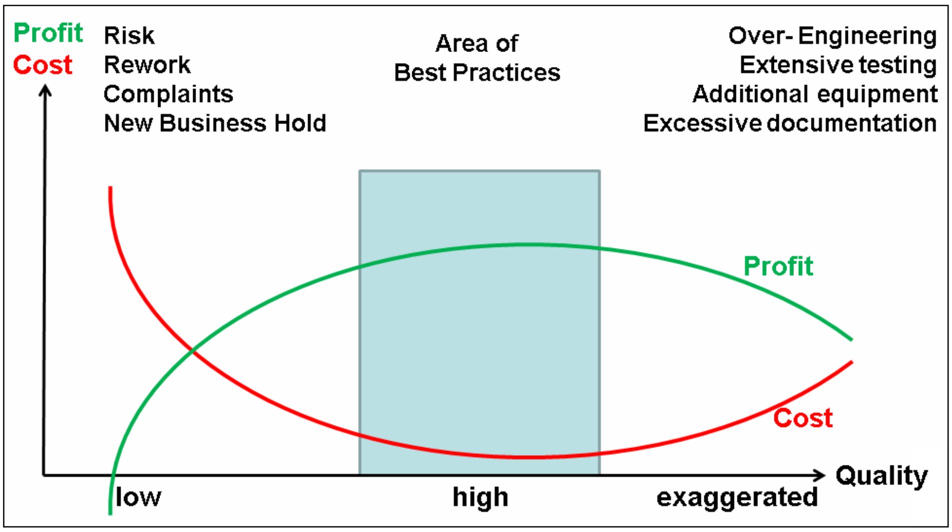

Figure 1: Area of Best Practices

As a synthesis of the above mentioned inputs the model will suggest a structure to support engineering to fulfil the requirements of current standards and keep the effort minimal at the same time. Because the initial situation may differ significantly the model has to be adapted to specific needs before it can be implemented in existing organisations. As the engineering sites analysed in the case studies have already systems in place another option is to select specific aspects of the business model to reach improvements faster or to reduce effort. This is shown in Figure 2.

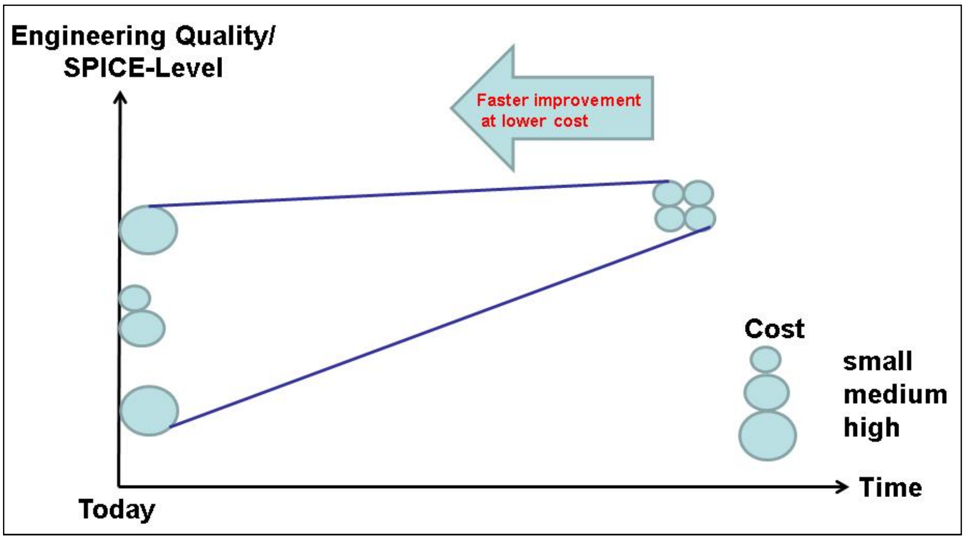

Figure 2: Quality of engineering sites

1.2 About MAGNA[2]

MAGNA is the most diversified automotive supplier in the world. It designs, develops and manufactures automotive systems, assemblies, modules and components, and engineers and assembles complete vehicles, primarily for sale to original equipment manufacturers (OEMs) of cars and light trucks globally.

The capabilities of MAGNA include the design, engineering, testing and manufacture of automotive interior systems; seating systems; closure systems; metal body & chassis systems; mirror systems; exterior systems; roof systems; electronic systems; powertrain systems as well as complete vehicle engineering and assembly, see Figure 3.

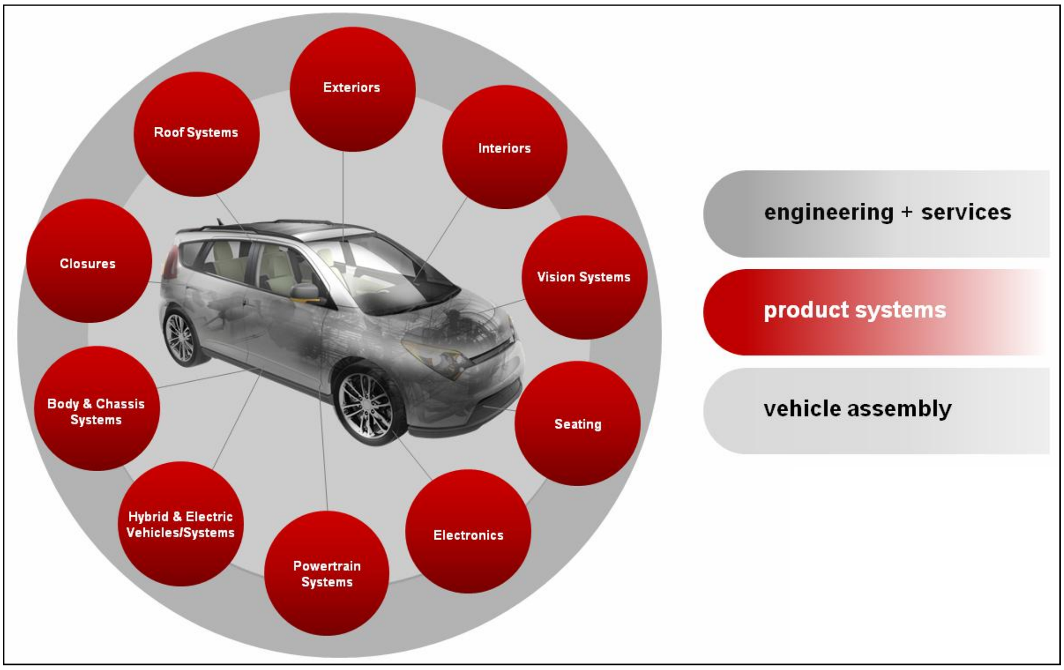

Figure 3: MAGNA's global capabilities[3]

MAGNA has 245 manufacturing operations and 80 product development, engineering and sales centers in 25 countries on five continents as of September 2010.

With about 20 billion US$ sales MAGNA is one of biggest suppliers in automotive industry. It is based on entrepreneurial spirit which is also influencing its operational structure, see Figure 4. The executive core team,

[2] Cf. http://www.MAGNA.com/MAGNA/en/about/ 31.01.2011

[3] MAGNA International (2011), p.3

consisting of the presidents of the single product groups, sets strategic direction for the organization and manages capital expenditure. The full operational responsibility lies within the product groups and here with the general managers. This very decentralized structure is seen as one of the key success factors for MAGNA.

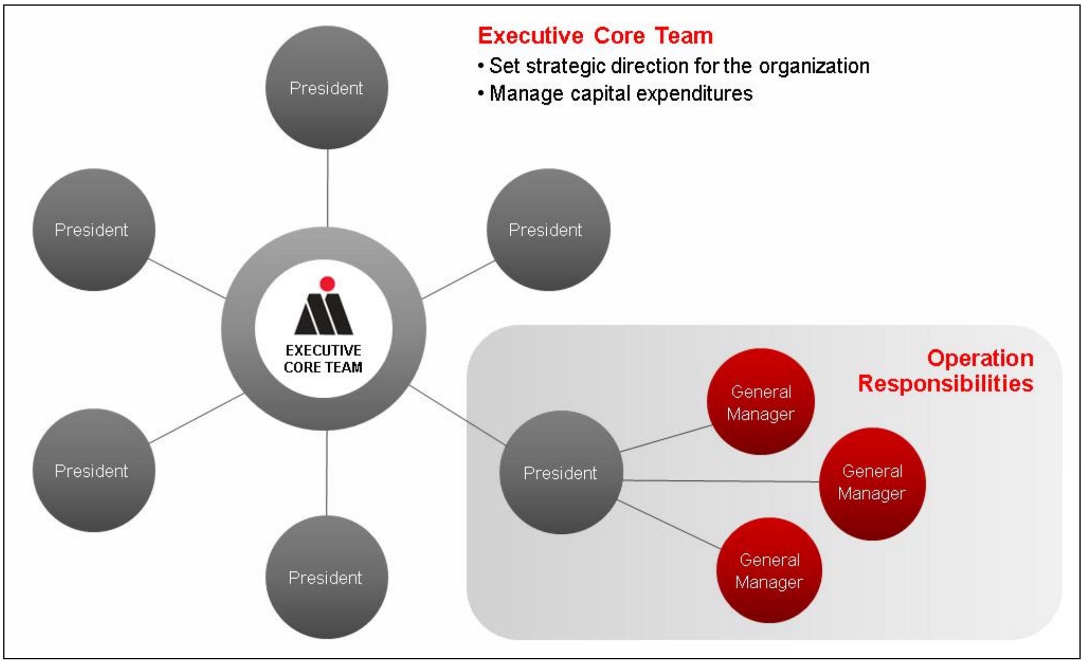

Figure 4: MAGNA's operational structure[4]

In contrast to competitors with centralized functions this structure is the reason why one can find different approaches to common tasks within MAGNA. One example for different solutions in the area of engineering is requirements management which will be described in detail in Chapter 3 Best Practices for Requirements Engineering at MAGNA. This Chapter describes the implemented requirements management system and five different case studies of three different MAGNA groups.

[4] MAGNA International (2011), p.5

2 Quality Management in Automotive Business

If people talk about quality everybody has his own perception of what quality means. Even in literature there is a huge number of definitions emphasizing different aspects. As the customers in automotive business are usually companies and not end consumers a technical oriented definition of quality is suitable The following definition of quality can be found by looking at the most common quality standard globally, the ISO 9000 series:

'Quality is the degree to which a set of inherent characteristics fulfils requirements.'[5]

For full understanding of the definition of quality it is necessary to define the term requirement, too:

'Requirement is the need or expectation that is stated, generally implied or obligatory.'[6]

The meaning of these definitions results in a product which is fully compliant with specified and even unspecified requirements if they are common practice or stated by law.[7]

2.1 History of Quality Management

Quality was always important in production. Before industrialization quality was ensured by craftsman controlling their product from beginning to end. With industrialization and mass production working standards were established and quality assured by dedicated staff. Over time the methods for quality control got more sophisticated and statistics was introduced. Statistical process control allowed to reduce cost for quality assurance and included even some failure preventive aspects. Nevertheless quality assurance was still an area creating cost but no value. In the 1980ies a shift from quality assurance to quality management started. It became common to empower people and to integrate quality control activities in the normal production process. Everybody had to take responsibility for his work.[8]

[5] ISO 9000:2005, p.18

[6] ISO 9000:2005, p.19

[7] Cf. ISO 9000:2005, p.19

[8] Cf. Masing (1999), pp.19

One of the most influential persons promoting quality management was Mr. J. Juran who developed a model called Juran-Trilogy based on the idea to convert business goals into results. This is to be reached through application of managerial processes in three areas, see also Figure 5:

- **Quality planning**[9]

During quality planning the customers and their requirements are determined and the products developed accordingly. Then the production process is designed and responsibility is handed over to operations.

- **Quality control**[10]

Operations has to take care that production is kept stable. This could be supported by usage of methods like statistical process control.

- **Quality improvement**[11]

During quality improvement the target is to change production in a way that a new status quo can be set. At the same time Lessons Learned should flow back to planning, to improve future products.

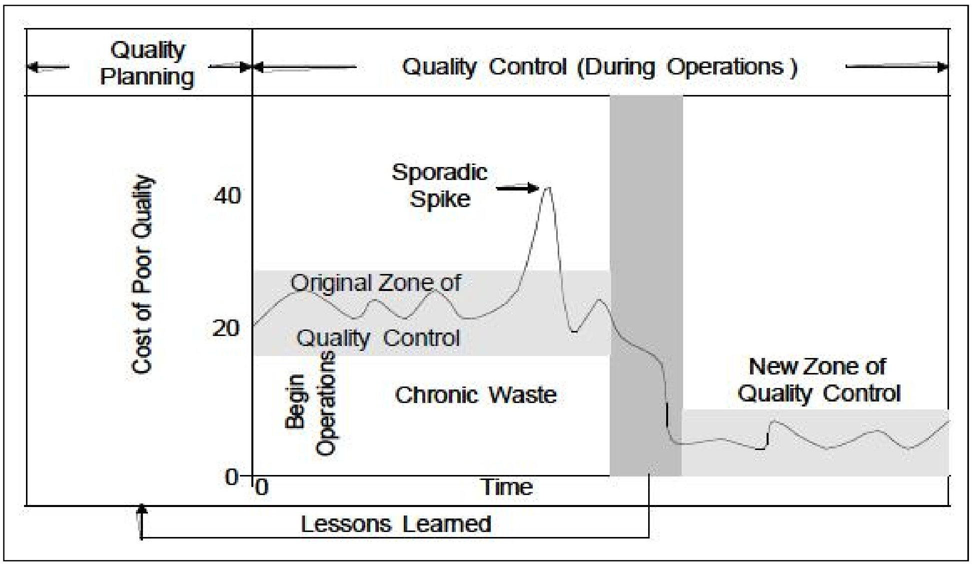

Figure 5: The Juran Trilogy diagram[12]

[9] Cf. Juran (2000), p.3.2

[10] Cf. Juran (2000), p.4.2

[11] Cf. Juran (2000), p.5.3

[12] Juran (2000), p.4.2

2.2 Quality Management in Automotive Industry

At the beginning of car manufacturing a huge number of independent manufacturers existed. Over time these companies either grew tremendous – like Ford –, got big players due to mergers and acquisitions – like General Motors – or faded away. These big companies developed their own quality standards. Over the years automotive suppliers had to meet various standards which were sometimes even conflicting. It got a problem for suppliers of different OEMs to follow different quality requirements within the same plants. In recognition to this situation the big three US companies Ford, General Motors and Chrysler implemented a task force for the creation of a common quality standard. This standard was valid for all suppliers of them and was released under the name QS 9000 the first time in 1994.[13]
The situation in Europe was similar, but at first the OEMs could not agree on one solution. Several standards for geographical and political differing regions were created. The German manufacturers established a joint standard called VDA 6.1 (Verband der Deutschen Automobilindustrie) which was already based on ISO 9001 (International Organization for Standardization).The French automotive industry as well as the Italian automotive industry developed own standards. Although all of these standards are very similar, the standards are not identical.[14]
It was soon recognized that a further harmonization could raise efficiency and cut cost. In 1996 the International Automotive Task Force was established to develop the first global accepted automotive quality standard. This standard was called ISO/TS (Technical Specification) 16949 and includes the whole ISO 9001 and automotive amendments. It was first released 1999 and revised twice in 2002 and 2009. Despite the global acceptance of this new standard there are still customer specific standards applicable, too.[15]
In current automotive projects it is not enough to have a quality management system established which fulfills the requirements of standards. Customers demand a zero defect philosophy and nearly defect free deliveries of products The quality is measured in PPM (Parts Per Million), which means that just a few out of a million products should be defective. This target can

[13] Cf. Leitner/Valastiak (2009), p.78

[14] Cf. Leitner/Valastiak (2009), p.79

[15] Cf. http://www.vda-qmc.de/zertifizierung/iatf, 26.02.2011

only be reached if quality is not assured by inspection but engeneered into the product.[16]

An overview about the main standards and their application timeline is shown in Figure 6.

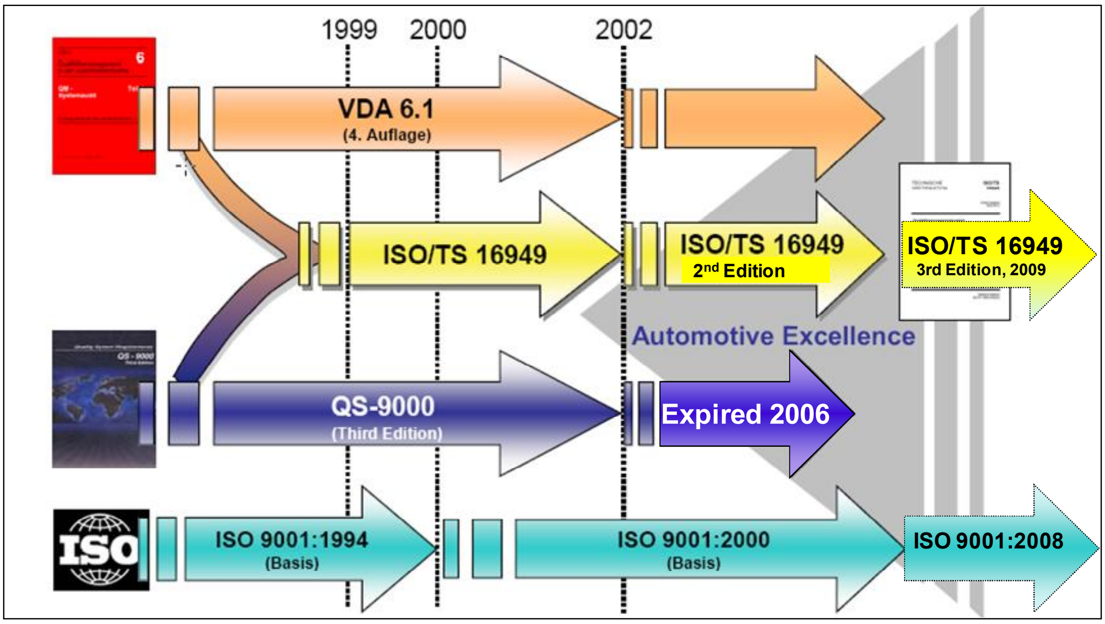

Figure 6: Overview about main automotive standards

ISO 9001 and ISO/TS 16949 require a process oriented quality management system, see Figure 7. Process orientation was first introduced in ISO 9001:2000. A process is an activity using resources and is managed to enable the transformation of inputs into outputs. In order to reach this in a suitable way the model emphasizes the importance of:[17]

- Understanding customer requirements and meeting them
- Establishment of processes for product realization
- Monitoring of process performance and customer satisfaction
- Implementation of continual improvement based on measurements
- Management responsibility for implementation and sustaining of the quality management system
- Resource management for efficient process performance

[16] Cf. Danzer (1995), pp.25

[17] Cf. ISO 9001:2008, pp6

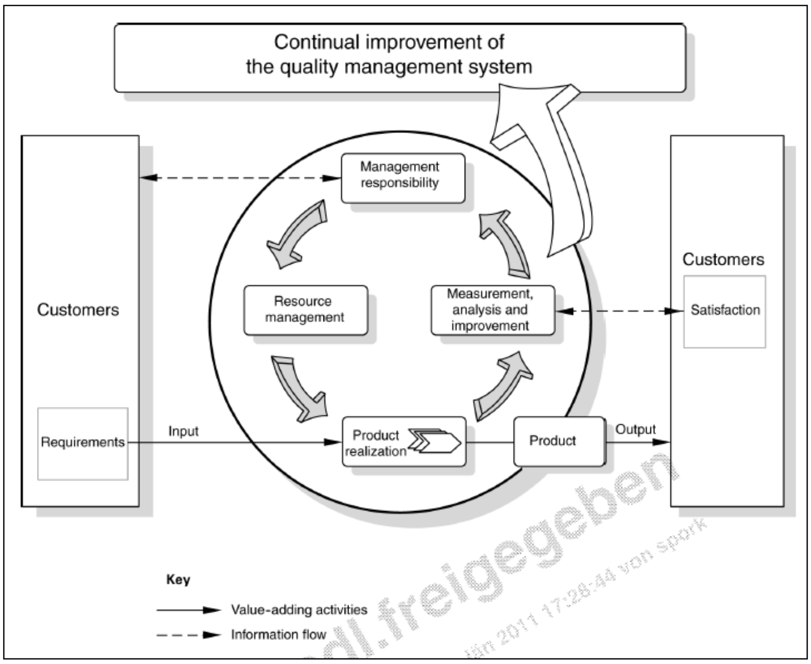

Figure 7: Model of a process-based quality management system[18]

Although the scope of a quality management system according ISO/TS 16949 is designing, developing, producing, installing and servicing of automotive related products development is only covered marginal. In contrast to ISO/TS 16949 some recently released standards are mainly dedicated to engineering.

2.3 New Requirements Regarding Quality Management in Automotive Engineering

During the last decade passenger vehicles changed from more or less mechanical products offering transportation of people and goods to mechatronic systems providing a huge variety of functions. This trend is also reflected in the shift of manufacturing cost as shown in Figure 8.

[18] ISO/TS 16949:2009, p.xi

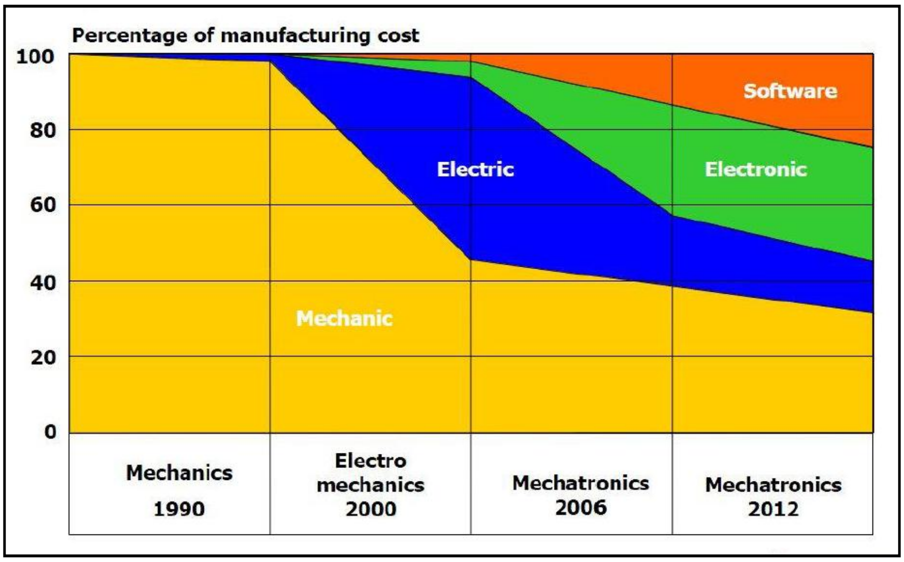

Figure 8: Percentage of manufacturing cost for passenger vehicles[19]

After a short but steep increase of electric components a significant move towards software controlled functionalities started a few years ago and is still ongoing. This change is even more dramatic in engineering as software usually is connected to development cost but not to manufacturing cost.

The basic design for passenger cars is stable for approximately 100 years. Of course a lot of engineering effort was put in the development of cars, leading to safer, more reliable and cheaper products. Considering the overall development effort it is said that there is no other product as well engineered as automobiles. Anyway the cars stayed to be mechanical systems approximately until 2000. Integration of electronics into passenger vehicles got possible because innovations in computer technology supported the mass production of cheap and reliable electronic parts. Since then a competition of innovation started. More and more functions were implemented in cars, many of them totally new but some also replacing former mechanical systems. Today an average car in Europe or North America has its multifunctional display delivering information about the condition of the car and its environment and offers a variety of media or sometimes even an entertainment center including a navigation systems. It is now common to have a medium double digit number of ECUs (Electronic Control Unit) even in cheap cars. These ECUs are interlinked via bus

[19] MAGNA Powertrain, Source:Audi

systems to each other and sensors thus leading to possibly infinite number of different system conditions and interactions.[20]

New, innovative functionalities are important in competition but increase the complexity of the product and the development process tremendous. As the use of software offers further potential for innovation this trend is expected to persist[21], see Figure 9.

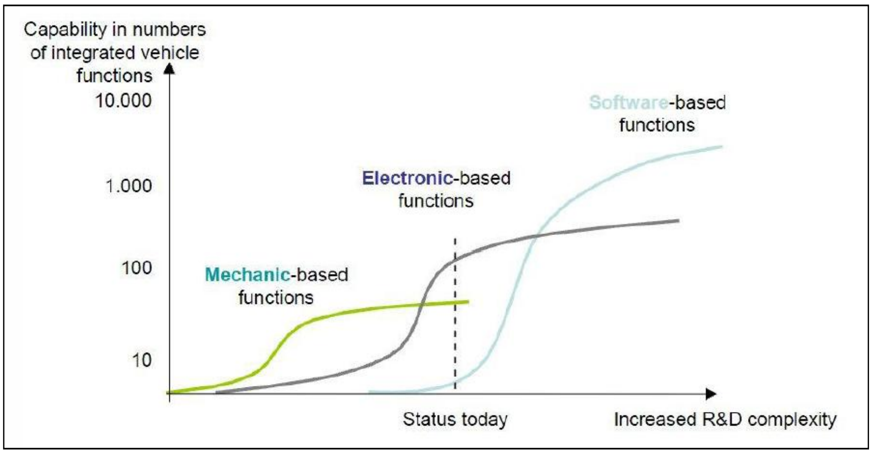

Figure 9: Increase of functionality in passenger vehicles[22]

A major driver for innovation is vehicle safety. According to the lecture vehicle safety about one third of the weight of a modern car is dedicated to safety.[23]

Besides this significant portion of mechanical parts for safety many new mechatronic functions are related to safety, too. Some examples can be found in Figure 10. A function like 'Adaptive Cruise Control' is extremely complex and distributed over several systems delivered by different suppliers.[24]

[20] Cf. Messnarz (2007), pp.5

[21] Cf. http://portal.automotive-his.de/index.php?option=com_content&task=view&id=7&Itemid=18, 26.02.2011

[22] MAGNA Powertrain

[23] Steffan (2009), pp.7

[24] Cf. Kiedaisch et al. (2001), pp.10.1

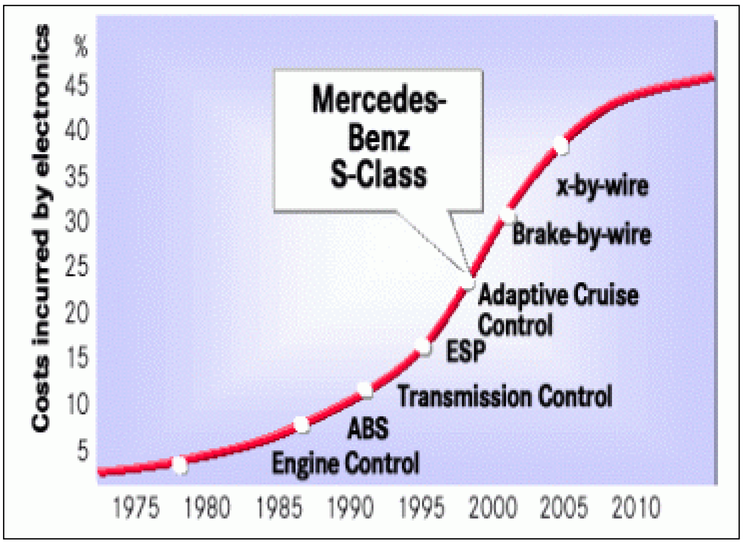

Figure 10: Innovative safety related mechatronic functionality[25]

Another trend in automotive projects is shortening of development times. This leads to a situation in which the customer requirements sometimes are not fully clarified at project start. The situation can even be worse in highly innovative projects in which neither the customer nor the supplier have a lot of experience. The stability of requirements during the first three month of automotive projects is shown in Figure 11[26].

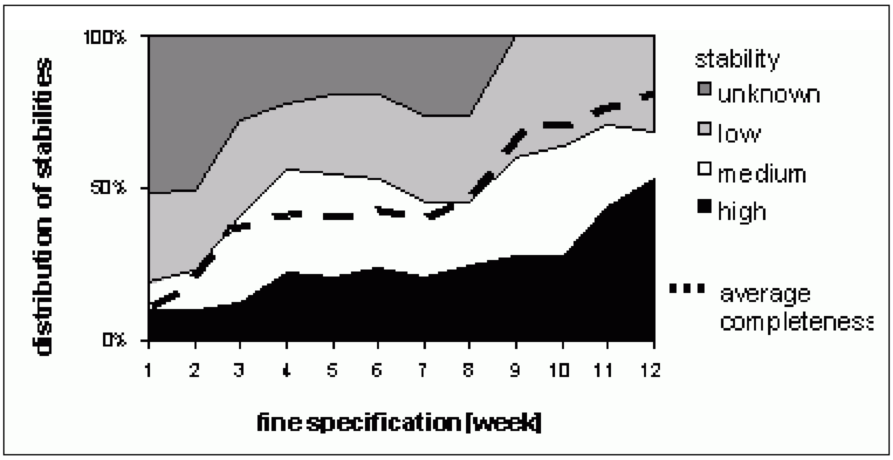

Figure 11: Stability of requirements during project start[27]

If the customer specification is not stable from the beginning and requirements change or new requirements emerge it is burdensome to keep

[25] Kiedaisch et al. (2001), p.10.2

[26] Cf. Kiedaisch et al. (2001), pp.10.9

[27] Kiedaisch et al. (2001), p.10.9

the system specification up to date. Any customer requirement change has to checked on its impact regarding technical and financial feasibility and the system specification has to be adapted if the change is accepted. In such an environment consequent requirements engineering is of utmost importance to avoid any misunderstanding both internal and to the customer. Any premature promise to the customer may later lead to project delays or additional cost for the product.[28]

Therefore the OEMs have identified engineering quality as key characteristic for success. In contrast to simple mechanical products it is not feasible to ensure engineering quality of such systems only with testing. Thus it is essential to provide suitable development processes and supporting activities. This led to automotive adaptations of ISO standards dealing with the development of mechatronic systems and their rigor enforcement. One standard, Automotive SPICE, is mainly dealing with processes during product development a second standard, ISO/DIS 26262, is dealing with the development of safety critical functions. Both standards emphasize that system characteristics are most important and that only an integrated development approach of all faculties will deliver the needed engineering quality.[29]

For many mechanical oriented suppliers this led to a paradigm shift where the traditional quality management was not enough and the whole engineering process had to be reengineered accordingly, see also Figure 12.

[28] Cf. Höhn et.al. (2009), p.119

[29] Cf. Ebert/Lederer (2008), pp.20

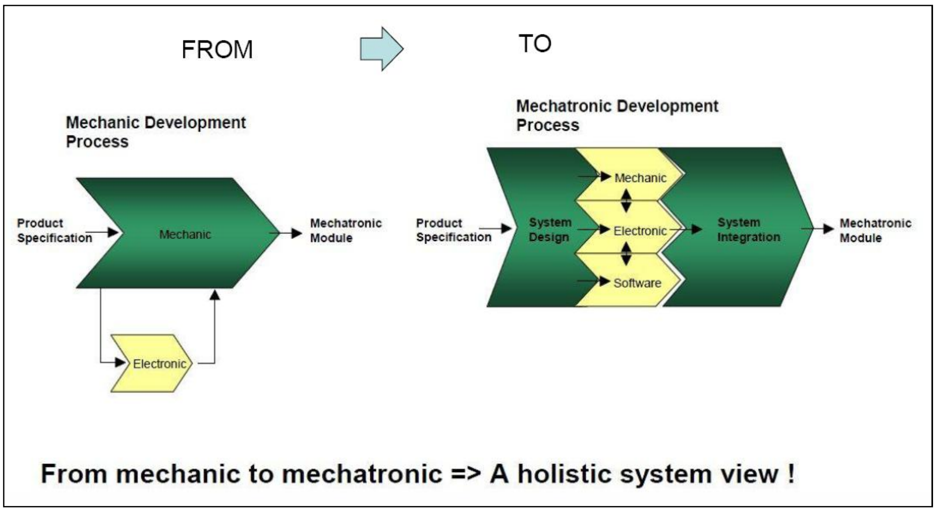

Figure 12: Paradigm shift in development[30]

2.3.1 Automotive SPICE

To cover all aspects of engineering of this mechatronic systems the generic standards like ISO 9001 or ISO/TS 16949 are not sufficient. The US department of defense recognized this early and promoted a standard called Capability Maturity Model (CMM), which was deployed further to Capability Maturity Model Integrated (CMMI). These standards were also taken on by American automotive industry but not enforced consequently in the whole automotive supply base. In Europe a project called Bootstrap with similar intention was founded by EU but it was closed with the publication of ISO 15504, which was at the same time the final version of Bootstrap, Figure 13. ISO 15504 is usually called SPICE (Software Process Improvement and Capability dEtermination), indicating its origin in software development and emphasizing not only the target of compliance but also the target of improvement. These standards were engineering specific amendments to the generic standards like ISO 9001 and had not the intention to replace them.[31]

[30] MAGNA Powertrain

[31] Cf. Müller (2007), pp.2

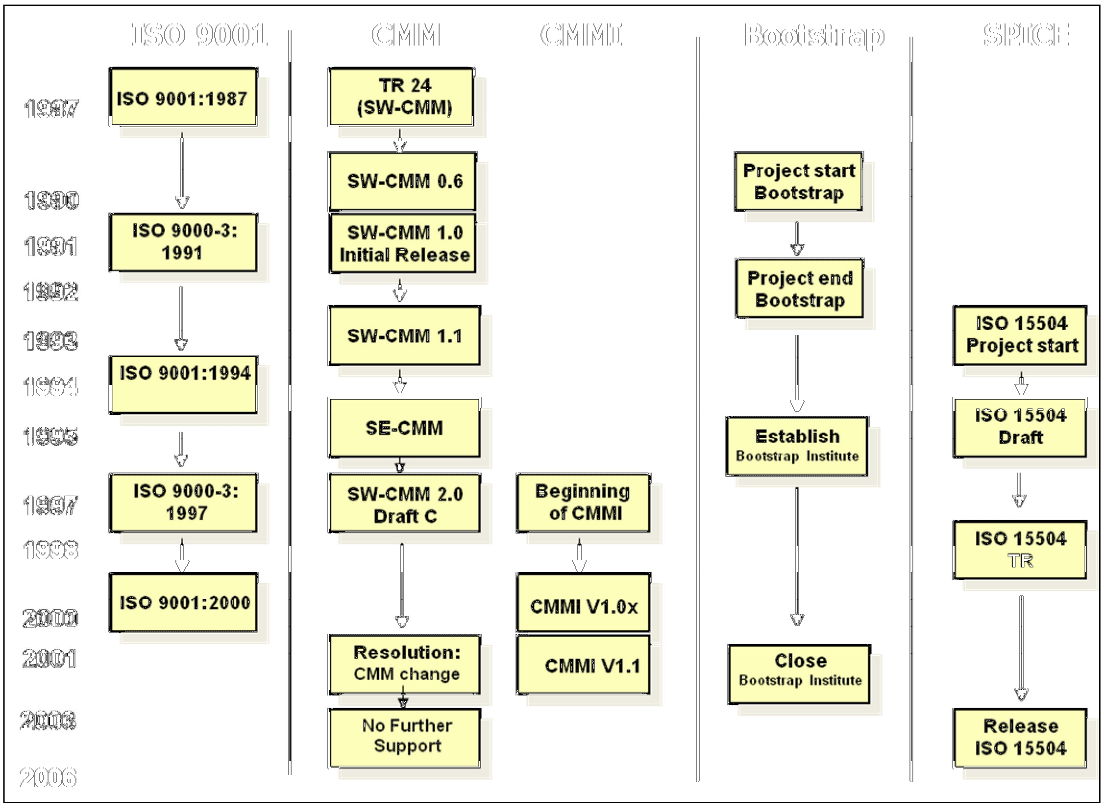

Figure 13: History of standards[32]

Automotive SPICE was deployed based on ISO 15504 by the Automotive Special Interest Group, an initiative consisting of Audi AG, BMW Group, Daimler AG, Fiat Auto S.p.A., Ford Werke GmbH, Jaguar, Land Rover, Dr.Ing.h.c. F. Porsche AG, Volkswagen AG and Volvo Car Corporation, see Figure 14. The first draft was published 2005, the final release was 2008 as VDA standard. The big German OEMs Audi AG, BMW Group, Daimler AG, Porsche AG and VW AG constituted an additional group called HIS (Hersteller Initiative Software). SPICE and later Automotive SPICE is a mandatory requirement by HIS for mechatronic suppliers since about 2005 leading to a fast dissemination in their supply base.[33]

[32] Methodpark (2009), Standard overview, p.3

[33] Cf. http://automotivespice.com/web/Introduction.html, 17.01.2011

Figure 14: Members of the Automotive Special Interest Group[34]

ISO 15504 consists currently of seven parts, but only part 2 is normative, see Figure 15. New parts are work in progress and part 10, dealing with safety management is in the final ballot and expected to be released soon[35].

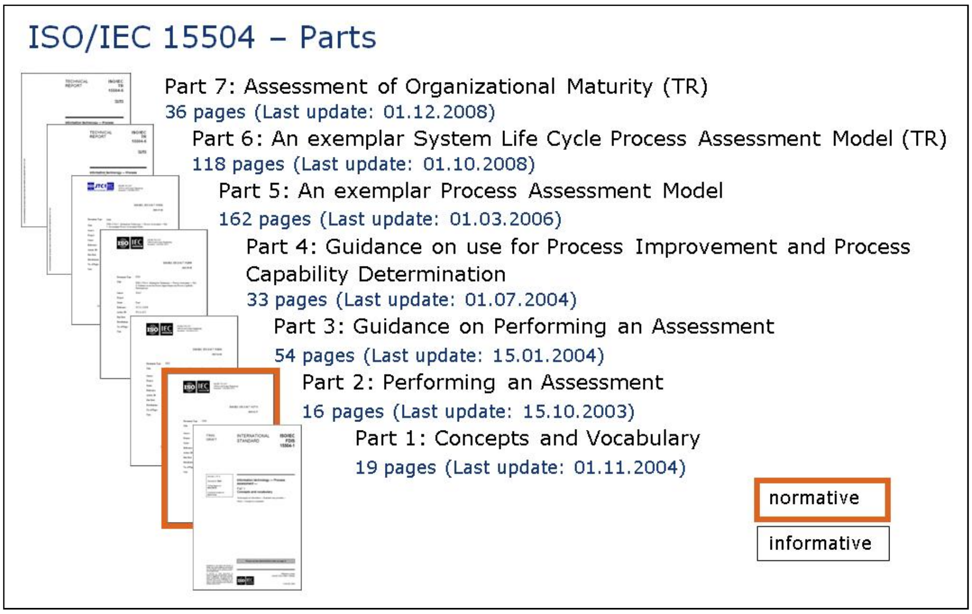

Figure 15: Structure of ISO/IEC 15504[36]

Part 2 defines how to perform an assessment, how capability is measured and which mandatory requirements have to be fulfilled by an Process Reference Model (PRM). Only the combination of the PRM and Measurement Framework lead to a Process Assessment Model (PAM) which is the base of each assessment. The standard includes just an exemplar PAM as part 5 and is open for new domains to deploy their specific PRMs, see Figure 16.[37]

---

[34] http://automotivespice.com/web/Introduction.html, 17.01.2011

[35] Cf. http://www.spiceusergroup.org/forum2/topics/isoiec-1550410-safety, 17.01.2011

[36] Methodpark (2009), Standard overview, p.4

[3737] Cf. Hörmann et.al. (2006), p.17

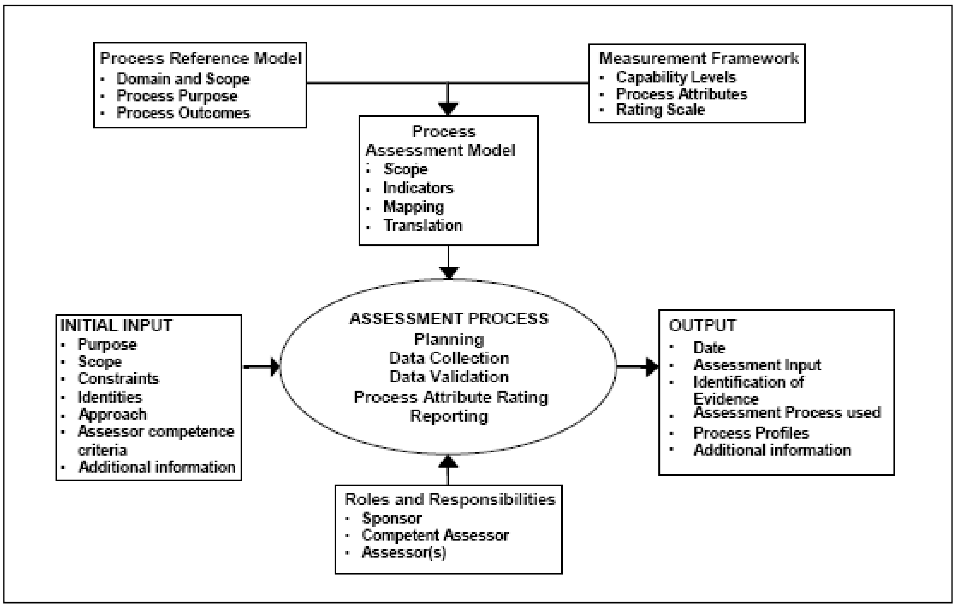

Figure 16: Normative elements of ISO/IEC 15504[38]

'The Process Assessment Model is a two-dimensional model of process capability. In one dimension, the process dimension, the processes are defined and classified into process categories. In the other dimension, the capability dimension, a set of process attributes grouped into capability levels is defined. The process attributes provide the measurable characteristics of process capability.'[39] The two dimensions and the indicators for process capability are shown in Figure 17.

[38] ISO/IEC 15504-2:2003, p.2

[39] ISO/IEC 15504-5:2006, p.2

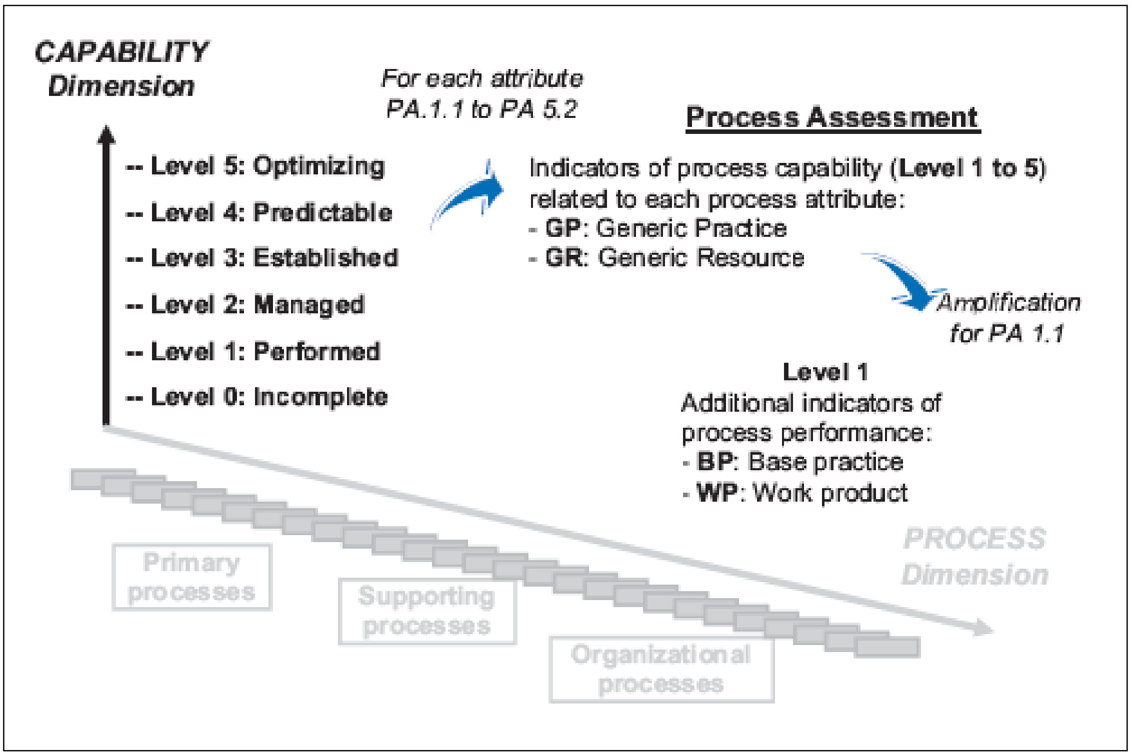

Figure 17: Assessment indicators[40]

Specific Base Practices (BP) are needed to produce the expected outcomes for each process and used to rate Capability Level 1 (CL 1), higher levels are rated with process independent Generic Practices (GP) and Generic Resources (GR). They are defined in the PRM. As different domains may establish different PRMs Automotive SPICE is one of them. In contrast to the exemplar PRM in ISO 15504 Automotive SPICE does not have additional Generic Work Products. Figure 18 shows this concept in a simpler way. The abscissa is used to represent the selected processes from the applicable PRM. The domain is given by the business, the scope which is the number of selected processes has to be defined. The Measurement Framework defines the number of CLs and how single BPs and GPs have to be rated, how this rating is condensed to Process Attributes (PA) and finally how CLs are derived.[41]

[40] Automotive SPICE, p.22

[41] Cf. Höhn (2009), pp.7

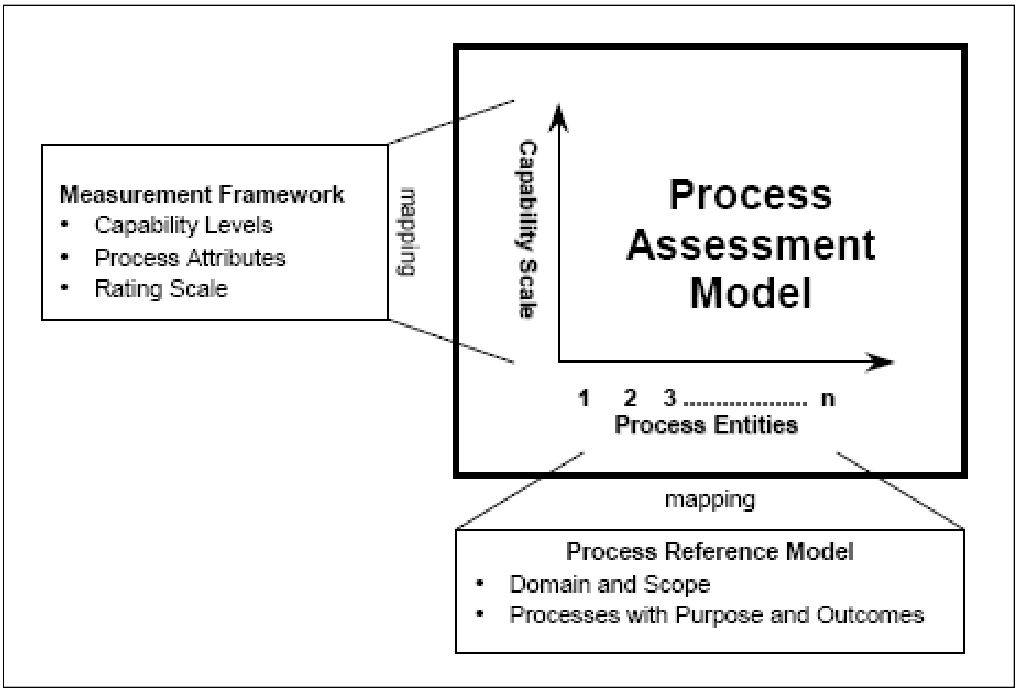

Figure 18: Process Assessment Model relationships[42]

The Measurement Framework of SPICE defines a six point ordinal scale called Capability Levels. It starts with Capability Level 0, incomplete, and reaches up to Capability Level 5, optimizing. Whereas Capability Level 0 has no Process Attribute and Capability Level 1 has just one Process Attribute, Capability Level 2 to 5 have two Process Attributes each, see Figure 19. The single CLs and their meaning are[43]:

Level 0: Incomplete process

The process is not implemented, or fails to achieve its process purpose.
At this level there is little or no evidence of any systematic achievement of the process purpose.

Level 1: Performed process

The implemented process achieves its process purpose.

Level 2: Managed process

The previously described Performed process is now implemented in a managed fashion (planned, monitored and adjusted) and its work products are appropriately established, controlled and maintained.

Level 3: Established process

[42] ISO/IEC 15504-2:2003, p.14

[43] Cf ISO/IEC 15504-2:2003, pp. 6

The previously described Managed process is now implemented using a defined process that is capable of achieving its process outcomes.

Level 4: Predictable process

The previously described Established process now operates within defined limits to achieve its process outcomes.

Level 5: Optimizing process

The previously described Predictable process is continuously improved to meet relevant current and projected business goals.

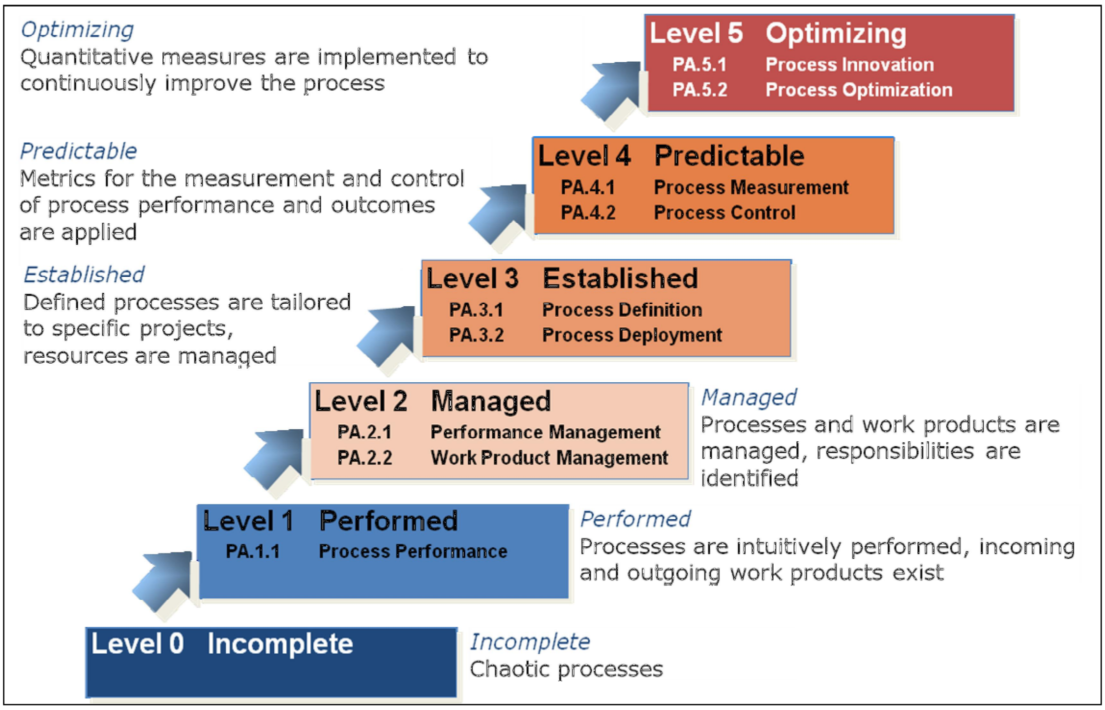

Figure 19: Capability levels[44]

The rating is made according an ordinal rating scale with four different rating points and shall be understood in terms of percentage:[45]

Not achieved (N): 0 to 15 % achievement

There is little or no evidence of achievement of the defined attribute in the assessed process.

Partially achieved (P): > 15 % to 50 % achievement

There is some evidence of an approach to, and some achievement of, the defined attribute in the assessed process. Some aspects of achievement of the attribute may be unpredictable.

---

[44] Methodpark (2009), Capability dimension, p.6

[45] Cf ISO/IEC 15504-2:2003, pp.10

Largely achieved (L): > 50 % to 85% achievement

There is evidence of a systematic approach to, and significant achievement of, the defined attribute in the assessed process. Some weakness related to this attribute may exist in the assessed process.

Fully achieved (F): > 85 % to 100 % achievement

There is evidence of a complete and systematic approach to, and full achievement of, the defined attribute in the assessed process. No significant weaknesses related to this attribute exist in the assessed process.

As the process itself can not be evaluated during an assessment the result is based upon the rating of indicators of process performance. The accomplishment of Base Practices and Generic Practicess is used as indicator. The determination of Capability Levels is based on the level of achievement of the single Base Practices and Generic Practices, each of them getting a separate rating. These ratings are condensed to one rating per Process Attribute. To reach a Capability Level it is necessary that all Process Attributes of the lower Capability Levels are rated fully achieved and the Process Attributes of the highest Capability Level are rated largely achieved at least.[46]

If you apply this rule on the example of an assessment output in Figure 20 it leads to the following results:

F.1.3.1 Requirements Elicitation: CL 2

F.1.3.3 System and Architectural Design: CL 3

F.2.2 Configuration Management: CL 1

F.3.1.4 Risk Management: CL 0

F.1.1.2 Supplier Selection: CL 1

[46] Cf. ISO/IEC 15504-4:2004, pp.4

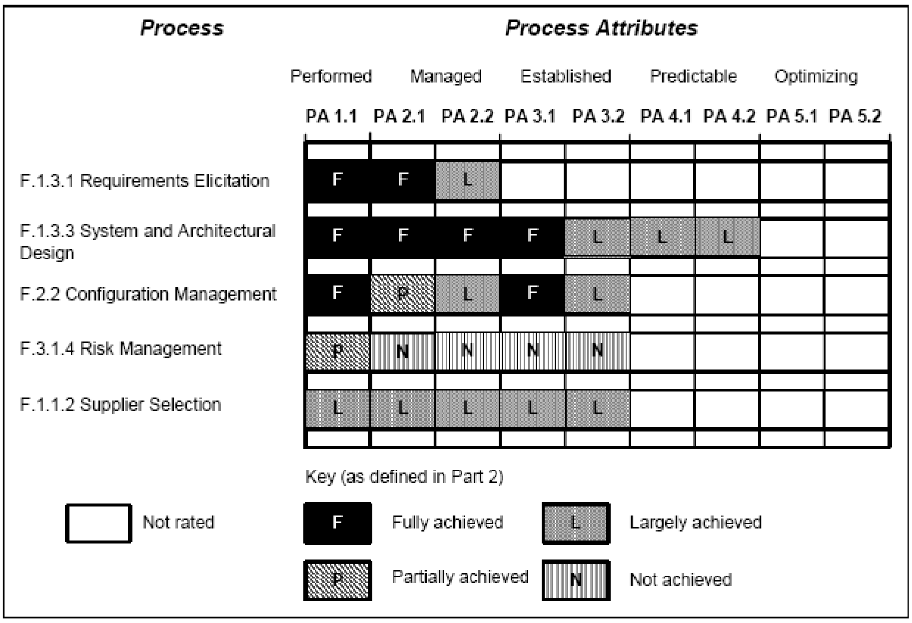

Figure 20: Example assessment output set of process profiles[47]

For the rating of Capability Level 1 the process specific Base Practices are used. On higher levels there is an interaction of process performance on Capability Level 1 of specific processes and the rating of some Process Attributes. E.g. it is not possible to get Process Attribute 2.1 ‘Performance Management’ rated fully if there are significant weaknesses in project management on level 1. This means that all other processes can be limited in their results, if just one process fails on level 1. On the other hand other processes are supported at level 2 by a good rating of project management at level 1. The relationship between Process Attributes and corresponding processes is shown in Table 1.

[47] ISO/IEC 15504-4:2004, p.4

Related processes	Process attributes PA 2.1	PA 2.2	PA 3.1	PA 3.2	PA 4.1	PA 4.2	PA 5.1	PA 5.2
SUP.1 Quality assurance	◆	◆						
SUP.2 Verification		◆						
SUP.4 Joint review	◆	◆						
SUP.5 Audit			◆	◆				
SUP.7 Documentation		◆						
SUP.8 Configuration management		◆						
SUP.9 Problem resolution management	◆	◆						
SUP.10 Change request management		◆				◆		
MAN.1 Organizational alignment						◆	◆	◆
MAN.2 Organization management			◆	◆				
MAN.3 Project management	◆			◆				
MAN.4 Quality management				◆	◆	◆		
MAN.5 Risk management	◆				◆			
MAN.6 Measurement				◆	◆	◆	◆	
PIM.1 Process establishment			◆					
PIM.2 Process assessment				◆	◆	◆	◆	
PIM.3 Process improvement				◆			◆	◆
RIN.1 Human resource management	◆		◆	◆				
RIN.2 Training				◆				
RIN.3 Knowledge management			◆				◆	
RIN.4 Infrastructure			◆	◆		◆		
REU.1 Asset management			◆					◆
REU.3 Domain engineering			◆					◆

Table 1: Related processes for Process Attributes[48]

The duration of the assessment of a project with HIS scope level 2 is four full working days. This applies for one single project which is under development. In comparison to an ISO 9001 audit this would be the time needed for an certification audit of an unit with approximately 1000 employees and would cover all areas from acquisition, development, purchasing, production, logistics and all organizational activities.

Such an assessment result is different to the result of an audit in the traditional quality management like ISO 9001. It shows very precise strengths and weaknesses of the organization by differentiating Capability Levels for different processes. ISO 9001 certification audits are checking for compliance with the standard and either confirm full compliance or do not certify anything. As nearly all producing companies supplying to industrial customers in Europe or North America are certified according 9001 this would mean that all these companies are strictly following processes or that an ISO 9001 certification is of limited value. Current results of Automotive SPICE assessments in companies which are ISO certified for many years include processes with Capability Levels 0 on a regular basis, sometimes even for leading Tier 1 system suppliers based in central Europe.

The exemplar PRM of ISO 15504 includes many processes which were not taken over to Automotive SPICE. An overview of all processes of ISO 15504

[48] ISO/IEC 15504-5:2006, p.97

is shown in Figure 21. The processes are categorized according their contribution to the life cycle of a project and further grouped within this categories.[49]

All 31 processes included in Automotive SPICE are marked with A, 10 out of them being in the engineering process group making engineering to the main focus area. 15 processes are additionally marked with H, indicating that they belong to the HIS scope, which is the most common scope for Automotive SPICE assessments today.

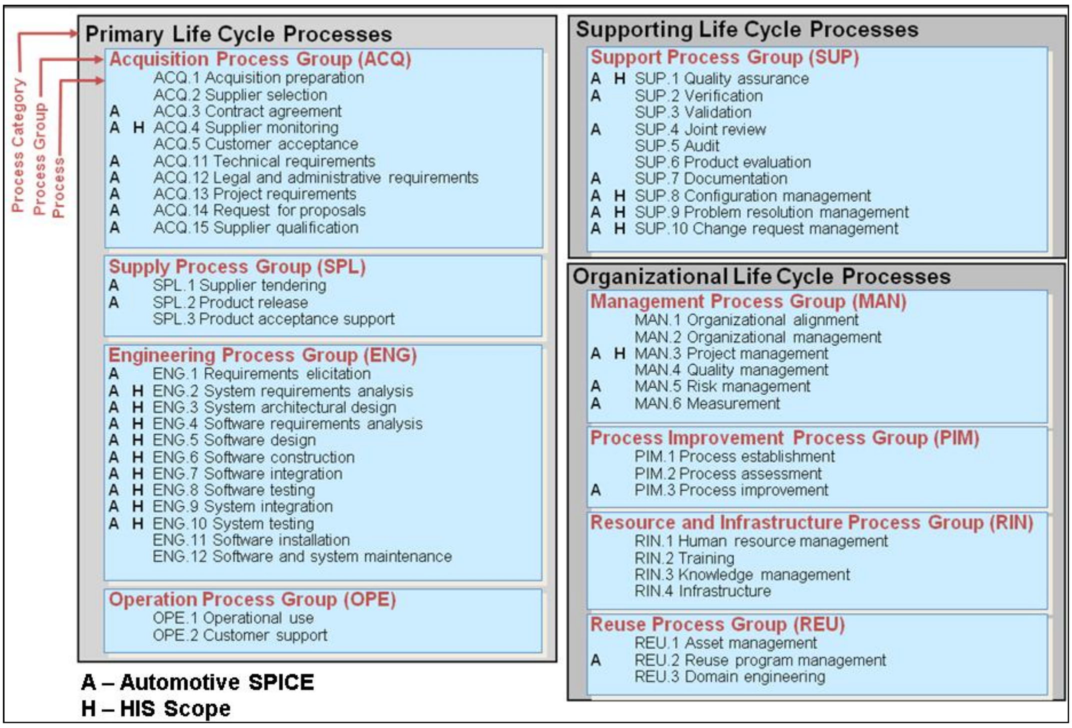

Figure 21: SPICE processes[50]

Automotive SPICE is one PRM fulfilling the requirements of ISO 15504 and therefore can be used within a PAM including the domain specific content. The structure to establish a valid PAM is shown in Figure 22.

[49] Cf. ISO 15504-5:2006, p.4

[50] Methodpark (2009), Relationship, p.40

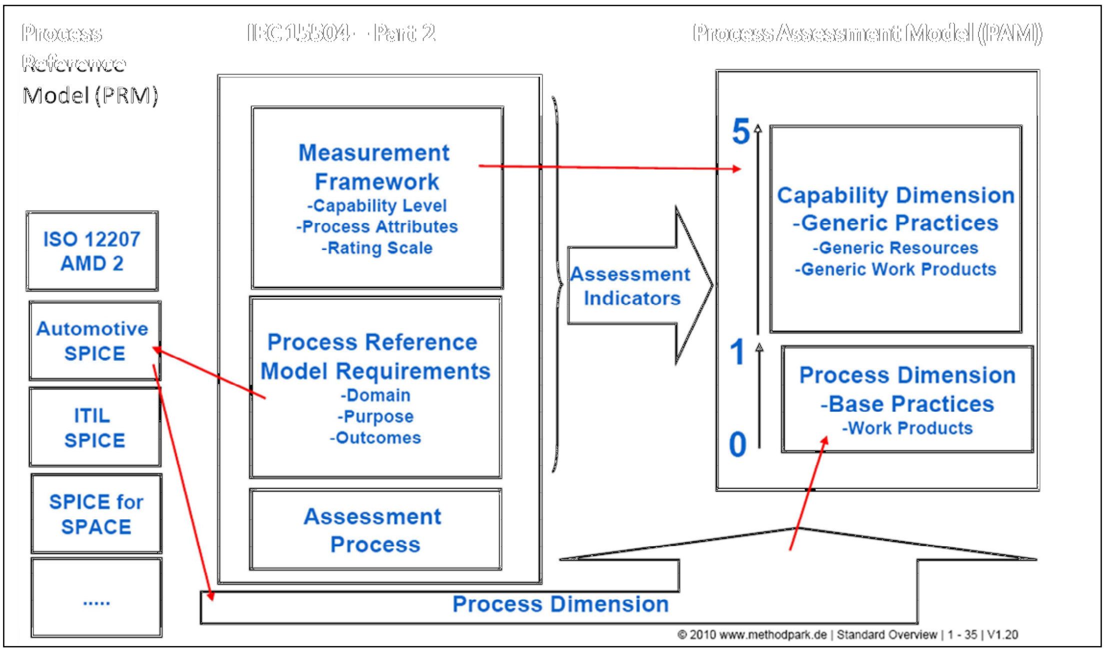

Figure 22: Structure to establish a Process Assessment Model[51]

As already mentioned the origin of SPICE is in the area of software development. Nowadays the focus is no longer on software alone but also on systems which are controlled with software. This can be seen in ISO 15504 by the recent release of part 6 dealing with systems and also in Automotive SPICE. The subtitle of it is '*Prozessbewertung gemäß Automotive SPICE in der Entwicklung von softwarebestimmten Systemen*'[52] which means process capability determination according Automotive SPICE during development of software controlled systems. The emphasis on the system level in Automotive SPICE can be demonstrated best by a close look on the key concept schematic shown in Figure 23.

System requirements are derived from customer requirements and further allocated to the subsystems mechanics, hardware and software each with own requirements. To be able to do this it is necessary to establish a system architectural design which translates system functionality (what a system is intended to do) into a feasible solution (how the system will look like). After the subsystems are developed and tested the system integration process is focusing on interfaces defined in the system architectural design. The verification criteria of each system requirement are then used for the system testing to verify compliance with the system requirements specification.

[51] Methodpark (2010), Standard Overview V1.20 p.1

[52] Automotive SPICE, cover page

Acceptance testing, the final step before market introduction, is usually made by the OEM. It is vehicle testing of the integrated super system (passenger car) in designated environment. For the subsystem software, which is seen as a system itself, the same logic applies again. The software requirements describe the expected functionality of the software system. The high level software design brakes this down to software components which are usually big parts like operational software, application software and interface layers. This parts are finally specified according a low level software design as software units, small elements which are coded and tested. The software units are integrated to software components which are integrated to the software system. During integration the operation of interfaces is checked and the correct implementation of software requirements is then tested on the subsystem level. If the project is a software product only, the software would then be validated by the customer. If the project includes hardware and mechanics the system integration follows.[53]

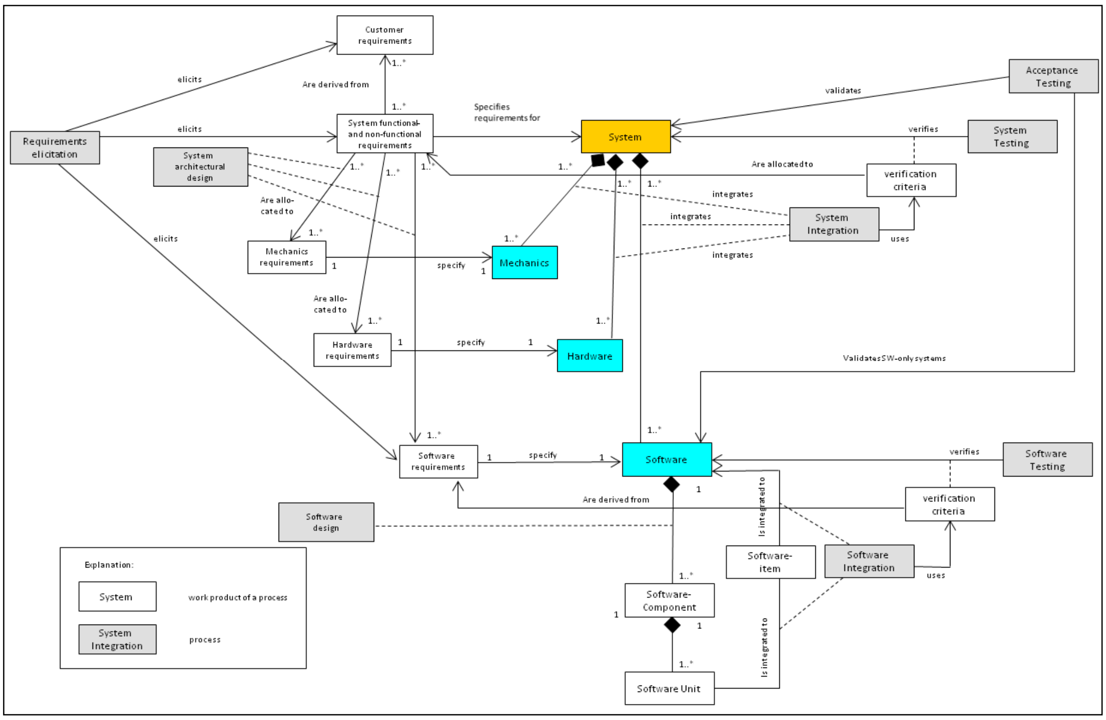

Figure 23: Key concept schematic[54]

According to the definition of ISO/DIS 26262 *'formal verification is the method used to prove the correctness of a system against the formal specification of*

[53] Cf. Müller (2007), p.14

[54] Automotive SPICE, p.180

its required behavior'[55]. This includes all testing activities from single components to system test aiming to prove the fulfillment of the specification. Acceptance testing is used for validation which is the *'confirmation, through the provision of objective evidence, that the requirements for a specific intended use or application have been fulfilled'*[56]. Validation in automotive industry is mainly vehicle testing done with vehicles assembled from already verified integrated systems. In the majority of cases lies the responsibility for verification with the supplier and the responsibility for validation with the OEM.

The key concept describes a logical proceeding including different levels of detail. This is also indicated in the names and numbers of the engineering process group:[57]

- ENG.1 Requirements elicitation
- ENG.2 System requirements analysis
- ENG.3 System architectural design
- ENG.4 Software requirements analysis
- ENG.5 Software design
- ENG.6 Software construction
- ENG.7 Software integration
- ENG.8 Software testing
- ENG.9 System integration
- ENG.10 System testing

A second sketch with essential information regarding requirements management can also be found in Automotive SPICE, see Figure 24. A special requirement of Automotive SPICE is bi-lateral traceability which is an addendum to ISO 15504. Bi-lateral traceability means that one has to be able to find all derived requirements and all related tests for each requirement starting from any level in requirements management. Traceability between requirements and architectural design on system and software level has to be established in addition, showing which components satisfy which requirements. On the other hand requirements for components can also be derived from architectural designs. Bi-lateral means also to be able to find back to the requirement level above until customer requirements are

[55] ISO/DIS 26262-1, p. 7

[56] ISO 9000:2005, p. 31

[57] Automotive SPICE, pp. 49

reached. Even the trace back from software unit test case to respective customer requirements has to be established. Because of bi-lateral traceability the implementation of a requirements management tool is inevitable for all big projects. It is also remarkable that other requirements are mentioned besides customer requirements. This can be legal, normative or internal requirements.

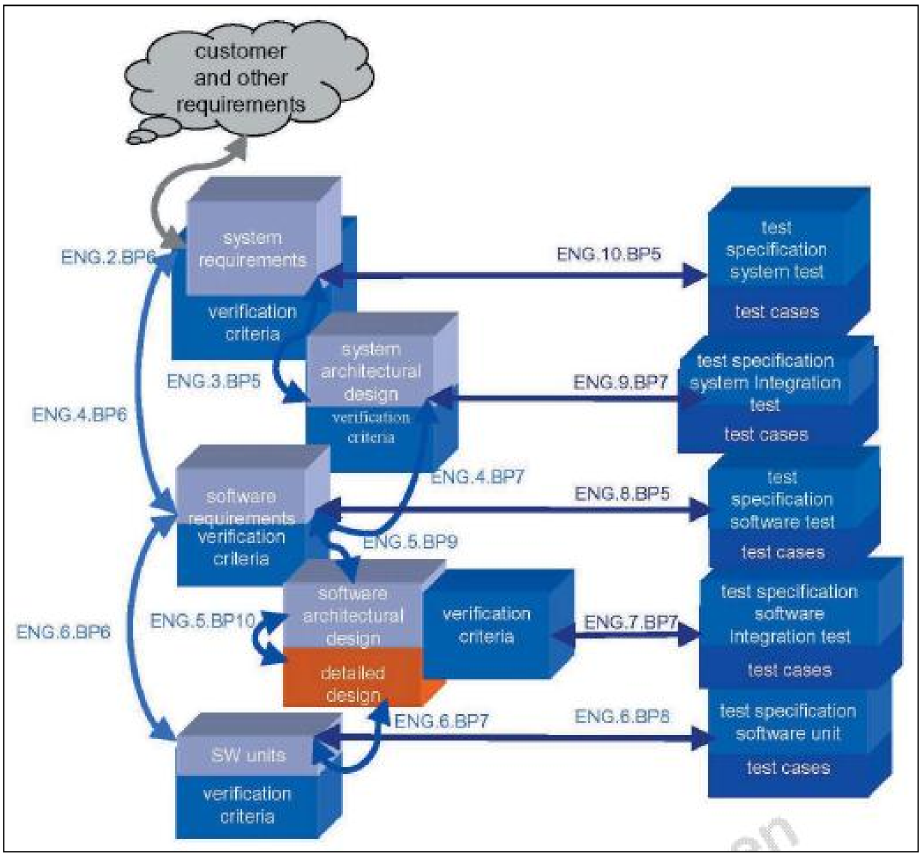

Figure 24: Bi-lateral traceability[58]

Looking at CMM one can see that it was originally a pure software development standard. CMM and SPICE have a similar definition of capability levels from level 2 onwards. Regarding Capability Level 1 the substantial difference is that CMM starts at Capability Level 1 and does not have a Capability Level 0, which means that Capability Level 1 is not comparable between CMM and SPICE. CMM was based on the assumption that engineering is doing well anyway and that most weaknesses can be

[58] Automotive SPICE, p. 182

found in management and supporting processes. Therefore the initial model started with focus on this areas. [59]

CMM is a concept in which the number of relevant processes increases from level to level, see Figure 25. To reach Capability Level 2 there is only one process in the area of engineering included which is requirements management. Major involvement of engineering starts at level 3.

Level	Focus	Process Area
5 Optimizing	*Continuous Improvement*	Process Change Management Technology Change Management Defect Prevention
4 Managed	*Product and Process Quality*	Software Quality Management Quantitative Process Management
3 Defined	*Engineering Process*	Organization Process Focus Organization Process Definition Peer Reviews Training Program Inter-group Coordination Software Product Engineering Integrated Software Management
2 Repeatable	*Basic Management*	Requirements Management Software Project Planning Software Project Tracking and Oversight Software Subcontract Management Software Quality Assurance Software Configuration Management
1 Initial	*Competent people (and heroics)*	

Quality
Productivity

Risk
Waste

Figure 25: Process areas and capability levels of CMM[60]

CMMI enriched the software content with systems engineering, acquisition and topics like training and assessment. Thus the overall content is comparable to SPICE. Also the capability is defined by the same six levels found in SPICE with process specific practices on level 1 and generic practices from level 2 on. CMMI offers two strategies regarding capability determination. One strategy is called staged and extends the number of processes from level to level like CMM does. The other strategy is called continuous and defines a fixed set of selected processes which are assessed starting at CL 1 like in SPICE. All in all the CMMI continuous model comes very close to SPICE.[61]

[59] Cf. Hörmann et.al. (2006), pp.7

[60] Methodpark (2009), Relationship, p.8

[61] Cf. Hörmann et.al. (2006), pp.8

A work group of VDA investigated if a direct mapping of assessment results between CMMI and Automotive SPICE is possible. The comparison was made for the processes of HIS scope and the processes additionally required by Ford, see Figure 26. It was found that CMMI covers most of the organizational processes fully but that not all aspects of engineering processes are covered. For example the bi-lateral traceability in requirements management has lower requirements in CMMI.[62]

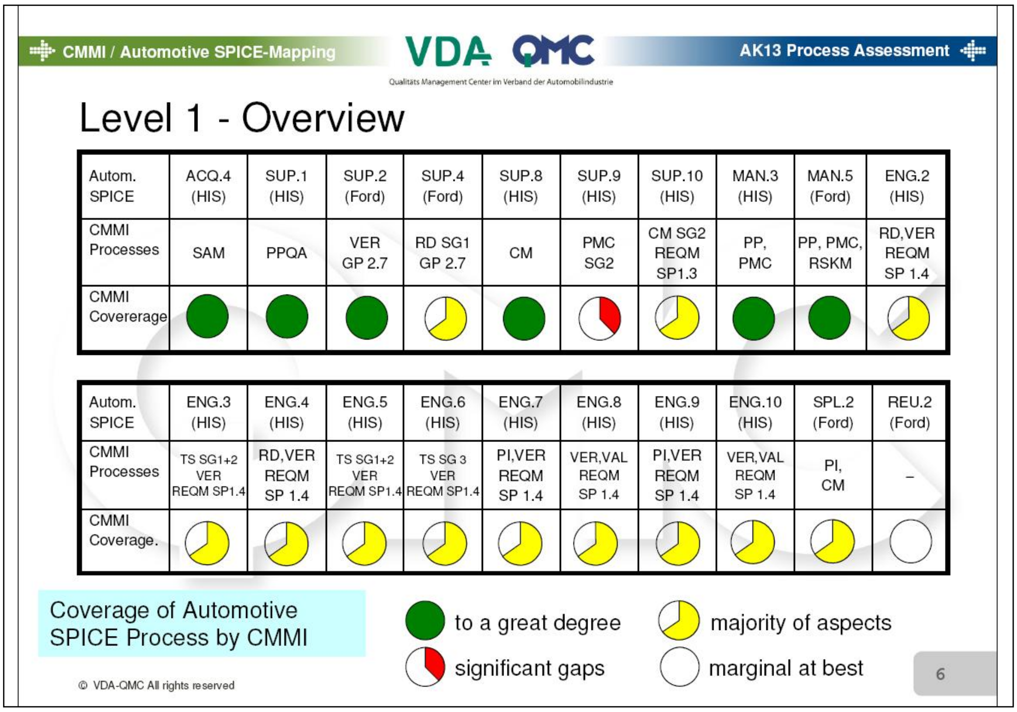

Figure 26: CMMI / Automotive SPICE-mapping[63]

As Automotive SPICE is more demanding this work will only deal with Automotive SPICE from here onwards.

2.3.2 ISO/DIS 26262: Road Vehicles – Functional Safety

The automotive specific standard ISO/DIS 26262 was published 2009 and is currently a draft version. *'ISO 26262 is the adaptation of IEC 61508 to comply with needs specific to the application sector of E/E systems within road vehicles. This adaptation applies to all activities during the safety lifecycle of safety-related systems comprised of electrical, electronic, and*

[62] Cf. Wlokka (2009), pp.16

[63] Wlokka (2009), p.6

software elements that provide safety-related functions.'[64] Before 2009 the general standard ISO/IEC 61508 (Functional safety of electrical/electronic/programmable electronic safety-related systems) was valid also for passenger vehicles.

In contrast to SPICE, which can be applied to all mechatronic projects, standards for functional safety are only relevant for development of products which include at least one safety critical function. The application is also limited to safety related activities and does not include the development of standard functions.[65]

Another difference is the fact, that adherence to standards in the area of safety is important legally This standards are understood as state of the art and any arbitrary deviation would lead to consequences in court proceedings in case of an accident. SPICE however is either a requirement from the internal management or a customer requirement which could be negotiated theoretically.[66]

Diverse to Automotive SPICE ISO 26262 does not give much attention to mechanics. Although mechanical components are essential for many safety functions the requirements for mechanical engineering are not specified accurately.

Finally Spice is clearly focused on processes and is open to any solution whereas ISO 26262 includes a safety lifecycle with its activities as well as technical content and recommendations for the use of decisive methods.[67]

ISO 26262 consists of 10 parts, see Figure 27. Part 1 is defining vocabulary, part 2 is dealing with management topics, parts 3 to 7 are describing the safety activities during the project lifecycle, part 8 includes the supporting processes, part 9 describes safety analysis and part 10 has information on the application. Parts 3 to 7 are aligned according the V-model, a model which is commonly used in engineering and will be described in Chapter 2.4 V-Model in Requirements Engineering. Also a separation of system level and subsystem level can be found, similar to Automotive SPICE. But ISO 26262 does not include mechanics as an own subsystem.

---

[64] ISO/DIS 26262-1, p.iv

[65] Cf. Löw/Pabst/Petry (2010), pp.121

[66] Cf. Löw/Pabst/Petry (2010), p.12

[67] Cf. Löw/Pabst/Petry (2010), pp.121

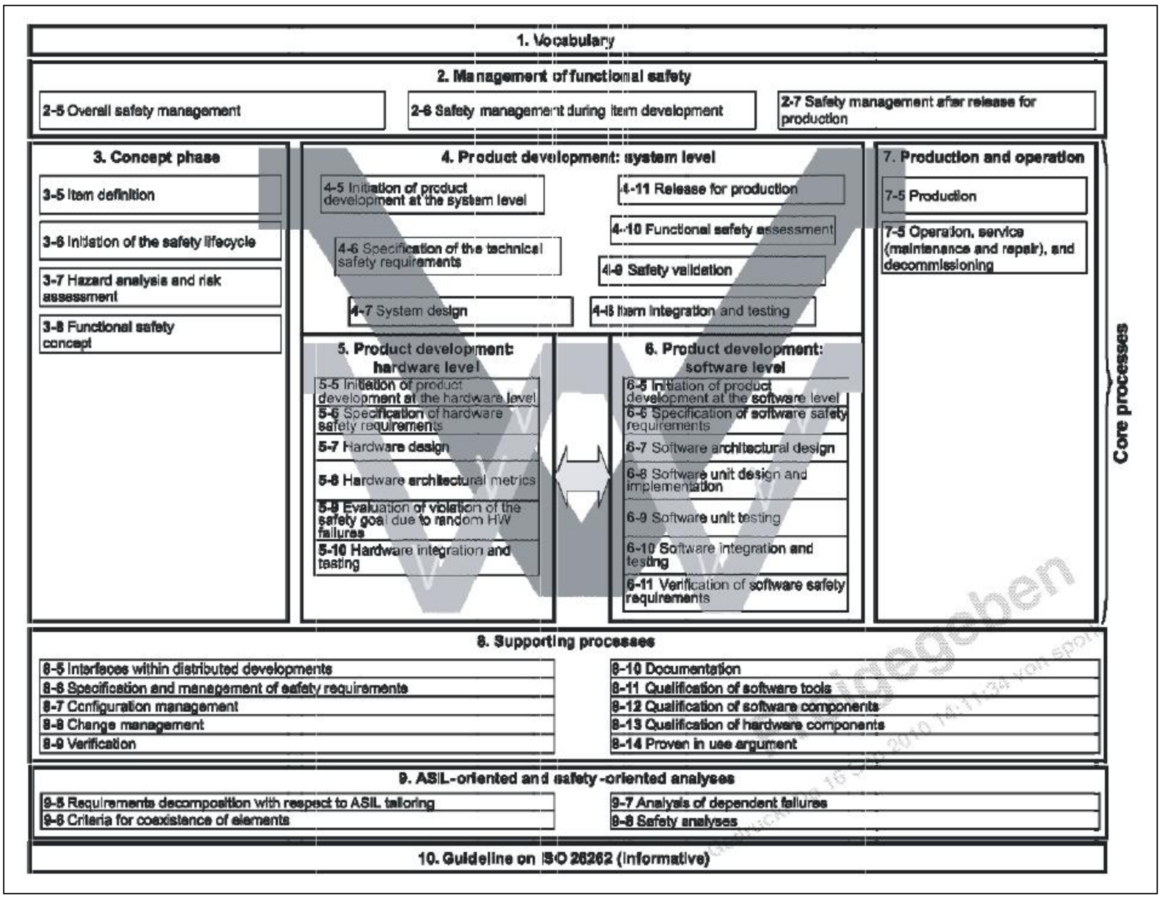

Figure 27: Overview of ISO 26262[68]

To understand the concept of ISO 26262 it is necessary to talk about risk. As risk can not be eliminated in technical systems it is the target to reduce the risk to an acceptable level. The standard requires a minimal risk reduction to the risk level of today's common technical solution. In addition the ALARP (As Low As Reasonable Possible) principle applies demanding a further reduction under consideration of cost and benefit.[69] This is also shown in Figure 28.

[68] ISO/DIS 26262-2, p.vi

[69] Cf. Löw/Pabst/Petry (2010), pp.65

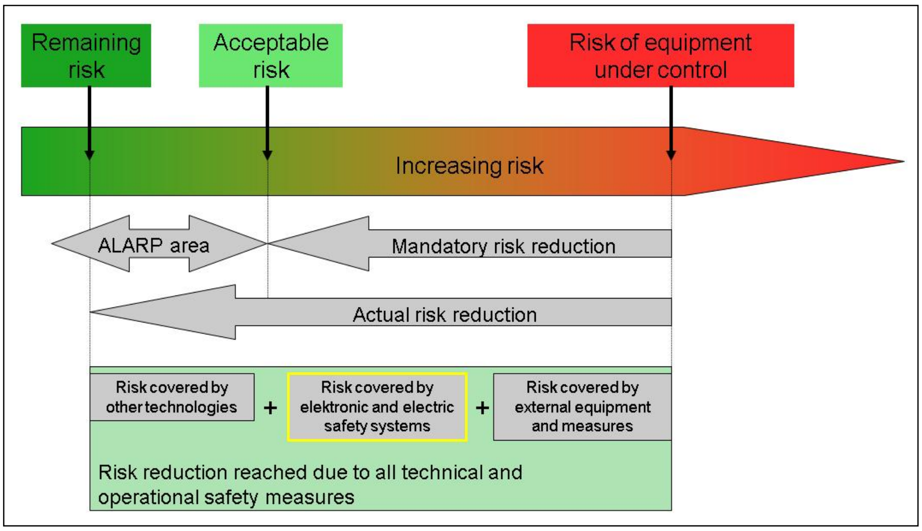

Figure 28: Concept for risk minimization[70]

The limits for acceptable risk levels depend on the product of severity, time of exposure and controllability. If there is no danger of involving harm to persons then the function is classified as QM (Quality Management). The higher the risk, which equals the probability for involving harm to people, the higher the ASIL (Automotive Safety Integrity Level) level, starting from ASIL A to ASIL D. Possible values for severity, exposure and controllability and their meaning are:[71]

Severity:

- S0: no injuries
- S1: light and moderate injuries
- S2: life-threatening injuries (survival probable)
- S3: life-threatening injuries (survival uncertain), fatal injuries

Exposure regarding operational situations:

- E0: Incredible
- E1: Very low probability
- E2: Low probability
- E3: Medium probability
- E4: High probability

Controllability:

- C0: Controllable in general

---

[70] Cf. Löw/Pabst/Petry (2010), p.65

[71] Cf. ISO/DIS 26262-3, pp.8

- C1: Simply controllable
- C2: Normally controllable
- C3: Difficult to control or uncontrollable

Potential combinations and their effect on safety classifications are shown in Table 2. A classification as QM denotes that ISO 26262 has no requirements in this specific area. Safety goals with related ASIL-levels have not only an influence on development methods but can also influence the product design. Especially ASIL C and D can be realized with redundant solutions and thus reduced in classification. This is shown in Figure 31.[72]

		Controllability C		
Severity S	Exposure E	C1	C2	C3
S1	E1	QM	QM	QM
	E2	QM	QM	QM
	E3	QM	QM	ASIL A
	E4	QM	ASIL A	ASIL B
S2	E1	QM	QM	QM
	E2	QM	QM	ASIL A
	E3	QM	ASIL A	ASIL B
	E4	ASIL A	ASIL B	ASIL C
S3	E1	QM	QM	ASIL A
	E2	QM	ASIL A	ASIL B
	E3	ASIL A	ASIL B	ASIL C
	E4	ASIL B	ASIL C	ASIL D

Table 2: ASIL determination[73]

ISO 26262 describes a mandatory safety lifecycle, see Figure 29. Requirements for many activities and artifacts depend strongly on the intended use of the product and its safety classification. Therefore it is extremely important to start with safety management right from project start stick to it during the concept phase as some omissions in this early stage can not be replaced by later effort. This is especially valid for hazard analysis and risk assessment, the starting point for safety activities. The hazard analysis has to be done for the whole system under consideration which is called item. The item definition describes the project scope, system borders, system

[72] Cf. ISO/DIS 26262-3, pp.9

[73] Cf. ISO/DIS 26262-3, p.10

interfaces and the interaction with the environment. All intended and possible unintended functions have to be evaluated regarding risk. For each risk a safety goal has to be defined. Then the safety functions are derived from the safety goals and each safety function is classified with the appropriate ASIL level. To be able to do this it is necessary to have a product concept in mind called preliminary architectural assumptions. Failures in the concept phase could lead to wrong ASIL classification of functions and thus defining an inappropriate functional safety concept.[74]

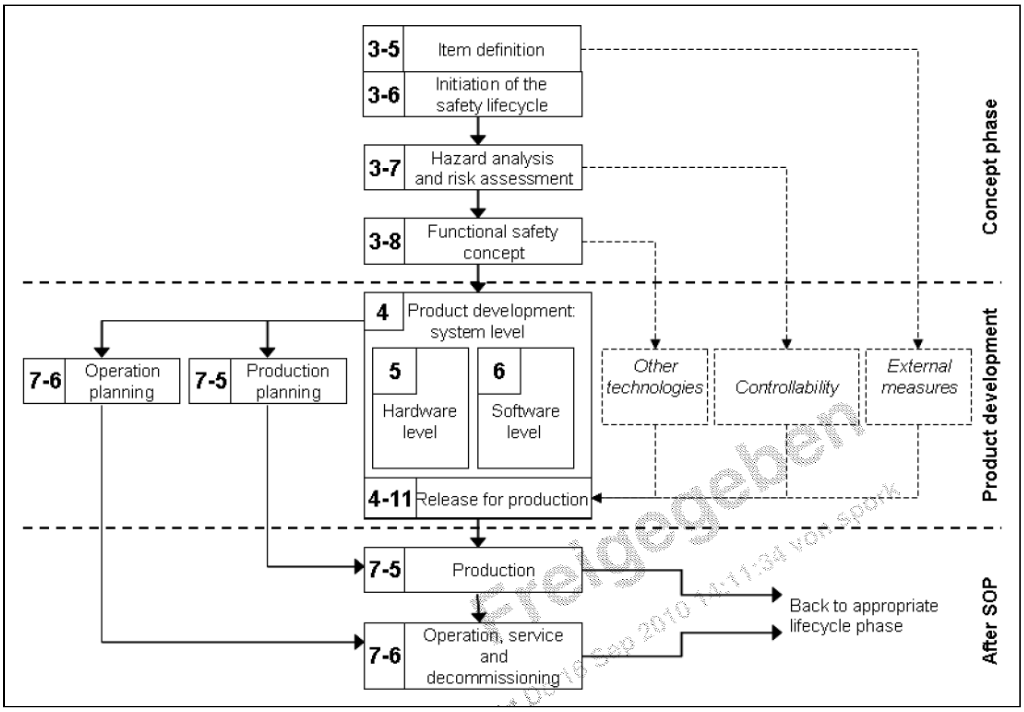

Figure 29: Safety lifecycle[75]

As the system architectural design established at the beginning of the development phase is based on the requirements including those of the functional safety concept it must deliver a technical solution which does provide sufficient safety. Also the use of adequate methods for specific ASIL levels as recommended in standard has to be ensured. Therefore any changes to the hazard analysis leading to higher risk are extremely critical. In the worst scenario the development process has to be started again. The development follows a similar logic as Automotive SPICE but is extended to operation planning, production planning and enriched with methods for risk

[74] Cf. Löw/Pabst/Petry, pp.53

[75] ISO/DIS 26262-2, p.3

calculation, verification, etc. Although the graph gives the impression of a strict linear proceeding the development phase is iterative. ISO 26262 deals with the activities after start of production, too. This area is only minor tackled by Automotive SPICE with change management and problem resolution management. ISO 26262 includes a development process description which can be mapped well to the key concept schematic of Automotive SPICE, see Figure 30. Both models are missing some elements of the other but the basic structure is the same. The safety standard starts with activities prior to the establishment of system requirements. This activities leading to system requirements can by summarized by other requirements as mentioned in Automotive SPICE. The further proceeding is the same as already described in the former chapter. What is not illustrated in the graph below but anyway required are tests for software and system.[76]

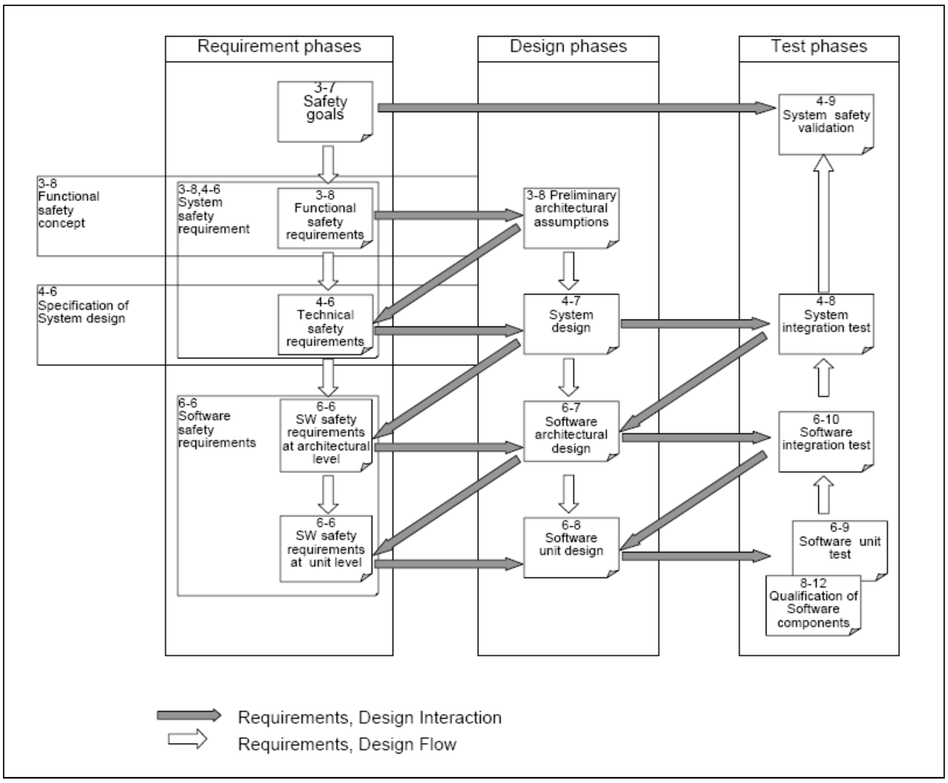

Figure 30: Software safety requirements process[77]

The system design as the final solution for the product is important for another reason. The initial safety classification of a technical safety requirement can be reduced by specific design measures on subsequent

[76] Cf. Löw/Pabst/Petry, pp.17

[77] ISO/DIS 26262-10, p.12

levels. The possible variants for the safety classification of subsequent requirement levels are called decomposition and shown in Figure 31. For example it is possible to decompose an ASIL C requirement to one ASIL B and one ASIL A requirement. Another potential solution is to derive one ASIL C requirement and one requirement just being classified as QM.[78]

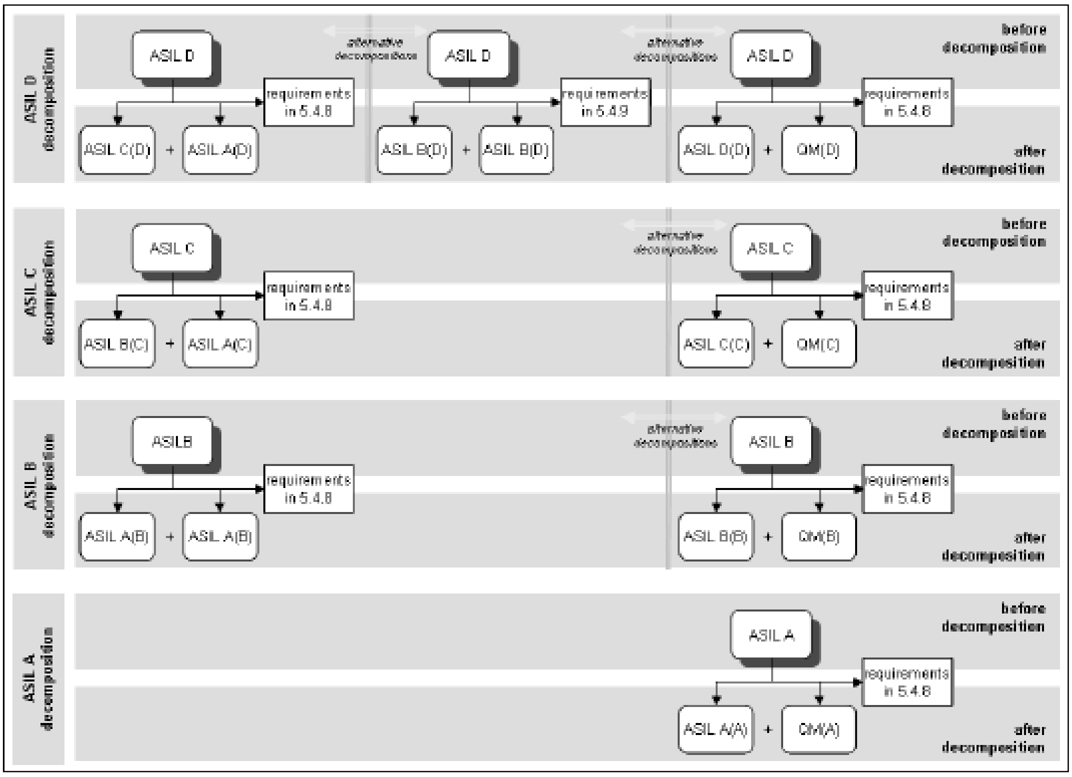

Figure 31: Classification scheme of ASILs when decomposing safety requirements[79]

In expert circles there is currently a discussion about the relationship and differences of ISO 26262 and Automotive SPICE. The Gate 4 SPICE meeting in October 2010, one of a periodic row of meetings of certified SPICE assessors, was dedicated to 'Safety in Automotive SPICE Assessments'. The discussion supports the opinion that a SPICE level of 2 or above will support functional safety and thus increase chances to be compliant with ISO 26262. SPICE is focused on processes for the whole project including non safety-critical activities. ISO 26262 is limited to safety critical activities but includes also technical content besides process aspects. Therefore the compliance with high SPICE levels neither guarantees the compliance with ISO 26262 nor does the compliance with ISO 26262 guarantee the compliance with

---

[78] Cf. ISO/DIS 26262-9, pp.4

[79] ISO/DIS 26262-9, p.6

SPICE. But as you can find a lot of structural similarities these standards support each other in any case.[80]

2.4 V-Model in Requirements Engineering

As already described requirements engineering is an essential task in automotive engineering. It is the structural backbone of standards and the main communication channel for technical experts. Requirements help to allocate information to different levels like system, subsystem or software unit as well as to different components. In addition it is possible to show interactions due to traces.
'On a high level view the requirements-engineering is an iterative, incremental and recursive thing. Figure 32 *describes this in a graphical form. Iterative means that a function/feature which is implemented in the first iteration can be added or fine-tuned in the next iterations. The incremental part is that not all functions/features are implemented in the first iteration. By prioritization and release planning the functions/features are realized. The recursive part is that the OEM and the suppliers are making their own requirements-engineering phase. This requirements-engineering is focused on different abstraction levels and different outputs but is done by all parties.'*[81]

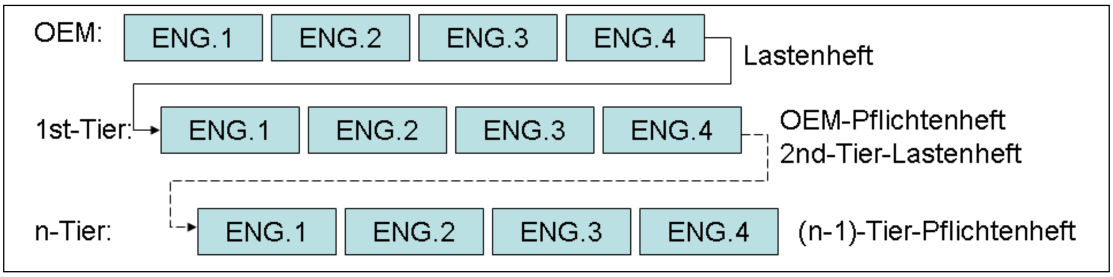

Figure 32:Typical recursive specification of a system on different abstraction levels[82]

Within one level of abstraction the proceeding in automotive engineering is usually based on the V-model. SPICE does not directly require the V-model but the engineering processes defined within the standard follow the logic of it. Even the names of the single processes could be taken out of a generic V-model. ISO 26262 has included different kinds of the V-model in its concept

[80] Pabst/Seegers (2010), pp.9

[81] Poth (2006), p. 3.6

[82] Poth (2006), p. 3.6

as already shown in Chapter 2.3.2. The whole system has to be developed according a big V, the subsystems hardware and software as self-contained systems are developed according their own Vs. The structure and logic is applied generally but at the same time it offers the possibility to have different cycle times and a different number of iterations for the three systems. In fact it is very common in automotive business to have 2 prototype phases (system Vs), three or four sample phases for the ECU (Electronic Control Unit) and some more software releases.[83]

The origin of the V-model was in Germany. Initially it was scheduled to use it for IT projects of the Federal Ministry of Defence or Public Administration. With the publication of the V-model 97 it got big importance in industry. Nearly at the same time a US version for satellite systems involving hardware, software and human interaction was created.[84]

Currently a huge number of variants with different appearance and scope exists. Some of the models include activities before and/or after the development phase, many focus on development alone. All have in common that development is shown as a V in a two dimensional system with time displayed on the abscissa and level of detail displayed on the ordinate, see Figure 33. If the model includes more than development, you will find horizontal extensions to the pure V.

[83] Cf. Löw/Pabst/Petry, pp.16

[84] Cf. Wallmüller (2007) pp.18

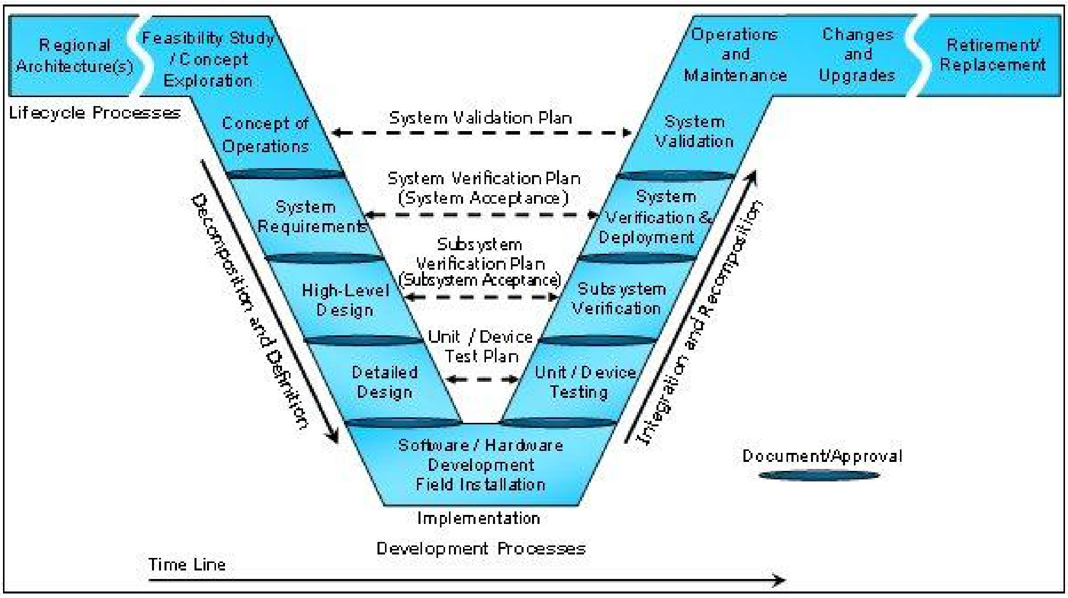

Figure 33: V-Model[85]

As this work is dedicated to requirements management it will focus on the development phase. The left part of the V – often called the defining stream - starts with a concept of operation which can be translated to customer requirements in automotive. This is then refined to the systems specification which also includes requirements from other sources and the definition of acceptance criteria. The next step is to design a technical solution with defined components and interfaces. These components are specified including acceptance criteria. They are developed independently. If necessary a complex component, like software, is again treated like a subsystem which needs an own architectural design. In big systems this logic can be applied several times, see Figure 34. The last stage of this decomposition of systems would be the smallest system including software, hardware and other technologies like mechanics. The level of detail as well as the number of requirements is increasing with each step down.[86]

[85] http://ops.fhwa.dot.gov/publications/tptms/primer/index.htm, 22.01.2011

[86] Cf. Hruschka/Rupp (2002), pp.181

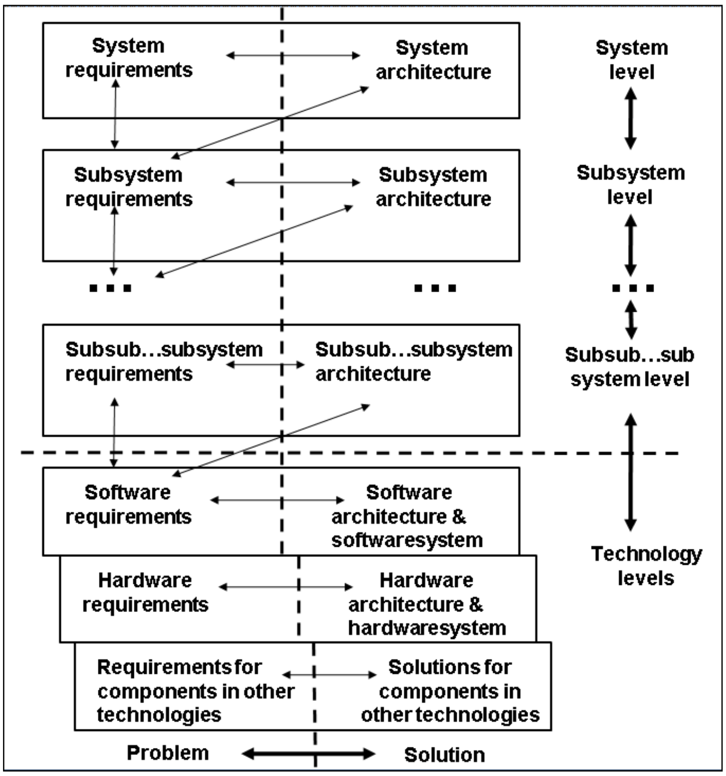

Figure 34: System decomposition[87]

The right half of the V represents activities needed for verification and validation and is commonly called verification stream. To fulfil these activities effective the basic input has to be specified in the defining stream already. This includes acceptance criteria for functional and non-functional requirements and interface specifications. The components are tested on their own as far as possible and the integration to bigger systems starts only after positive testing. Depending on the integration strategy you will have different integration steps with test targeting on interfaces alternating with tests targeting on functionality. At last the overall system is validated before final release to the end customer.

The V-model itself exhibits only activities, results and their dependencies. A strict time line or milestones at the end of a phase are missing. Despite this absence of timing guidelines it is possible to map the V-model to lifecycle models in project management which are predominantly stage gate processes. Also the fact that development is done in iterations in most business areas is not represented. Therefore many companies adapt the structure of the V-model to their business needs. Figure 35 shows as one

[87] Cf. Hruschka/Rupp (2002), p.183

example the company specific V-model of Continental AG, one of the world's biggest automotive suppliers. The model is enhanced with reviews which are used as quality gates and a circle on system level representing iterations. On the bottom of the system V you can see three subsystem Vs, the software V being elaborated separately. On top of the system V you can find additional processes running parallel to engineering. All but one of these processes are included in the Automotive SPICE HIS scope.[88]

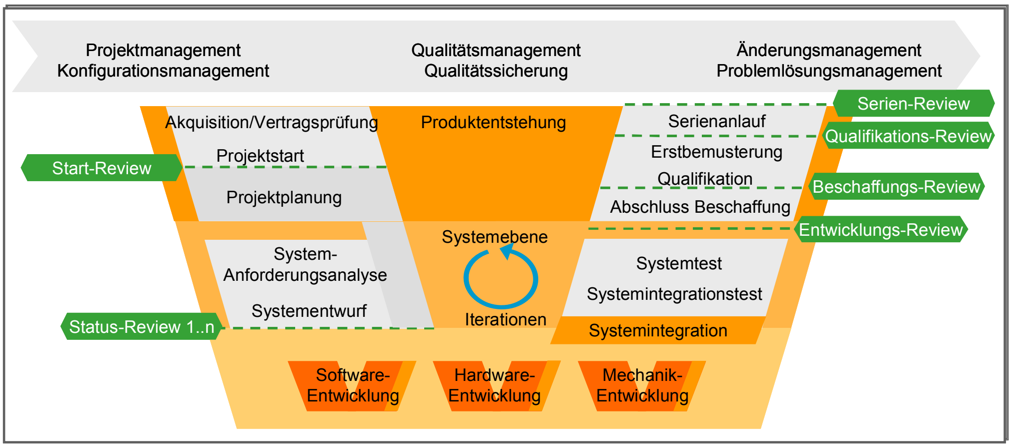

Figure 35: V-Model of Continental AG[89]

A second example for a company specific V-model is shown in Figure 36. This model is taken from an ABB division developing safety critical automation systems. This figure shows the relation between the ABB Gate model, the ABB V-model and the ABB UML method used for the development of safety related products. This model has included the gates of project management below the V. What is missing is any information about iterations. The reason might be that in customer specific plant engineering and production the first unit build is usually also the shipped product. On the other hand the model shows some additional safety relevant work products. The structure of the same V is applied to safety critical and non safety critical requirements.

[88] Cf. Höhn et al. (2009), pp.23

[89] Höhn et al. (2009), p.23

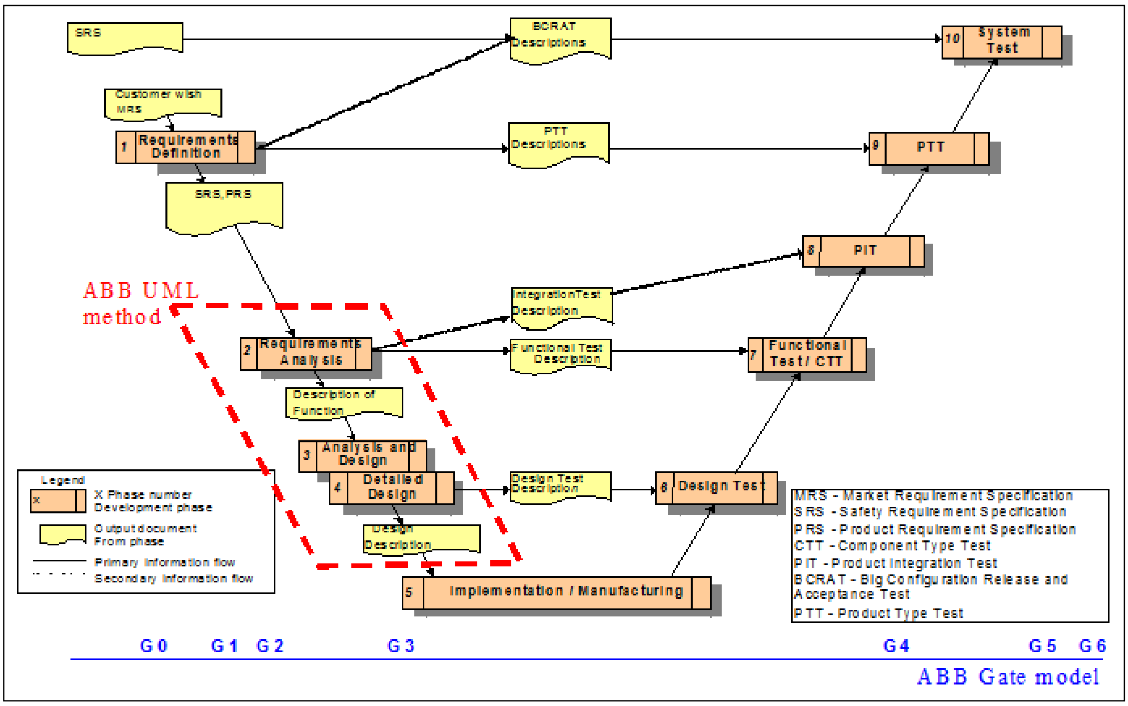

Figure 36: V-Model of ABB[90]

Although these examples have a different appearance they show a common structure and logic. It is necessary to fit the model to the existing processes. If a company uses special tools or methods they have to be integrated in the model.

2.5 Main Tools in Automotive Requirements Engineering

It was already discussed that it is necessary to set up tool support for requirements engineering of complex and sometimes even safety critical projects. There are a lot of tools on the market, most of them coming from software engineering. In automotive engineering the spread of requirements management tools started this millennium. Especially the release of Automotive SPICE and its immediate enforcement through the HIS group boosted the appliance of tools beginning from 2005. Although most OEMs create at least their specifications partly with support of a dedicated tool one can still find a noteworthy part of specifications written with Microsoft tools.
The tremendous increase of demand for requirements management led to a situation in which new suppliers tried to enter the market. At the same time the number of users and expectations regarding functionality of tools raised

[90] Thorsen et al. (2004), p. B.21

significantly. Any statement regarding functional advantages can change shortly. This work is therefore limited to the two most common tools, DOORS and MKS. Those two are also the only tools in use in the analyzed case studies.

If a requirements management tool is used by an OEM it is predominantly DOORS. Any automatic exchange of data is facilitated if suppliers stick to DOORS, too. Although a standardized interchange format for requirements between different tool suppliers is announced for years and draft versions are on the market you will hardly find it in reality up to now. The other tool currently common in automotive industry is MKS. MKS was originally a tool for configuration management of software source code. Later it offered an add on for requirements management with a separate license. Within the last years the modules for configuration management and requirements management were integrated to one product. As many companies used MKS for configuration management some of them moved along with it to requirements management. Both tools are capable to fulfill the demands of automotive standards, in particular bi-lateral traceability can be established with both. Still they have different strengths and weaknesses. Several comparisons between DOORS and MKS could be found within MAGNA the content of which was partially proven wrong by some case studies. Therefore this section will only give a brief overview about main functions of these tools and their evaluation according the experience of the case studies. Relevant remarks to tool functionality are included in the single case studies, too.

- **DOORS**

DOORS has an additional tool for data exchange with DOORS, making exchange easy with any customer or supplier also using DOORS. Data editing is similar to MSWord and processing of OLE (Object Linking and Embedding) objects is easy. Data are stored in modules which give the user the impression of MSWord documents. Simultaneous work on one module is possible with some restrictions. Functionality for baselining, branching and merging are available. Exports to common formats like MSWord or PDF are possible. Use of DOORS for tracking of activities related to requirements management is not possible. The graphical monitoring of progress with dashboards is limited, a web client is only available as option. DOORS cannot be used for configuration management of source code. The list price is about one third higher than MKS.

- **MKS**

Data exchange with DOORS is difficult, but possible. Data editing is only simple if it is plain text. Handling of OLE objects is burdensome. Data can be stored in single elements or can be grouped to something similar to a document, depending on company settings. In the first case parallel work on the same project is easy, in the second case limitations apply. Branching and baselining is possible, merging not. Exports are available via HTML (Hyper Text Markup Language) -format or PDF. PDF exporting still creates some troubles. MKS offers the ability to implement additional elements in the requirements management tool like tasks, problems or change requests. MKS has a good dashboard function which can be used for monitoring of requirements or any other implemented tasks. The web client is included in the standard license as well as a full configuration management tool. The list price is significantly lower.

3 Best Practices for Requirements Engineering at MAGNA

A main part of this thesis is the analysis of different solutions for mechatronic engineering processes with the main focus on requirements management and related processes. This chapter includes five case studies of three different MAGNA groups. MAGNA is states to be the most diversified automotive supplier in the world. The variety of products with diverse complexity developed in a double digit number of engineering sites around the globe leads to significantly different solution for requirements engineering. Nevertheless customers expect the same high quality from all engineering sites and legal requirements have to be fulfilled anyway.

The case studies describe the general strategy in the area of engineering processes thus setting the context for requirements management. The structure of requirements management is illustrated accurately, but some details have to be omitted for know-how protection. An overview of the interaction with test management is followed by a summary emphasizing strengths of each case.

3.1 Case Study MAGNA Powertrain D&CCS

The engineering in Lannach is the main development site for MAGNA Powertrains D&CCS (Drivetrain and Chassis Control Systems) group. A second, smaller engineering site of this group is located in Troy, Michigan. As this second engineering site follows nearly the same processes and has more or less the same infrastructure, this paper will only focus on the situation in Lannach.

3.1.1 General overview

In Lannach exists a well elaborated V-model as basis for all engineering activities. Many additional processes are defined to support and manage the activities along the development. The established eProcess (engineering process) landscape can be found in Figure 37.

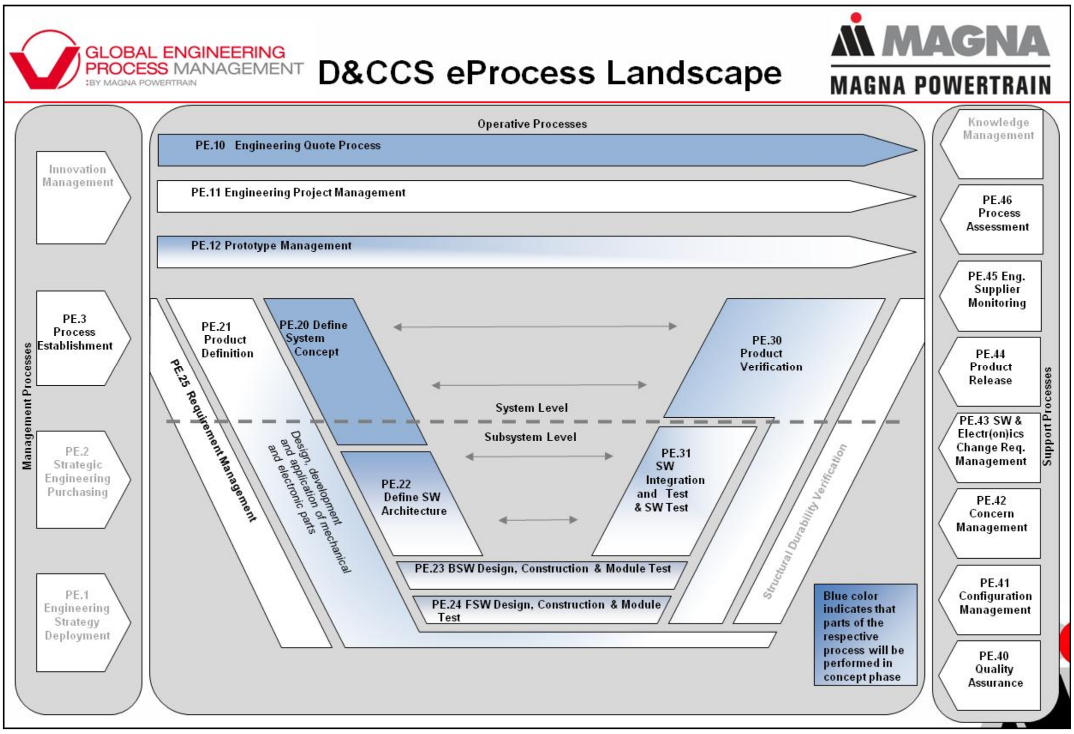

Figure 37: Engineering Process Landscape[91]

The defined processes include the whole engineering lifecycle. Besides the content of the HIS scope the eProcess Landscape implies topics dealing with development of mechanic parts and their verification which are not required by current automotive standards. Also the quote phase is covered.

[91] MAGNA Powertrain

Furthermore the two processes ‘process establishment’ and ‘process assessment’ are targeting on effective process definition, roll-out and continuous improvement. Last but not least some strategic aspects are included in the landscape, too.

In the V-model one can find the common separation between system level and subsystem level. This is similar to the decomposition of the system to the subsystems mechanics, electronics and software in Automotive SPICE. The software development is then detailed by the definition of the software architecture and followed by separate processes for different parts of the software: operating software, which is manually coded and application software, the code of which is automatic generated.

The V-model includes a second separation which is rarely to be found. The design of mechanical and electronic parts is distinguished from software engineering, both on the defining and verifying part of the V. One specific aspect to be tested for mechanical components lies on reliability and is incorporated in the process structural durability verification.

A reference to timing can be found in the process landscape, too. A blue colored process indicates that parts of the process are performed during the concept phase. The relative darkness of the process ‘define system concept’ shows the importance of a system design concept for project success even before a customer order. Feasibility, timing and profitability can be essentially derived from it.

3.1.2 Requirements management structure

A point to draw the attention to is the fact, that requirements engineering is a single process including all levels of the product from system level down to software unit level. Usually requirements management is an inherent part of the processes at the different levels, sometimes it is one of the supporting processes.

Requirements management is structured in four levels:

- Customer requirements: SORs (Specification Of Requirements)
- System requirements: SOWs (Statement Of Work)
- Subsystem requirements: SCRs (System Component Requirement)
- Software Module Requirements: SMRs

All relevant documents on customer level are stored in the configuration management system SAP EasyDMS and evaluated in the initial simultaneous engineering team meetings by nominated experts. The review is documented in a form called ‘Requirerments Analysis Matrix’, in which the content of the customer documents is either accepted, a different solution is suggested or in a few cases even rejected. The integrated result is then discussed with the customer and the final adjustments are basis for the development of the product. The SORs on customer level are not entered into MKS, the database for requirements management.

As next step requirements management is established. Figure 38 gives an overview of the requirements structure. All arrows show possible links between single elements. As the customer level is not imported to MKS, tool supported requirements management starts on the system level. Requirements of the system component level are derived from the system level and software module requirements are linked to requirements of the system component software. As far as technically required also additional content for other modules is derived from system components.

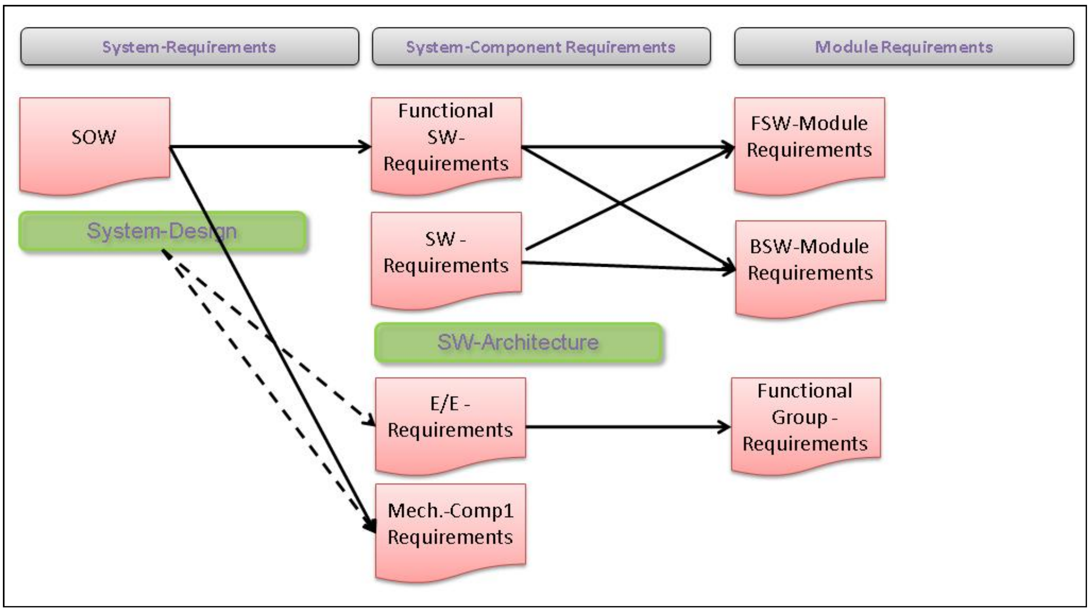

Figure 38: Structure of requirements management[92]

Requirements management is supported by a combination of MKS and MSWord. MKS is used to ensure bi-lateral traceability from customer requirements to internal requirements until the test cases. Technical challenging projects have some hundred system requirements and by breaking down these requirements to system component requirements and software module requirements this figure increases significantly. Furthermore

[92] MAGNA Powertrain

a lot of requirement related engineering activities are tracked with MKS, including change management, problem resolution management and task tracking. It is also possible to establish traces between these activities and the related requirements. Because of functional weaknesses of MKS regarding editability of OLE-objects in MKS MSWord is used to overcome this limitation, too.

All elements with red color in Figure 38Figure 38 consist of requirements in MKS and description in MSWord. System design and software architecture are written only in MSWord. The system design includes an overview about the main components and their interfaces as well as a simulation. It describes not only the expected functionality but also how this functionality is realized. The way of realization can lead to requirements for the components, too. The software architecture again gives an overview about the software components, but is divided in a high level design and a low level design. The high level design consists of the main software components and their interfaces. In case of software which is delivered by a third party, e.g. a vehicle control component of the customer, there would be no low level software design. The software architecture defines in addition time behavior of the software and decides which functionality has to be fulfilled in which task.

The combination of MKS and MSWord is realized with a self developed interface between these tools. The specification is written in an MSWord-Document which is enriched with additional information and structured in a way that helps the reader to understand the content. Functional and non-functional requirements are embedded in this document, but can easily be identified because each requirement has always a small table called requirements frame at the beginning. The frame includes the following information, see also Figure 39:

- MKS ID: identification number in MKS
- Subject: short description similar to a headline
- Description: detailed description including text, graphs, tables, etc. to fully specify the requirement
- Control boxes: notifications for attribute settings in MKS

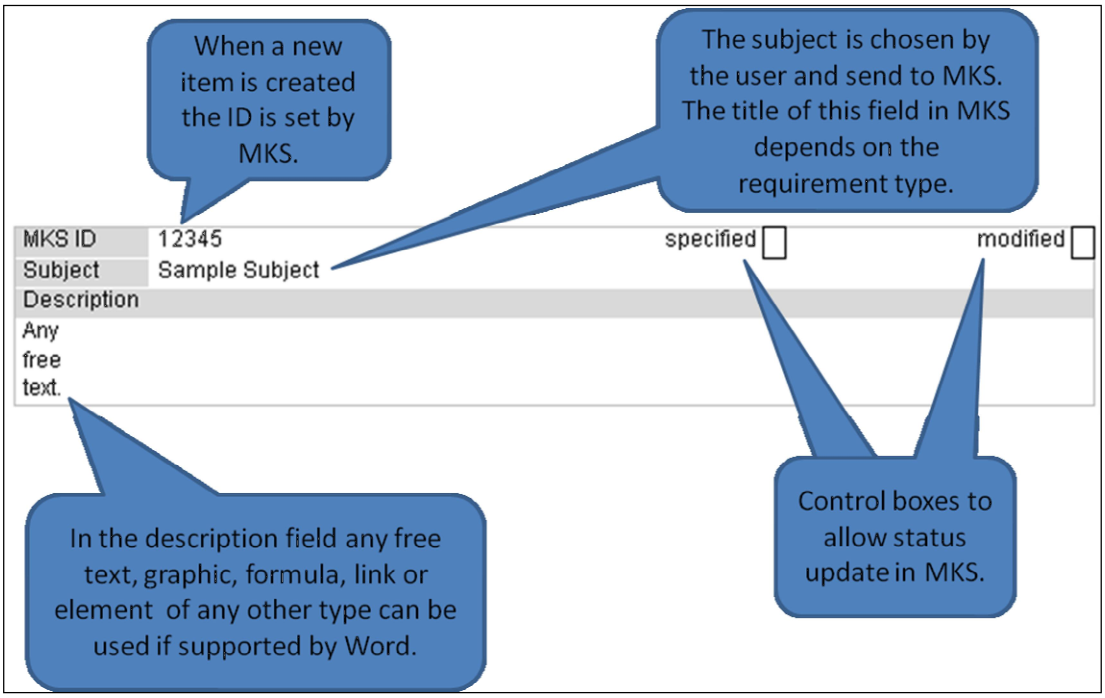

Figure 39: Requirements frame[93]

The specification of requirements starts with the creation of the MSWord document which is stored in SAP EasyDMS. At the beginning the requirement frames do not have MKS IDs. When the specification has a certain degree of maturity the document is imported to MKS the first time. The interface between MSWord and MKS creates for each requirement frame an item in MKS including the information in the cell subject and the EasyDMS number of the whole document. Subsequently the MKS ID is transferred back to the requirements frame in MSWord. This procedure ensures bilateral traceability between MKS and MSWord. This is shown in Figure 40, where the fields with red frames are used for traceability. After further refinements in MSWord it is always possible to make an update in MKS. Additional requirements not having an MKS ID in the requirements frame will be added in MKS and their ID transferred back to MSWord as described above. When the field ‘description’ was changed the information is passed on by ticking one of the checkboxes. The box ‘specified’ is used when the description is completed and the status of the requirement in MKS should be set to the respective status. The box ‘modified’ indicates changes to the text which might be of interest for other people. If for example the

[93] MAGNA Powertrain

functional behavior is changed or the acceptance criteria set to an other level the test department gets the information also via MKS.

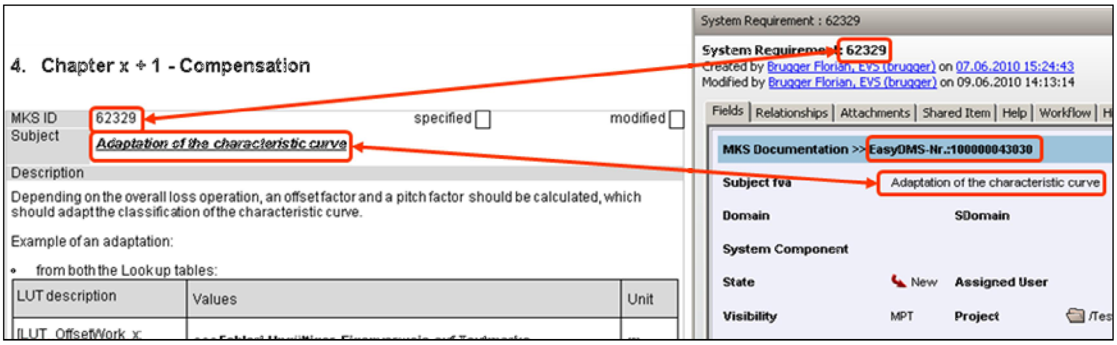

Figure 40: Traceability between MSWord and MKS[94]

Requirements in MKS are used to ensure traceability. Each requirement has a unique ID and can be linked to other requirements one level above and/or one level below. In addition links can be set to other items like change requests or problems. But MKS is not only used to establish traces. Requirements have a list of attributes supporting several other processes. The next paragraph will just focus on the most important and their use.

The majority of attributes are in the area of project management One can find the name of the project, the customer, the responsible person for the topic, a field for priority and a field for the milestone until implementation is planned. Furthermore has each requirement a state. The lifecycle of requirements and their use will be explained later. An attribute indicates the defined ASIL-Level for the requirement enabling to filter for all safety critical functions and their related requirements easily. The requirement is categorized either as functional or non-functional and in a multi-value field the visibility is set to all relevant parties. This field is used in all MKS elements to make directed exports for customers and suppliers. Affected components can be allocated to the requirements in an attribute containing a list with the components used in the architecture at the relevant level.

The requirements workflow is equal on all requirements levels. This ensures a common understanding of expected quality of the content, no matter if one is looking at a system requirement or a module requirement on software level. The number of different states is kept to a minimum by purpose and can be seen in Figure 41.

[94] MAGNA Powertrain

- Unspecified
  This state is technically necessary but is not used.
- New
  The requirement is identified but not fully described and analyzed. This state is not sufficient to proceed with implementation and test case creation.
- Specified
  The requirement is specified and its safety level is defined. It can be started with implementation and verification activities. A performed analysis and review would be beneficial but are not mandatory.
- Released
  The requirement is fully specified, analyzed, reviewed and planned. However, it is not necessarily implemented yet.
- Closed
  The requirement is rejected and will not be implemented.

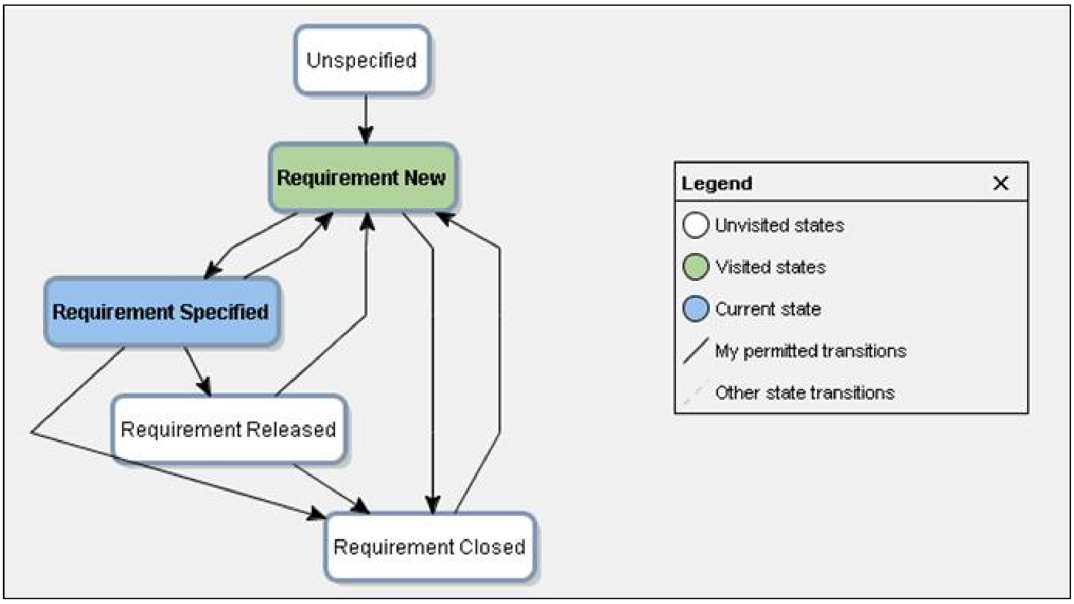

Figure 41: Requirements workflow[95]

The graphical representation of the workflow shows also all possible transitions. If transitions are assigned to specific roles and restricted to the current user you would find dotted lines. The adherence to the workflow is fostered by mandatory fields, which are required to realize transitions.

[95] MAGNA Powertrain

3.1.3 Link between requirements management and test management

Test management is done similar to requirements management. The detailed test specification for tests done in the vehicle or on test benches is written as MSWord document and stored in SAP EasyDMS. In this way it is easy to start with the test planning early and to complete the test specification over time as the requirement specification gets complete. From an early phase in the project it is clear which and how many tests are needed and fixtures or other needed material can be organized in time. The traceability between the test item in MKS and the test specification in MSWord is again the unique MKS ID. The link between a requirement and test is realized in MKS and the detailed requirement specification can be found in the functional specification in MSWord. The traceability is shown in Figure 42, where the red arrows indicate traceability due to unique IDs and the blue arrows indicate traceability within the MKS database.

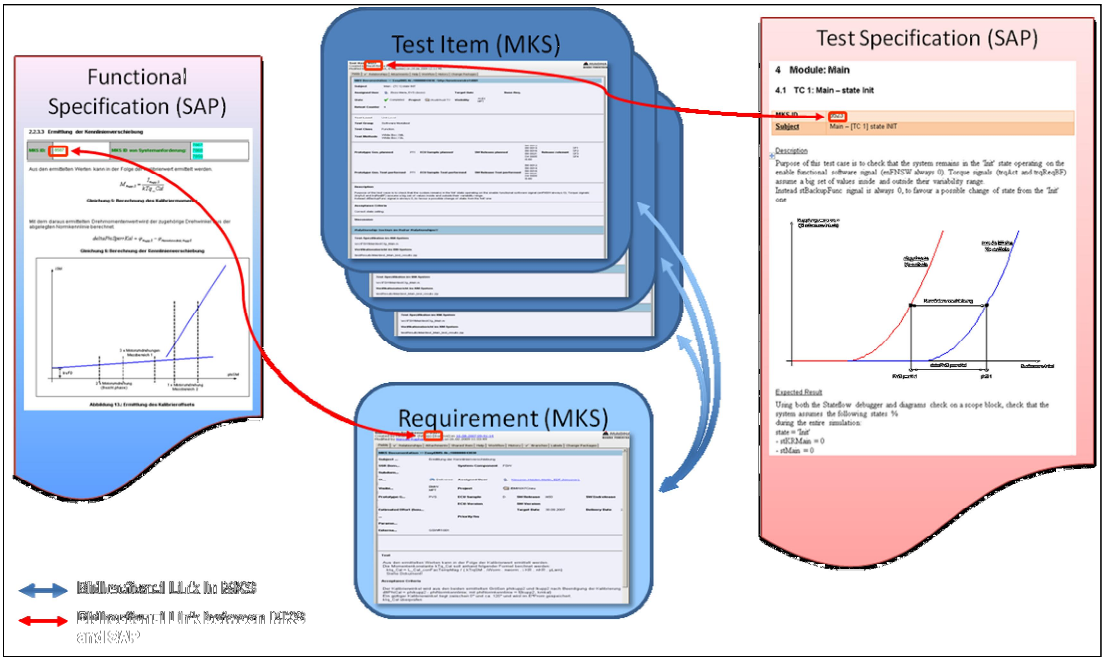

Figure 42: Traceability between requirements and tests[96]

Tests on software level are done in a special tool environment. Test specifications and test reports are stored in the configuration management tool of MKS. These tests are established in a way that they can be performed automatically and the test results are imported in MKS. Because of the high

[96] MAGNA Powertrain

number of software tests and the regression of tests for each major release an automated interface is essential to limit effort and avoid errors during data transfer.

Whereas all links between MSWord and MKS are based on the ID and apply just for a 1:1 relationship, MKS offers a n:m relationship for links. The picture in the figure above shows an example with three test cases for a single requirement. It is very common that it is necessary to perform more than one test for one requirement. On the other hand it is also common that specific test cases check more than one requirement. This can lead to a complex net of thousands of traces between requirements and tests in large projects which can be handled only with a database.

Although the testing has to be done requirement based, a simple check if each of the requirements has a test case is not enough. Nonetheless this check helps to identify missing tests. Additionally it is necessary to make reviews if the already planned tests are sufficient for functionality and to create further tests cases like tests against potential failures.

Having the information of current requirements states and test case states in one database delivers several advantages. First it enables each team member to access relevant information in short time. At the same time all team members are only responsible to keep their work up to date on one single point. Moreover the effort for updating of specific information is eliminated. For example there is no state ‘tested’ in the requirements management workflow because it can be calculated with the states of the linked test cases. Finally it is possible to monitor the development progress of the whole project within one tool, as long as the information in the system is correct.

The whole chain of activities is shown in Figure 43. One can see how traceability is realized, when specific activities are performed and which tools are used.

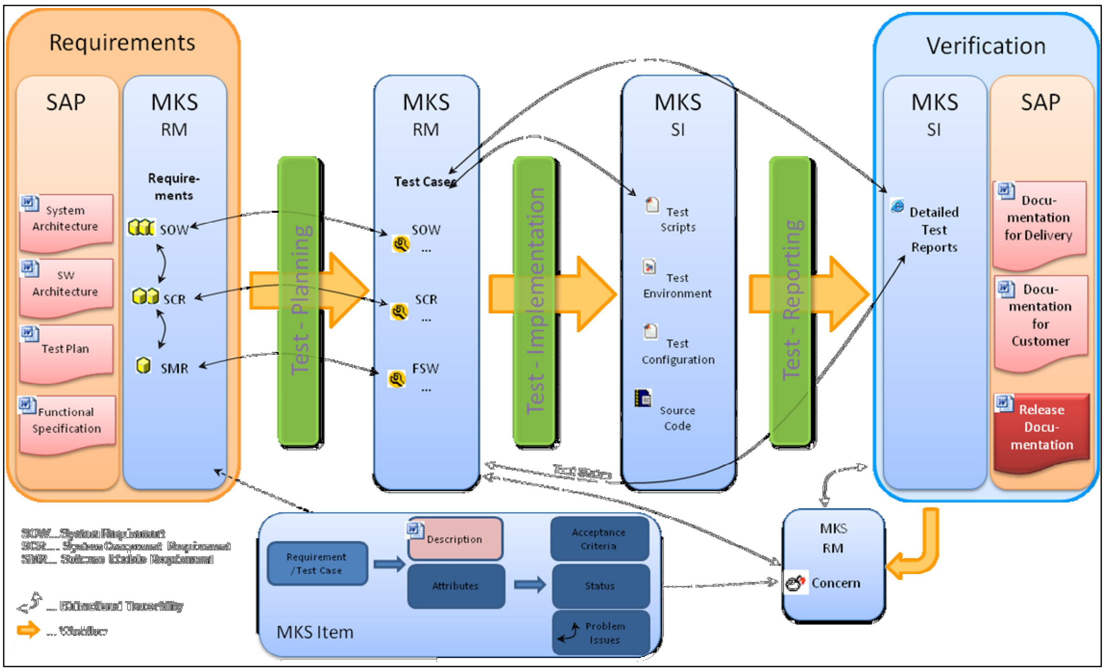

Figure 43: Overview of activities and related tools[97]

The specification starts in MSWord and is imported to the requirements management of MKS, where all attributes are set and the links between requirements are drawn. Then the test cases are defined and for software tests the test implementation is done in MKS SI. The detailed test reports for software tests are stored in MKS SI, the test reports on system level are stored in SAP EasyDMS. For each release which is delivered to the customer the whole status of requirements and tests is documented and stored in SAP EasyDMS.

As MKS is an effective tool for task tracking and monitoring several other requirements related activities are set up in it. In the picture you can find the item ‘Concern’ which is the type of task for problem resolution management. By having the concern again in MKS it can be linked to the test case where the problem occurred and to the requirement which is not fulfilled. All necessary information for dealing with the concern are easily and fast acquired. A similar situation is with change management, where all affected requirements can be linked to the change request and thus analysis is supported.

[97] MAGNA Powertrain

3.1.4 Summary

The solution at D&CCs provides several advantages. It is very user friendly as it supports the work with MSWord, which is best known by most of the developers and has superior functionality as document editor. The attributes are set in MKS and the functionality of traces and monitoring is used here. The storage of the documents in SAP EasyDMS enables access also for team members outside engineering who might not be connected to MKS. In this way it helps to keep license fees on an affordable level and saves training time for people who need this information only from time to time. A special feature of this installation is the possibility for offline work. The requirements specification can be further deployed in a discussion with the customer on his site and the result can be imported in the requirements management system after return to the office. An additional benefit is the fact that also the requirement related activities and tests are tracked in MKS. This helps to meet the objectives of supporting processes like change management and problem resolution management. Furthermore it allows the project manager to get an excellent overview of the progress of his project.

Besides all this advantages one drawback has to be mentioned. The reusability of requirements for platform engineering is limited. This is due to the fact that the content of requirements is split in content in MKS and content in MSWord. Therefore it is necessary to rework the specification document first and import it again in MKS. With the new import new elements are created and all traces have to be established again. Also the link to the parent project is lost.

With the chosen structure of three levels of requirements and the additional design documents for system and software the requirements of the automotive standards are fulfilled. The close reference to the structure in the standard make it easy to map the expected outcomes of the Automotive SPICE processes to the processes at D&CCS. The requests on requirement management of ISO/DIS 26262 are also covered with this structure. As not all projects are safety critical, the processes are enriched with the necessary activities only if needed.

3.2 Case Study MAGNA E-Car, Thondorf

MAGNA E–car has a special situation within the MAGNA groups. In the case of development of an electric vehicle this group acts like an OEM. The engineering in Thondorf is focused on the development of battery modules. Even in the case of development of battery modules it can happen that the customer is not experienced yet and MAGNA E-car is developing mainly according its own specifications which are released by the customer. As the electrification of vehicles is rather new there is limited experience in respect to safety critical engineering. In combination with the publication of the ISO/DIS 26262 it is obvious that there is a need on the market to set an example which will be considered as state of the art for similar products. As long as there is no general accepted example as guideline different interpretations of the standard make effective and efficient development difficult. The combination of partially undefined customer specifications and uncertainness of requirements from the safety standard lead to a situation in which the system level reaches exceptional importance.

3.2.1 General Overview

Like in Lannach a well defined process lifecycle model is established, see Figure 44. The model is based on the V-model and the processes additionally required by Automotive SPICE are also implemented. The model is enhanced with milestones referring to freeze and release points.

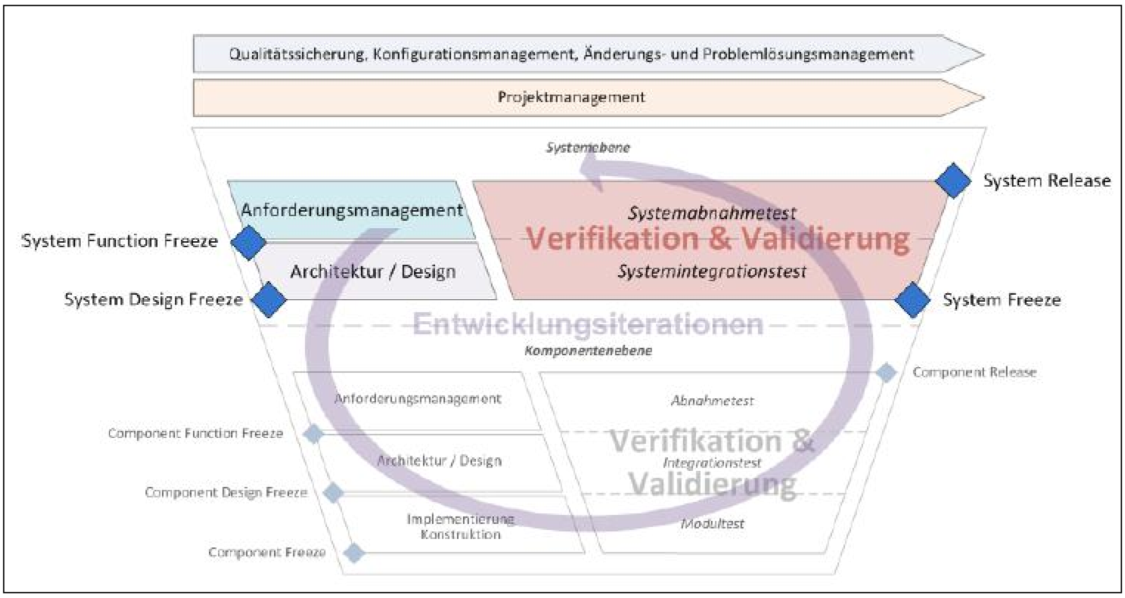

Figure 44: Project lifecycle model with freeze points[98]

The V-model is divided in the system level, which is indicated through bold letters and colored shapes, and the component level, which is indicated through grey letters and white shapes. The close relationship between defining and verifying activities is shown by the immediate following of verification processes giving the impression of a funnel. The intention is to show that planning of verification has to start immediately after defining the product to be able to have all test specifications and the necessary test equipment ready in time. Whereas a usual Tier 1 supplier - like all other MAGNA Groups - is clearly focused on verification of a product MAGNA E-Car is sometimes validating its products, too.
The single activities are described in a process model directly referring to Automotive SPICE. All processes required by the HIS group are described separately. The process model is shown in Figure 45.

[98] MAGNA E-Car

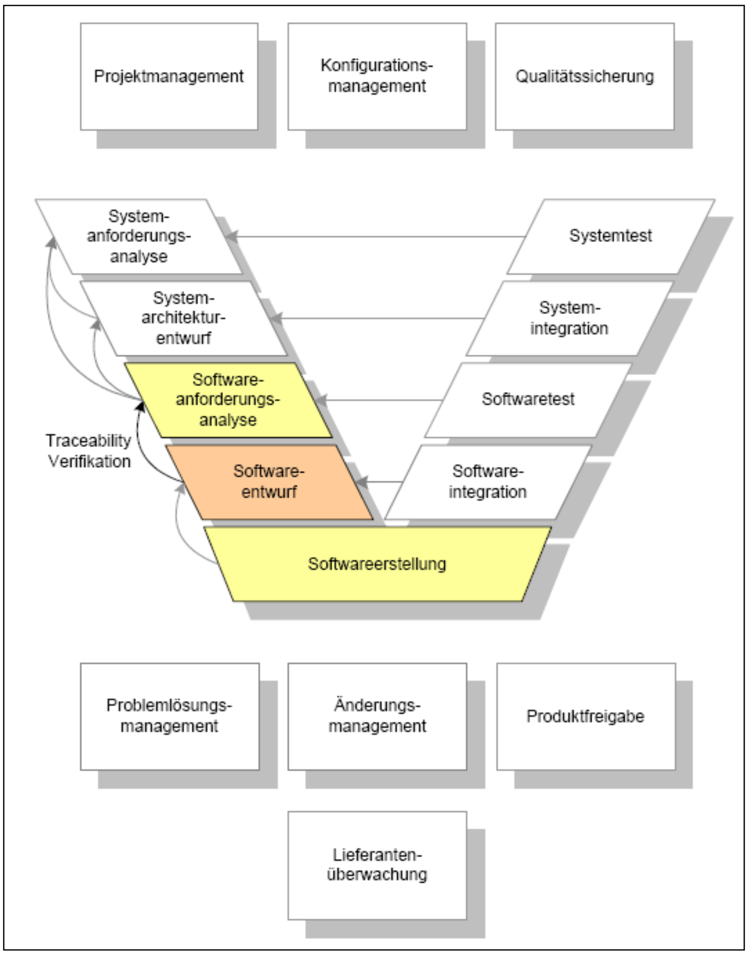

Figure 45: Process model[99]

The importance of the correct understanding of the system and its context and environment can be seen in Figure 46, too. The exact definition of the system and the intended use of it is the mandatory first step in ISO/DIS 26262, in which it is called item definition. To be able to make the item definition, all internal and external requirements from all stakeholders have to be listed. This includes not only the functional requirements of the customer but also all non-functional, legal or internal requirements like reuse of components.

[99] MAGNA E-Car

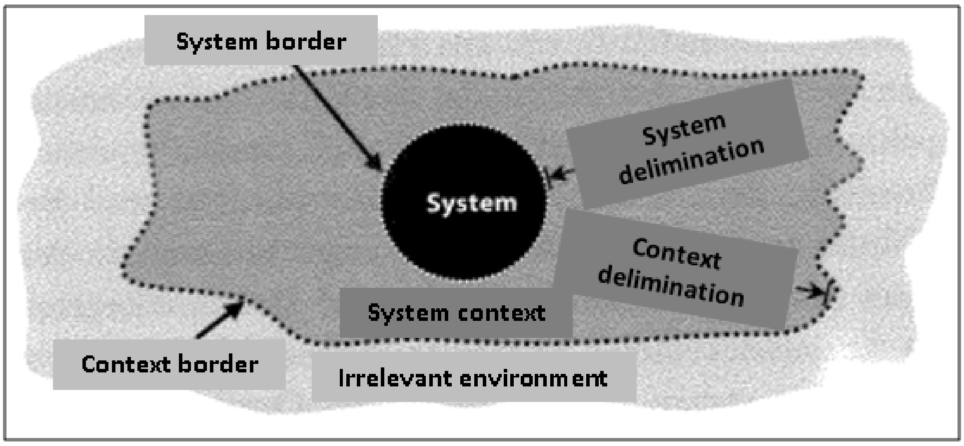

Figure 46: System and system context[100]

Based on the item definition the Hazard and Risk Analysis is done which leads to the safety classification of the system behavior and to safety targets. The safety targets are used for the establishment of a functional safety concept and ultimately lead to the safety functions of the system. As functional safety is in the responsibility of a dedicated department, these activities are driven and documented by members of this department. Finally the safety functions have to be integrated in the system specification as a separate chapter.
The safety related documents are based on templates and are prepared for partial reuse. The outcome of safety related activities has to be used as input for many activities in the standard development. Therefore both are heavily cross-linked and similar content is kept repeatedly. Sometimes several documents have to be updated in case of change and there is always a danger of contradictoriness. This area will be improved by the introduction of an additional level of requirements and design which is described in Figure 48.

3.2.2 Requirements structure

The current requirements structure consist of customer requirements and three internal levels. The levels are:

- Customer requirements: can be input for system or subsystem (component) level
- System level: delivered product

[100] Pohl/Rupp (2009), p.21

- Subsystem level: component of the delivered product according system design
- Module level: established only for software

As already mentioned the development starts with the item definition which is based on the whole input of all relevant stakeholders. At project start it is not always possible to get the needed information completely. This is especially the case for innovative products and products with limited customer expertise. Therefore it can happen that the initial specification includes some fuzzy aspects regarding interfaces, functions or quality. These fuzzy aspects are eliminated during the development iterations. Any resulting changes are processed according the change management process and an impact analysis is done whenever necessary.
In contrast to the solution of MAGNA Powertrain this solutions offers the possibility to link customer requirements either to system requirements or to subsystem requirements. This structure and the possible links are shown in Figure 47.

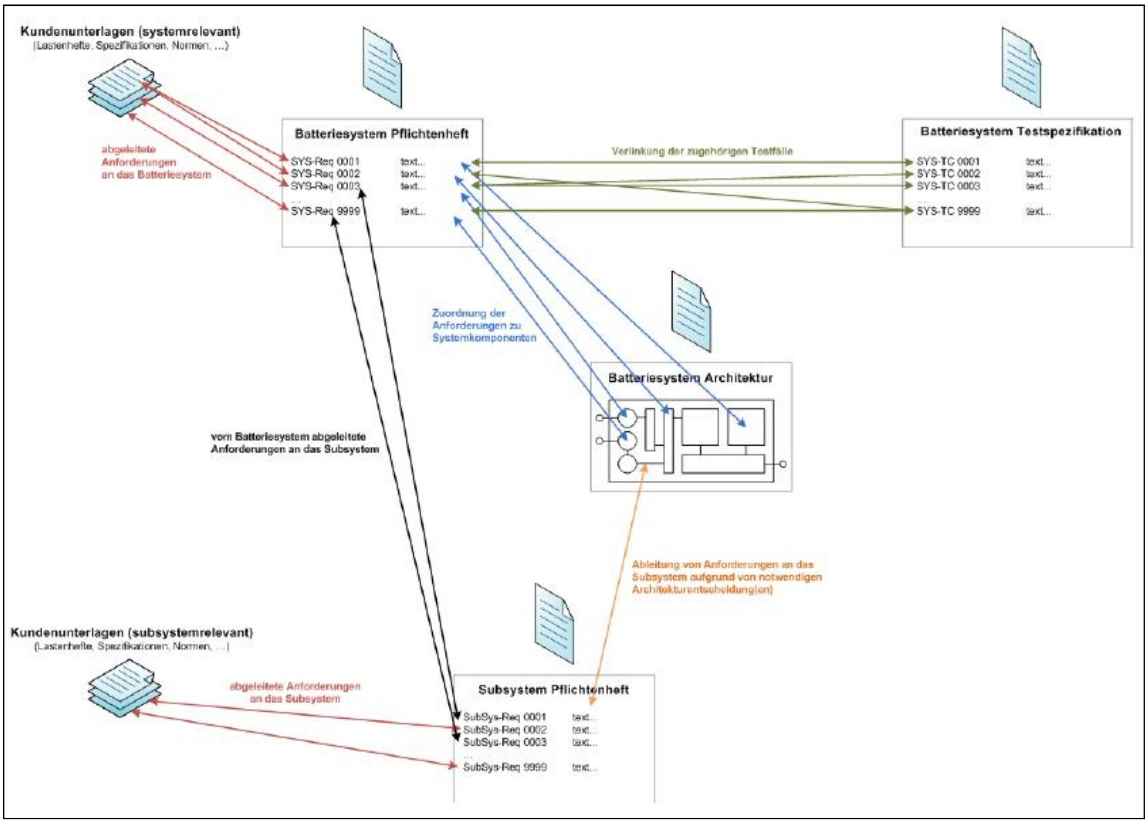

Figure 47: Requirements structure and possible traces[101]

[101] MAGNA E-Car

The advantage of this structure is a reduction of effort. All OEMs deliver very detailed specification in specific areas. One example are interface specifications. To ensure that the components fit seamless into the vehicle the customers provide the needed information like package for mechanical components and communication interface for controlled equipment right from the start. The communication interface may include some hundred signals which can be found on the connected vehicle bus system. Also the electromagnetic compatibility is accurately specified, mostly in a separate document. In addition it is quite common for some customers that their parts of their specifications are directed to software functionality. If it is possible for subsystems to link directly to these requirements the system level can be kept lean and it is easy to prove to the customer that all his requirements are met. On the other hand safety critical development always requires traceability for all safety critical functions starting at the system level. The reason is to ensure full evidence that in any case all affected elements of a system are considered during initial development and later change. This leads to a mandatory system requirement for all safety relevant subsystem requirements, even if specified already by the customer.

Customer specifications are imported in DOORS. System specifications, system design and subsystem specifications are written in DOORS. Only software is further broken down in the requirements management system. Software design is also written in DOORS. Software is developed with the programming language C. The software module specification is directly written in the comment section of the source files and extracted to HTML by a tool called DOXYGEN. Thus a software module requirements specification in DOORS can be avoided and the documentation effort can be kept lean. Prerequisite for this strategy is that the code and the comments are independent. Traceability is ensured by indication of the DOORS ID of the implemented requirements.

In contrast to MKS, where each element can be tracked with a workflow, DOORS has a document oriented workflow structure. The single requirements are dedicated to documents and the whole document is released at certain milestones like the freeze points in the project lifecycle model. The state of each requirement is indicated as an attribute within this document. Any change of a requirement after the last release is indicated with color. DOORS allows an easy import of customer specifications which are written either in MSWord or DOORS. The document oriented mode of operation is very similar to MSWord and the processing of OLE objects is supported in an adequate way.

For future projects an additional requirements and design level between system requirements and subsystem requirements is planned to be implemented. The structure with the new elements as 'low level system specification and design' highlighted in the grey box is shown in Figure 48.

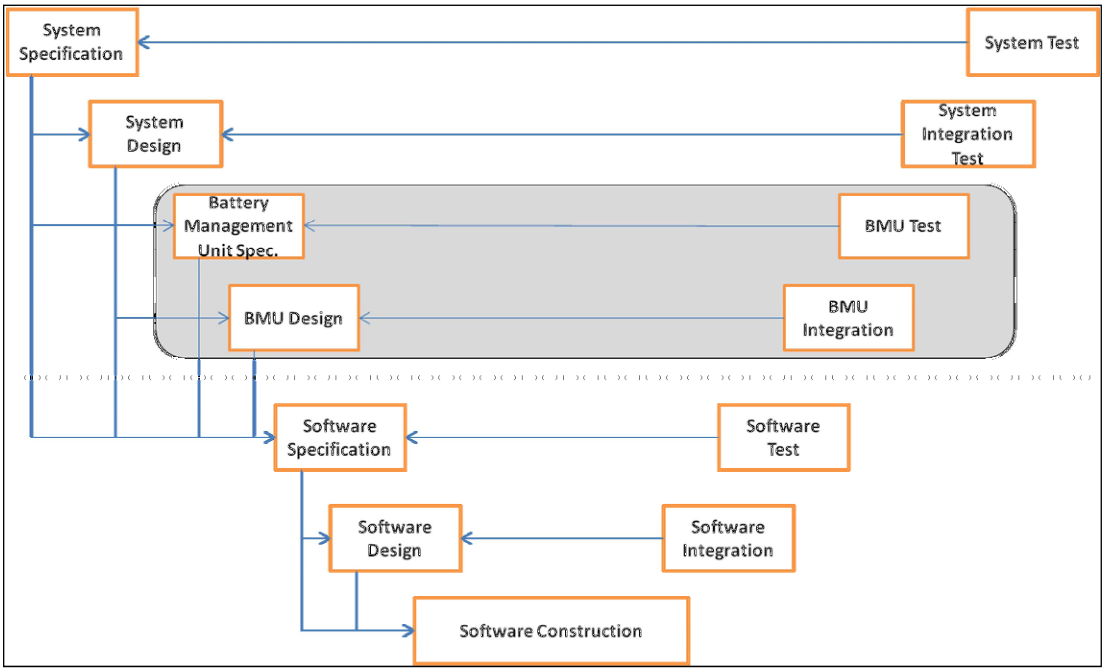

Figure 48: Planed requirements structure

The implementation of a second level will lead to some additional effort which is outbalanced by some advantages. In automotive engineering it is common that customers get the system requirement specification in written form, whereas all other documents are only available for inspection. If the system requirement specification can be limited to general functionality of the product and all know-how critical content can be transferred to other documents like the battery management specification it is much easier to protect intellectual property. The battery management unit consists of the core elements developed by MAGNA E-car and shall be documented in a way to support reuse of components. Other elements of the system design are matter of make or buy decisions and need to have specifications which can be provided to potential suppliers. The interfaces of these components have to be defined carefully at the beginning of the project in the system design to avoid effort later. Another intention in the new structure is the total integration of safety related content in the documents. Instead of safety related chapters the safety critical functionality will be indicated with a specific attribute if needed. The safety relevant aspects in the design will be marked in the system, battery management unit and software design document. Overall this shall support not only the reuse of components but also of functionality within the product and documentation by simultaneously reducing the effort of documentation for similar, subsequent projects.

3.2.3 Link between requirements management and test management

Test management is done in a similar way like in Lannach. The test cases for system and subsystem level are grouped to specific documents in DOORS. This enables again the simple check if all requirements have at least one test case. The overall status is again monitored with queries. Test cases for software modules are checked with MSExcel.
A difference can be seen in HIL testing. Whereas Lannach has a nearly 100% automation of test cases on the HIL test bench automation is lower here. The reduction of effort for automatic testing is set in contrast to additional effort for creation of automated test cases leading to a mix of both and longer test phases per release.

3.2.4 Summary

In this solution the focus on safety is much more pronounced. As all projects will be safety critical the needed activities according ISO/DIS 26262 are included in the standard processes. By now the documentation effort is rather high because of separate but overlapping and thus redundant kept content for standard and safety critical functions. This problem is already addressed and a new structure will deliver also various advantages like the support of reuse and know-how protection.
Requirements management is done with DOORS. Traceability is ensured and user acceptance is fair. An additional requirements level is to be introduced but the above mentioned advantages pay off the additional effort. To ensure reusability in the intended way it will be necessary to force the subsystem level to draw a link to the system level. The direct link to a customer requirement can be used optional. The documentation of the software module requirements is lean.

The requirements of the automotive standards in the area of requirements management are fulfilled. DOORS has advantages in importing customer specifications, processing OLE objects and user acceptance but is a dedicated tool for requirements management. Other related activities like change management and problem resolution management have to be supported with other tools.

3.3 Case Study MAGNA Powertrain Electronics, Sailauf

MAGNA Electronics was an own MAGNA group until it was recently split into two parts. One of them was integrated into MAGNA Powertrain, the other part was integrated into MAGNA E-car. Sailauf was the main European engineering site of this former unit. Several small engineering departments, most of them attached to production sites, were served by Sailauf with infrastructure and processes. Sailauf is still a service provider for those small engineering sites, reducing administration effort and supporting with expertise. Like in Lannach the products were not classified as safety critical in the past, but by increasing functionality some new and innovative products are now considered as safety critical.

3.3.1 General Overview

The quality management system starts with a very high level overview similar to what can be found in ISO/TS 16949. This general process is refined in the second level of the value adding processes to a stage gate process which can be found in Figure 49.

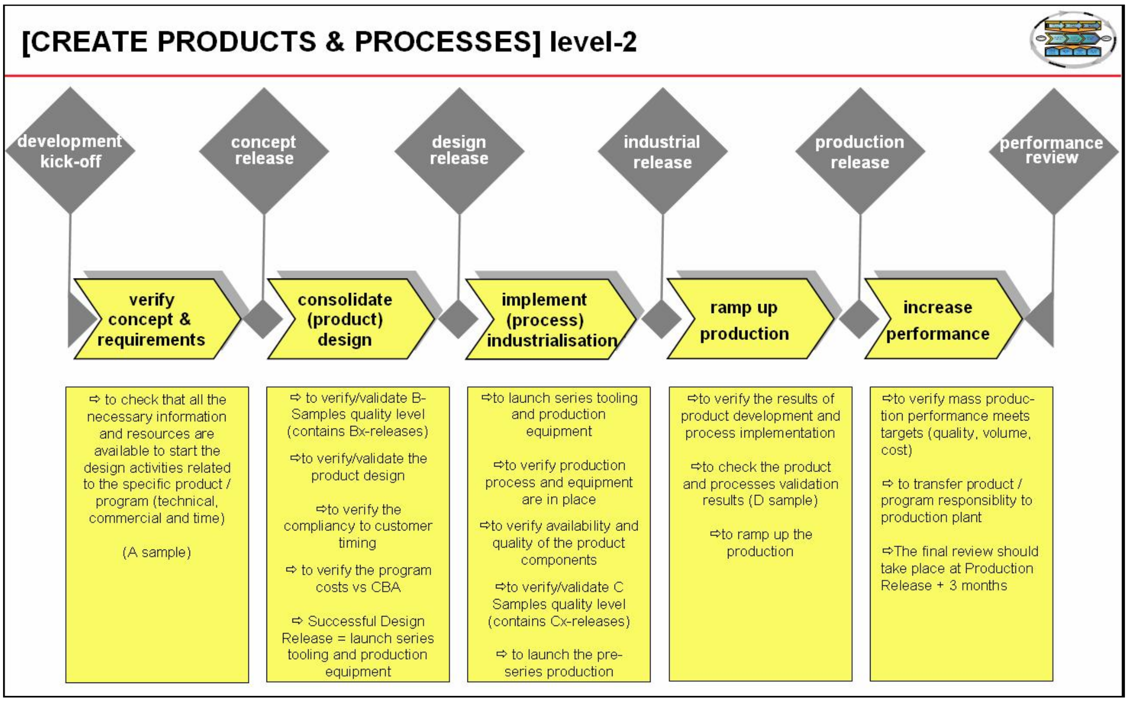

Figure 49: Level 2 process "Create Products and Processes"[102]

[102] MAGNA Powertrain

In contrast to most other MAGNA units MAGNA Powertrain Electronics is using samples at the various gates instead of prototype generations. This indicates that the electronic and software content is dominating the product development and production processes. The V-model is the more detailed description of the development process and considered as process level 3. The V itself follows once again directly the process definition of Automotive SPICE, see Figure 50.

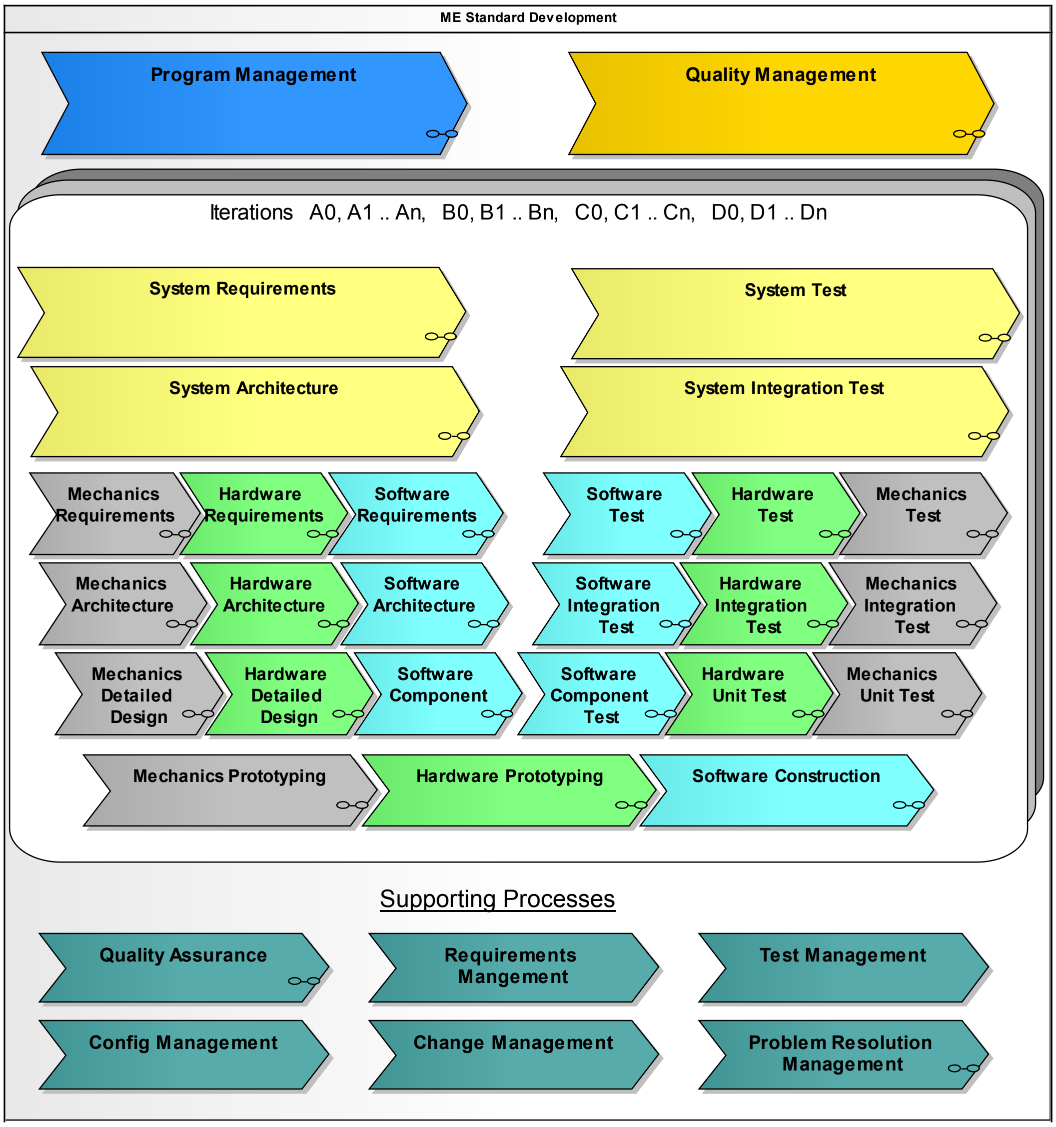

Figure 50: V-model[103]

Although the general structure of the V-Model is identical to the former case studies, some specific characteristics are included. The subsystem level is internal called technology level and especially mechanics is further split in

103 MAGNA Powertrain

products where this makes sense. The iterative character of the development is denoted with the potential high number of samples in the line directly above the V-Model. The supporting processes include one dedicated process for requirements management and test management besides the processes from the HIS scope.

The engineering processes are color coded. This color code can be found in the V-Model with yellow for the system level, grey, green and light blue for the subsystem and module level, a specific color for each management process and a common color for all supporting processes. This code is also used in the turtle diagram of each process, for indication which process has to deliver inputs or is getting outputs as well as for all involved roles. Figure 51 shows one example of such a turtle diagram.

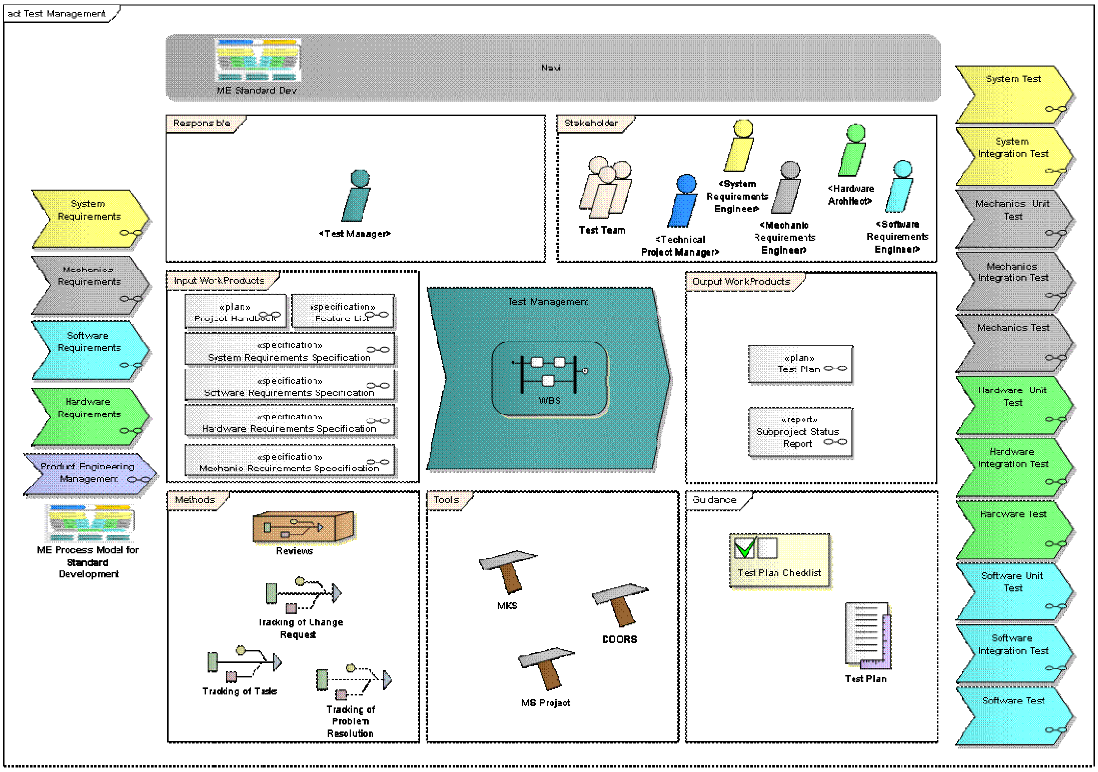

Figure 51: Turtle diagram of test management[104]

The turtle diagram is a common high level overview of a process in the area of quality management describing the process environment. The process itself is shown as a black box in the centre, head and tail of the turtle consist of input and output. The legs are represented by the involved roles, description of necessary tools and used methods, guidelines and templates.

[104] MAGNA Powertrain

Although turtle diagrams are very common for high level processes in different variants it is still not standard to use them in low level process descriptions. The combination of a turtle diagram with the strict usage of a color code fosters convenience.

3.3.2 Requirements management structure

As already shown in the V-model there are again four levels of requirements:

- Customer requirements
- System requirements
- Subsystem requirements: also called technology requirements
- Module requirements: also called technology component requirements

Although usually a relatively high number of customer requirements are imported in DOORS and their processing is monitored, the main communication with the customer is based on feature lists. A product has to fulfill a small double digit number of system features which the customer is really interested in. From each feature a number of dependent system requirements is derived with a factor of ten or even more. The project planning rests on the needed implementation schedule of the single features. The possible links between this levels are shown in Figure 52.

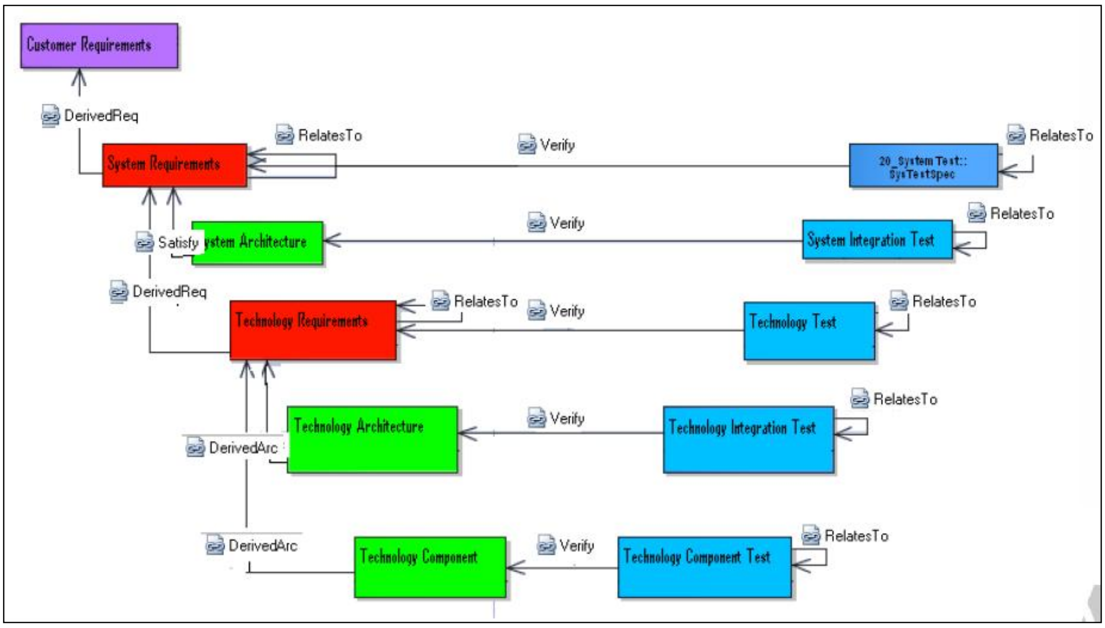

Figure 52: Structure of requirements management[105]

[105] MAGNA Powertrain

The customer requirements are imported to DOORS and the system requirements are derived from the customer requirements. In contrast to the solution at MAGNA E-car in Thondorf all customer requirements have to be linked exclusively to the system level. The system level requirements are used as source for subsystem requirements and system architecture requirements. System and technology requirements are specified in DOORS and the architectural design is specified in an UML tool called Enterprise Architect. Traceability between requirements and architecture is ensured with a table in which each UML element's ID is mapped to all requirements which are satisfied within this module. This pattern relates to system requirements and system design as well as to technology requirements and technology architecture. Yet technology component requirements are only specified in detail for software.

Technology requirements in the area of mechanics are further split for some products. The reason for this split is that the single documents are referring to clearly self-contained subsystems of different faculties which are in the area of responsibility of different departments. If a subsystem is purchased from a supplier the technology requirement specification of this subsystem is also used as technical specification for the supplier.

The use of the UML design model is unique within MAGNA. The same model is used in different layers of abstraction for the whole development process. The system design is a rough block diagram showing interfaces and most technology components still as a black box. By stepping down level by level one will find at the end a software component specification similar to a model used for automatic code generation. Although coding is still done the conventional way - partly because the qualification of tool chains for automatic coding is difficult - a change to automatic coding could easily made. The UML Model includes dynamic characteristics of product in addition to the static architecture. E.g. one can find the state machine or sequence diagrams. The central role of the UML model always ensures that there are no contradictions between different levels and everybody is using the latest version. The textual description of software modules is avoided, too.

Probably the main advantage of the single UML model is the excellent possibility of reuse. The whole design is done in an object oriented way. Even though coding is done in C which is not an object oriented programming language, some techniques are adopted to benefit from the main advantages of object oriented languages, e.g. reuse and encapsulation. Essential for the ability to reuse software modules or even function flows is an appropriate design with dedicated relationships. An example is shown in Figure 53, where the different line types and line endings indicate a different kind of connection between the single elements.

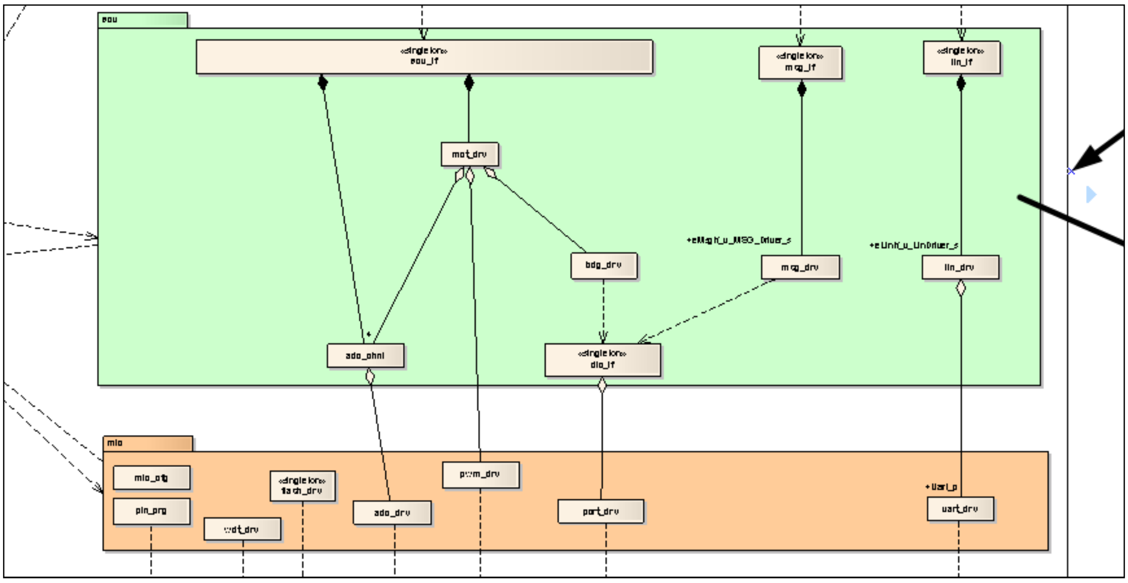

Figure 53: Excerpt of UML model[106]

For the monitoring of the status of requirements MKS is used additionally to DOORS. All requirements in DOORS are linked to tasks in MKS which can be observed better. MKS tasks are also used for the control of most requirement related activities like change management, problem resolution management and test management. So the progress of development can more or less be checked with queries in MKS. This aspect is similar to the implementation in the D&CCS division. This way it is possible to apply the tools according their strengths.

Figure 54 shows all elements and their connections for the processes software requirements analysis, software architecture and software construction. The graphic shall just illustrate dependencies and it is not the intention to be able to read details. The upper row of screenshots include the specification in DOORS linked to the tasks in MKS which are used for monitoring with dashboards like in the upper left corner. The central section shows the UML-model and some diagrams of dynamic behavior. In the upper left corner of the central section one can see the mapping table between

[106] MAGNA Powertrain

requirements in DOORS and elements in the UML-model. The MKS tasks are indirect linked via the DOORS requirements. The third row includes several depictions of one software module including a black box diagram with input and output variables and a detailed specification of an algorithm. The lowest row shows some screenshots with source code.

To enable developers to work efficient the establishment of tool interfaces is needed. After initial, manual drawing of links between elements in different tools the further processing is supported automatically.

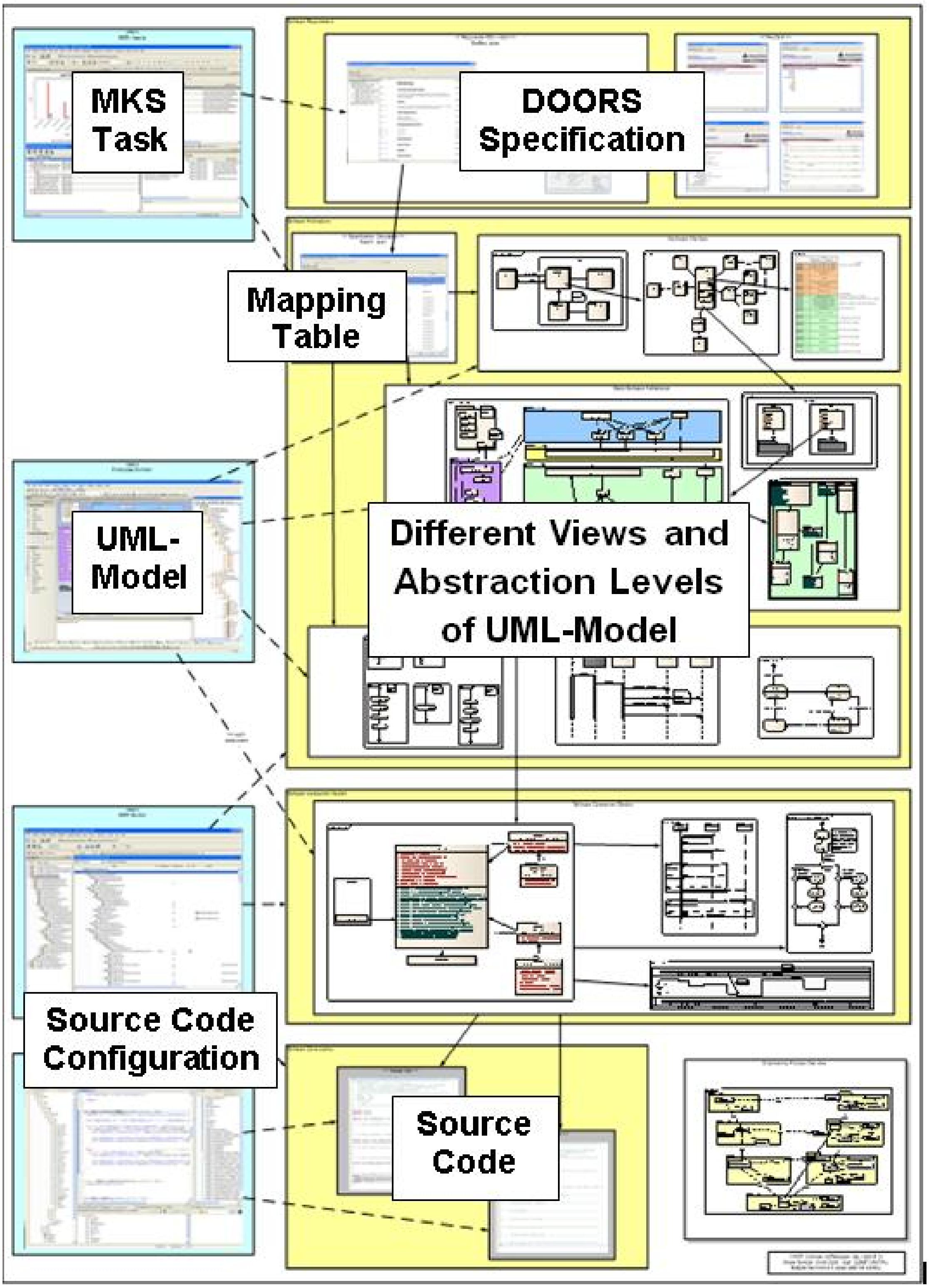

Figure 54: Structural overview about linked elements in software development[107]

[107] MAGNA Powertrain

3.3.3 Link between requirements management and test management

The test specification for the requirements on system level and technology level is made in DOORS and linked to the respective requirement. This again ensures that each requirement is covered at least with one test case. The test cases are then created and executed in a test tool called LabView. The results are then imported to DOORS with an interface.

As system and software architecture are designed exclusively as UML-model the definition of an integration strategy is the inevitable first step for test specification. The test cases are then created and executed in a test tool called Tessy.

The use of tools and the traceability is shown in Figure 55.

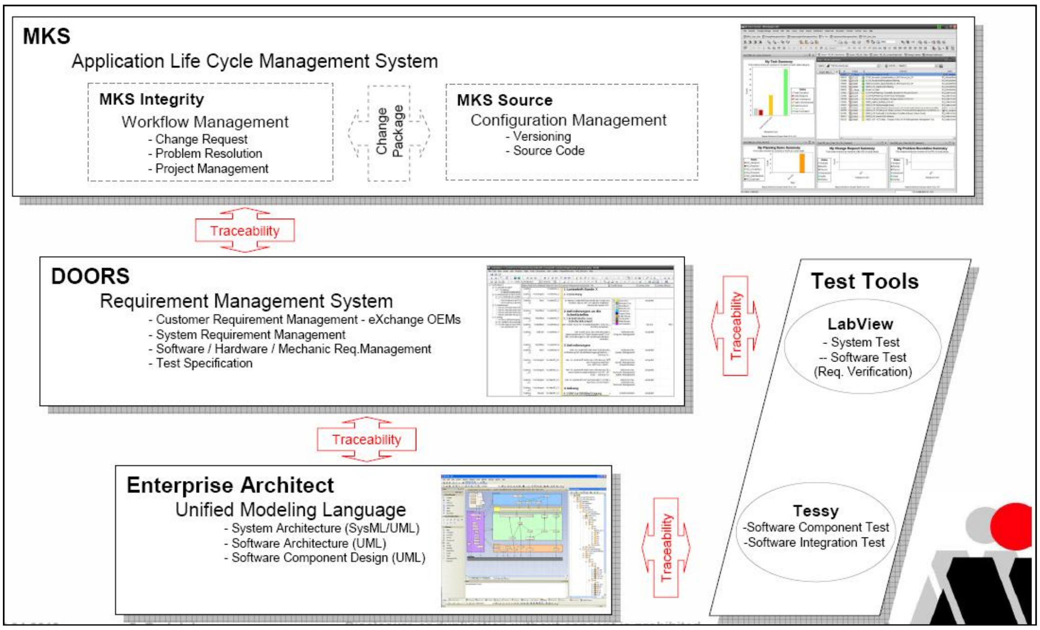

Figure 55: Traceability and tools[108]

Like in the case studies before it is easy to check for requirements that are missing any test case. The tool interfaces support also an automated monitoring of test progress.

The feature list which is agreed with the customer at project start supports the test activities, too. The regression test strategy is based on it as well as planning is enhanced in early project phases.

---

[108] MAGNA Powertrain

3.3.4 Summary

The realized solution with a combination of DOORS and MKS is rare, if not even unique. It provides the opportunity to use both tools according their strengths. This leads to a very user friendly development environment and high user acceptance. As requirements management related activities are included in MKS, again an easy and efficient monitoring of the development progress is possible. The drawback of additional costs for IT infrastructure, administration and interfaces is reduced by the fact, that requirements management is offered to other MAGNA engineering sites as a service provider.

Also a unique feature of this solution is the fundamental importance of the UML-model as architectural design for system and software as well as detailed specification for software modules. This offers the chance to eliminate an additional level of requirement documentation and to keep the documentation effort low. Moreover the usage of one single model with different layers of abstraction avoids contradictions and problems with out-dated versions. But the most important advantage of this solution is the excellent reusability not only for single components but even for features or functional flows over several components in the model. The model still provides further opportunities like automatic code generation or simulation.

As most projects are not yet safety critical, activities required by ISO/DIS 26262 are optional extensions to the standard processes. As the basic structure is again established referring Automotive SPICE this standard is fulfilled anyway.

3.4 Case Study MAGNA Mirrors & Closures, Sailauf

MAGNA Mirrors and Closures is a MAGNA group in which the electronically controlled functionality traditionally was less important. Until recently some of the products had no or very limited software controlled functions like interior or exterior mirrors. Other products are electronically controlled but compared to the products of the former case studies their complexity is lower. As in all areas of automotive engineering the trend to integrated systems with intelligent functionality can also be found here. A good example for this evolution is the rear view mirror. In the past it was a mechanical system controlled manually by the driver. Current rear view mirror modules are systems with lightening functions, a sensor to detect outside luminous intensity and automatic dimming to prevent the driver being dazzled. Although the products are not influencing driving dynamics, the typical area of safety critical features, ISO/DIS 26262 is relevant for selected products. Power window lifts include functions classified as safety critical because of several accidents in which children's necks have become trapped, leading to suffocation.

3.4.1 General Overview

The increasing requirements in the products cause a rapid growing demand of competence and manpower in the area of mechatronic engineering. Currently some customers are requesting compliance with process standards like Automotive SPICE, some do not. For safety critical functions the application of ISO/DIS 26262 is mandatory.
Together with the evolution of the products the organization is deployed towards meeting the new requirements. Currently tool supported requirements management is being implemented but the proceeding is not yet established for the whole engineering. The analysis in this case study is limited to one ongoing development project using tool supported requirements engineering.

3.4.2 Requirements management structure

As this case study is limited to one specific project, some of the characteristics apply only to this project. The development team is distributed to Sailauf and the scheduled production site. For full roll-out adaptations to a general valid structure have to be made.

As the project complexity is lower than in the former case studies the processing of requirements is simplified. The requirements management is done with mainly two requirements levels in DOORS, the customer requirements and an internal requirements level.
When customer requirements are received they are imported to DOORS. The requirements are categorized into the following types based on the responsibility.

- Customer
- Supplier
- MAGNA Closures - Mechanical
- MAGNA Closures - Electronics

Requirements which are in the scope of MAGNA Closures are moved to a separate DOORS document for further analysis, requirements referring only to the supplier are handed to the supplier.
Requirements are processed according the workflow in Figure 56. During the analysis an attribute is set to either mechanical or electronics, if just one subsystem is involved.

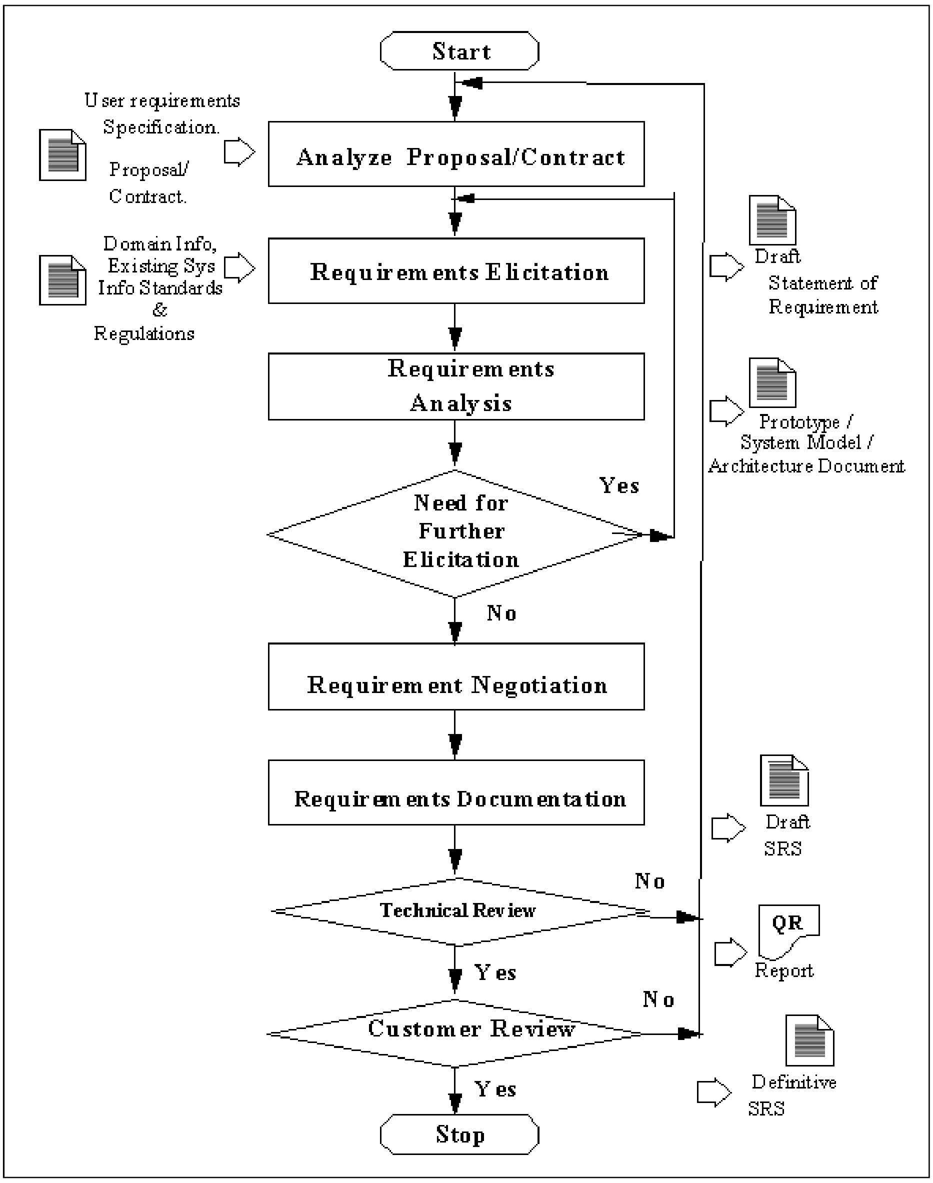

Figure 56: Requirements analysis workflow[109]

Requirements which are rejected have to be discussed with the customer and accepted requirements are scheduled for implementation. All accepted requirements are integrated in the draft system requirements specification which definitive released after a review with the customer. The system design is included in the safety concept, covering all needed aspects besides the safety relevant, too. The integration of safety design and system design avoids contradictoriness and effort for keeping information twice. The software design is done in two steps -the high level design and the low level

[109] MAGNA Mirrors & Closures

design in MSWord. The software module specification is done in MSWord, too.
The solution of one document for requirements of system and subsystem level was feasible because of relative simplicity of the product and a small number of requirements. The needed separation of requirements for the different subsystems is reached by usage of appropriate attributes. This makes it possible to extract the specification for a single subsystem of the product, but the dependence of system requirements and subsystem requirements is not obvious because of the missing link structure between those levels. This makes an impact analysis or the identification of interactions more difficult in case of product changes and problems.

3.4.3 Link between requirements management and test management

Test management again is implemented by the use of attributes. Test methods are defined for the project and all test methods which are to be utilized are implemented as separate attribute in the requirements document. For each requirement it is then decided if a test case has to be created based on the specific test method or not. For all defined tests it is further documented if a test case is already specified and finally the test result is set. All this is done manually thus generating some effort although the test documentation is integrated in the requirements specification.
Like in all previous case studies it is easy to detect requirements missing any test case. The absence of interfaces for importing test results leads to potential delay of information update, but as long as the number of test cases is limited the establishment of an interface is not justified.

3.4.4 Summary

The implemented solution has some benefits regarding documentation effort. It might also be an advantage that the danger of information loss or contradictions is much lower if information is kept within on document, especially if development is done by a distributed team. This refers to the design documentation in particular.
Although the effort is reduced by using one document for system and subsystem requirements this solutions seems to be suitable only for small teams. With the use of appropriate attributes it is possible to fulfill most requirements of the standards, but dependencies of requirements are hardly

evident. This would be a weak area in customer assessments and a drawback for reuse.

The close integration between safety management and requirements might be supported by small team size and low complexity of the product. Still it is an advantage.

Altogether it is necessary to make adaptations to the implemented structure before a general roll out should be considered.

3.5 Case Study MAGNA Powertrain Electronics, Rochester Hills

The general situation of MAGNA Powertrain Electronics is already described at the beginning of case study 3. All comments related to organizational changes and trends for products are applicable to this case study, too.

3.5.1 General overview

The engineering in North America has its own engineering processes and has established an independent requirements management system. Nevertheless, products in the automotive industry are more and more developed for global platforms the processes have to comply also with the main requirements from global customers. Currently the existing product development process is reengineered. As some customers were asking for Automotive SPICE it is fully considered in this new version. The deployment of the new processes was almost finished by the time of analysis and the roll out will start at the beginning of 2011.

The new project lifecycle shows the integration of the V-model to the traditional stage gate process in an excellent way, see Figure 57. The design phase is done iterative according the engineering processes required by Automotive SPICE and is based on samples like in Chapter 3.3 Case Study MAGNA Powertrain Electronics, Sailauf. After completion of the design phase an additional iteration can be triggered by problems found in the validation phase. This is indicated with the backwards arrow. Also the supporting processes can be found in the bottom part of the model.

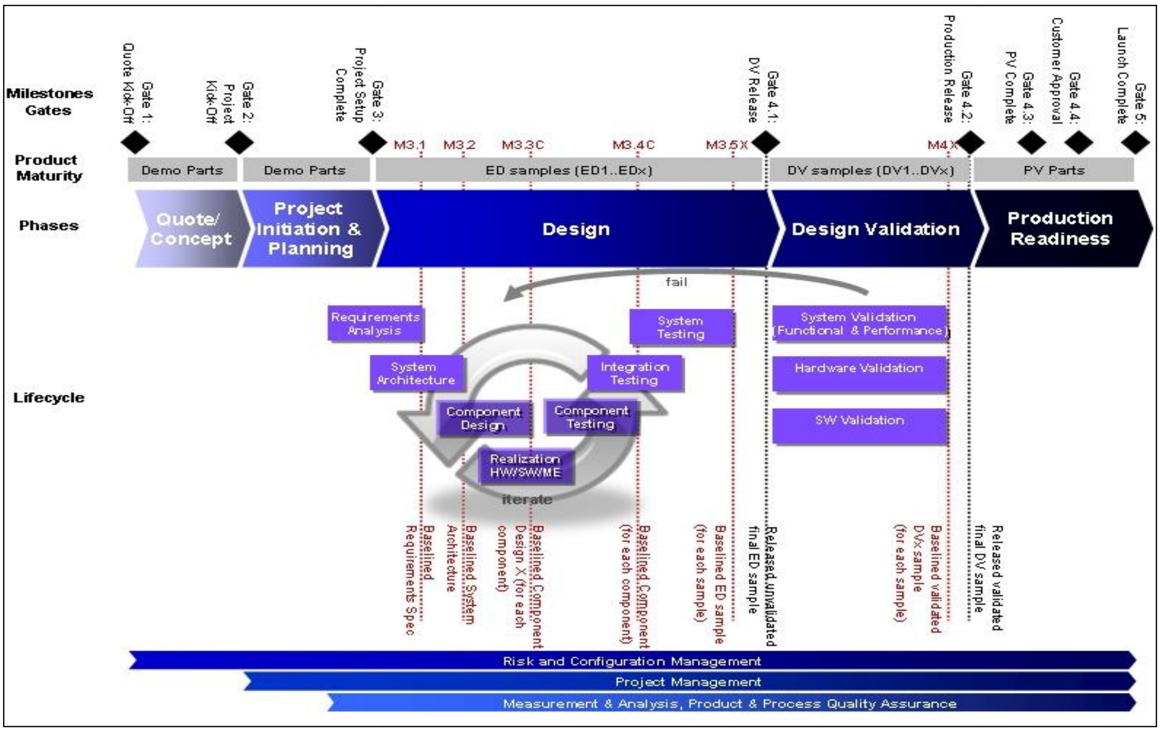

Figure 57: Project lifecycle model[110]

This general model is further refined in a high level activity map called discipline interface chart, see Figure 58. It includes the same phases and milestones like the project lifecycle model, but the general processes are replaced by the main activities of each stage. Different colors of the activities indicate different responsibilities.

110 MAGNA Powertrain

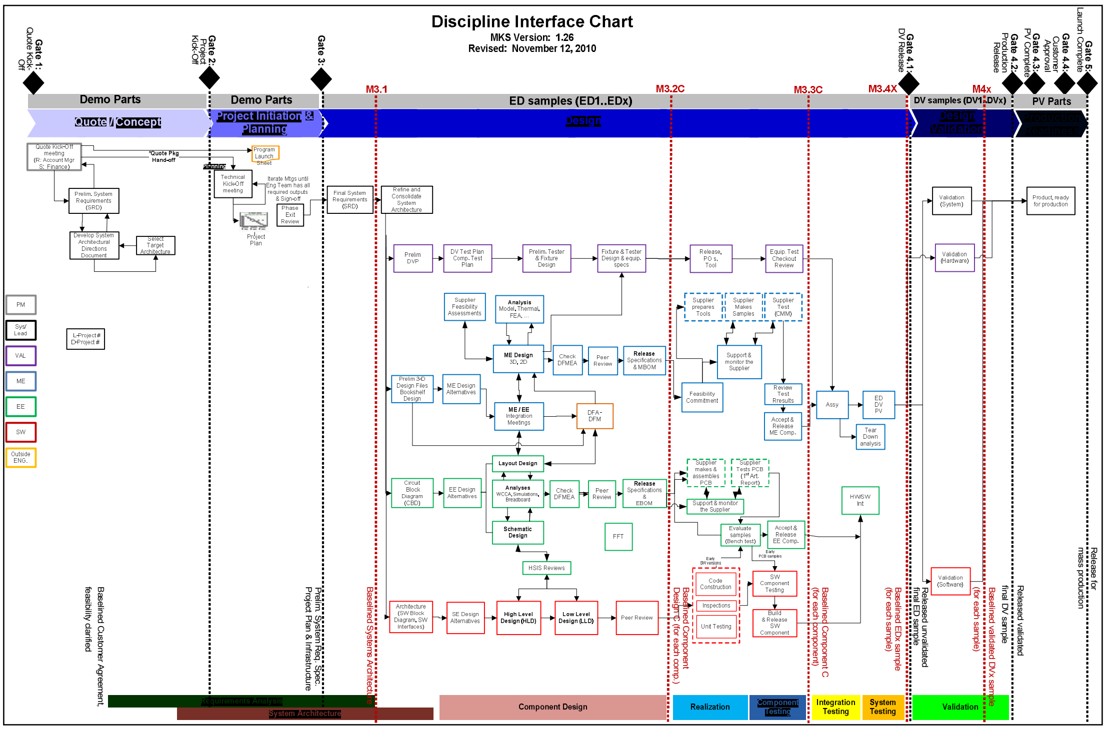

Figure 58: Discipline interface chart[111]

As the content of the single activities is know how related it is not intended to make the chard readable. Still it can be seen that between milestones M3.1 and M3.2C there are some interactions and dependencies between activities across all departments.

An additional point which I would like to draw attention to is the quote phase in the upper left corner. The activities and loops here are dealing with the preliminary product architecture thus supporting requirements of ISO/DIS 26262.

3.5.2 Requirements management structure

Requirements management is done with MKS and there are 4 levels of requirements used:

- Customer requirements
- System requirements
- Subsystem requirements: called discipline specific requirements
- SW Module Requirements

[111] MAGNA Powertrain

Customer requirements are imported to MKS. Although it is a common assumption that it is not yet possible to have automatic data exchange between MKS and DOORS, the implemented solution disproves this. The exchange is based on a standardized format called RIF (Requirements Interchange Format) which is available in a pre-released version and should be applicable for all customers in future. This is important as DOORS is most prevalent among OEMs.

The system level requirements are then derived from the customer requirements and the acceptance criteria are defined for each system requirement. After an internal review of the system requirements specification a validation of the customer should be attained, which means that the customer should sign off the specification. Finally the specification is released in the requirements management tool which leads to the first baseline. This pattern is also shown in Figure 59. Due to the fact that customers sometimes are not able to specify the full functionality right from the start of a project or have a demand for later changes it is common that the system specification has to be established with several iterations called baselines. In this case baseline 1 is usually contracted and all subsequent baselines are matter of change management.

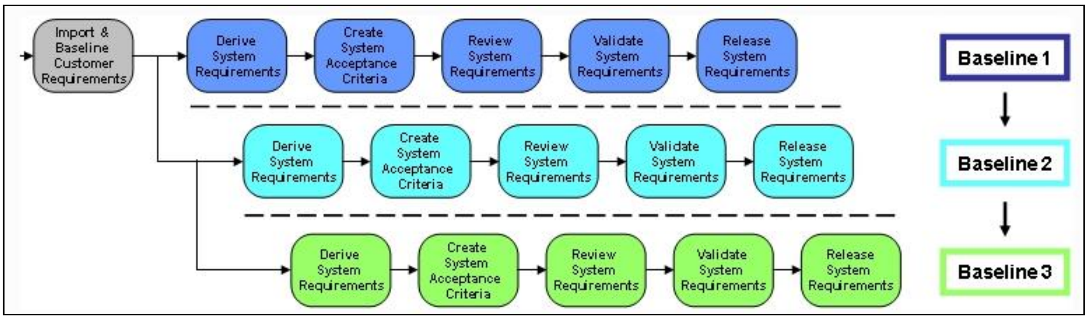

Figure 59: System requirements baselines[112]

Every internal requirement follows a lifecycle which is indicated by the status of the requirement. Possible transitions are depending on the current status and the role a person occupies. The lifecycle of requirements including transitions is shown in Figure 60.

[112] MAGNA Powertrain

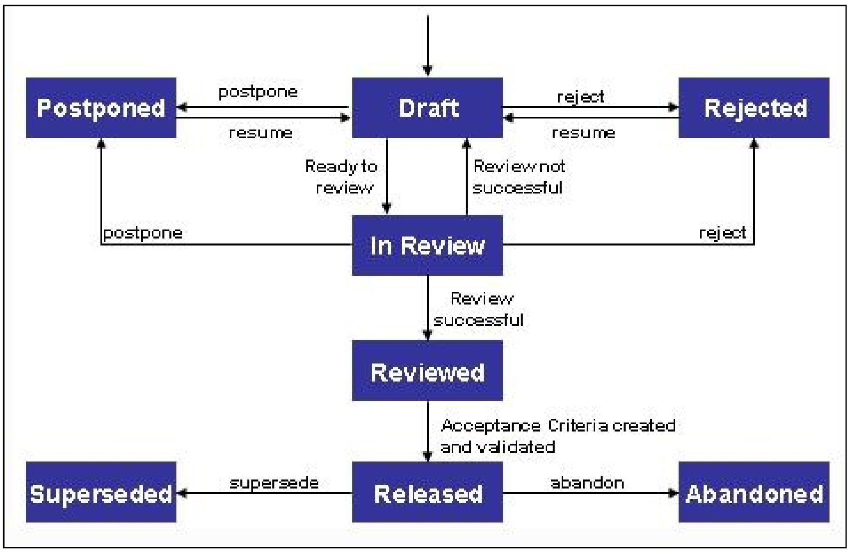

Figure 60: Lifecycle of requirements[113]

The integration of engineering activities and requirements status changes leads to a model in which the progress of the project can be followed by monitoring the status of requirements. Figure 61 shows this integration graphically.

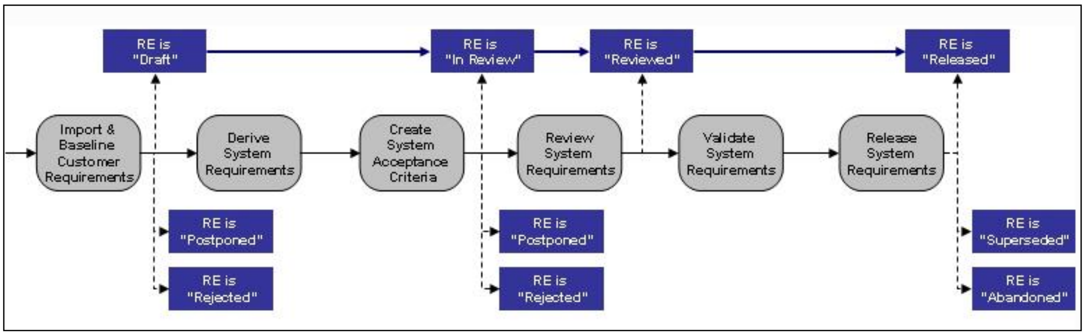

Figure 61: Integration of engineering activities and requirements lifecycle[114]

Simultaneous to creation of the system requirements specification the preliminary product architecture is refined to the effective system design. The system requirements are allocated to the system architecture and discipline specific requirements are derived from the system requirements. This structure enables only the linkage of requirements of two adjoined requirement levels. Figure 62 shows in addition that the system architecture also can create requirements on the discipline specific level.

---

[113] MAGNA Powertrain

[114] MAGNA Powertrain

The system architecture is created in MSWord and the requirements IDs of MKS are allocated to the affected subsystems. Completeness is assured with reviews.

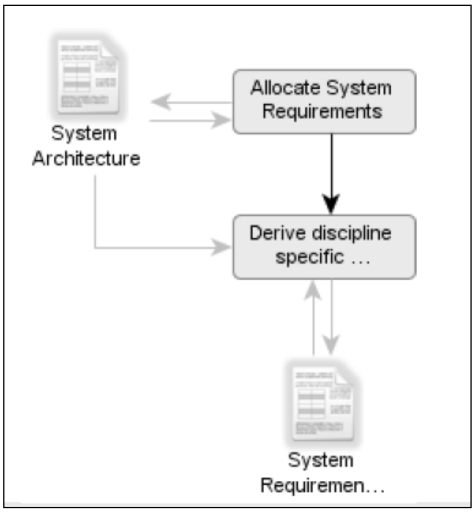

Figure 62: Link structure on system level[115]

Besides the obvious components like electrical, software, mechanical and optics there is a further category for domain specific requirements. Production is listed separately and thus supporting the requirements of ISO/DIS 26262 that already engineering has to define verification activities to meet safety targets throughout the whole project lifecycle.

The software design is then created with MSWord based on the discipline specific requirements for software, first as high level design and then as low level design. Later each software requirement is allocated to a software module. Also internal interfaces like HSI (Hardware Software Interface) are to be found on this level. Finally the module specification, covering all requirements allocated to the respective module, is created in MSWord.

To protect intellectual property a simple, but effective guideline is implemented in the requirements engineering policy: *'As a general rule, everything what is shareable with the customer belongs probably to the requirements specification. Information which is not shareable belongs to MAGNA internal documents (e.g. design specification, etc.).'*[116]

[115] MAGNA Powertrain

[116] MAGNA Powertrain, MAGNA Electronics Requirements Engineering Policy

3.5.3 Link between requirements management and test management

As all defining engineering processes and requirements management were reengineered just right now the link between requirements management and test management is not yet fully automated. The current solution is to export requirements to MS Excel and to track test results in a table. An implementation of the Test Suite of MKS and an interface between requirements management and test management are under evaluation. The chosen extension will be established shortly after the decision.

3.5.4 Summary

The project lifecycle model gives an excellent overview of the interaction of the linear overall stage gate process and the iterative engineering approach. In combination with the discipline interface chart it is easy to get a good impression of the main work packages, their interactions and which department is responsible. This enables new employees to orient themselves very fast and supports internal and external cooperation.
An extraordinary feature of the established requirements management solution with MKS is the automatic exchange of requirements specifications with customers using DOORS. In most implementations of MKS it is seen as a weakness that the import of customer requirements is time consuming and they lack DOORS exchange possibilities at all.
A big advantage is the fact that also other activities like change management and problem resolution management and task tracking can be done within a single tool as in the other case studies in which MKS is implemented. Again the monitoring of engineering relevant topics is easy and the needed dashboards are under progress.
As the new PDP was just deployed and is about to be rolled out it still has to proof that it works well. Probably it will need some minor adaptations after getting feedback from project teams. The processes are based on experience with CMMI and as well enriched with as aligned according to requirements of Automotive SPICE. They include some aspects related to safety but several safety related activities have to be amended if needed. Anyway the solution is capable to fulfill the requirements of Automotive

SPICE, which is not common by now for US-based engineering, and ISO/DIS 26262.

3.6 Conclusion of Case Studies

MAGNA culture with decentralized responsibilities led to very different solutions for requirements management not only between MAGNA groups but also within a MAGNA group. This enables single engineering sites to establish processes and tools which support their individual needs best. On the other hand it hinders the sites to utilize synergies and benefit from expertise already gathered within MAGNA.

During analysis of the case studies many valuable aspects could be identified which have to be considered in the definition of an efficient requirements management system considering automotive standards. These inputs cover a wide range of points which have to be taken into account, reaching from general targets like protection of intellectual property to technical details like tool interfaces.

Out of the advantages of the case studies it is possible to identify some highlights which are unique within the analyzed solutions. MAGNA Powertrain D&CCS, Lannach, offers the ability to work with requirements even when you are offline. MAGNA E-Car, Thondorf, has a good integration of safety and protects know-how of the its core component battery management unit. MAGNA Powertrain Electronics in Sailauf has put the design in the center and can benefit from the UML-model in several ways. Magna Mirrors & Closures, Sailauf, has a very lean pilot implementation which could lead to a reflection on required complexity of processes and tools. Finally MAGNA Powertrain Electronics in Rochester Hills has a good lifecycle model in which the V-model is well aligned with the overall process.

4 Requirements Management Business Model

In this chapter a business model for requirements management shall be deployed. It is of course based on best practices from the former case studies Furthermore it implies input from literature research, personnel professional experience and ideas gathered during Automotive SPICE assessments of suppliers. It will at least cover all major requirements of Automotive SPICE and ISO/DIS 26262. As standards in the area of processes are established in a way that they are open for different solutions, the deployed business model can only be an exemplar model.

4.1 Assumptions and Preconditions

The business model shall describe an effective and efficient solution for requirements management of an automotive supplier developing complex *‘systems comprised of electrical, electronic and software elements that provide safety-related functions’*[117]. It is assumed that it is possible to establish tool supported requirements management which includes the budget for needed IT infrastructure, tool deployment, licenses and tool administration. Some activities need specific skills which either have to be present in an organization or should be acquired parallel to the implementation of the model.
As each existing organization has a different initial situation based on experience, size, product complexity, IT-infrastructure, etc., it is not possible to make allowance for all this influences. Therefore a precondition for implementation of this model is either the ability to make a rigorous change to build a new business right from the start. On the other hand it is possible to stick with existing structures and pick out beneficial aspects of the model which can be integrated in established processes.

4.2 Project Lifecycle

According the well known principle ‘structure follows strategy’ the first step is to define the overall project lifecycle. This includes all major steps of product realization as already identified in ISO 9001 or ISO/TS 16949. These steps have to be transformed into a process with milestones representing the

[117] ISO/DIS 26262-1, p.iv

strategy of the company. At milestones the appropriate panel, in most cases the project steering committee, has to approve the current progress or to take action. This high level process is described in linear phases and includes all departments. In reality proceeding is not as simple. Due to simultaneous engineering different teams work at the same time on different tasks and many tasks are fulfilled iterative. Although a software developer will not find his contribution to the project in this overview the linear stage-gate representation is still valid from a top management perspective and progress is evaluated at the gates. For efficient steering committee meetings it is necessary to standardize the main work packages and expected outcomes of a stage. The outcomes are rated against checklist criteria, summarized in a reporting and finally condensed to rating according traffic lights:

- Red: major risks, project success unlikely or decisions outside project scope needed
- Yellow: risks but counter measures already in place or problem can be solved by the project team
- Green: minor or no risks

The next step is the definition of the V-model for engineering. It has to cover the needs of Automotive SPICE and ISO/DIS 26262 and keep the effort as low as possible. In the relevant area of mechatronic systems MAGNA is Tier 1 supplier to OEMs. Even if different groups have products consisting of different components on subsystem level the basic structure of software, hardware and mechanics is comparable. The four levels of requirements as shown in Figure 63 apply to all groups.

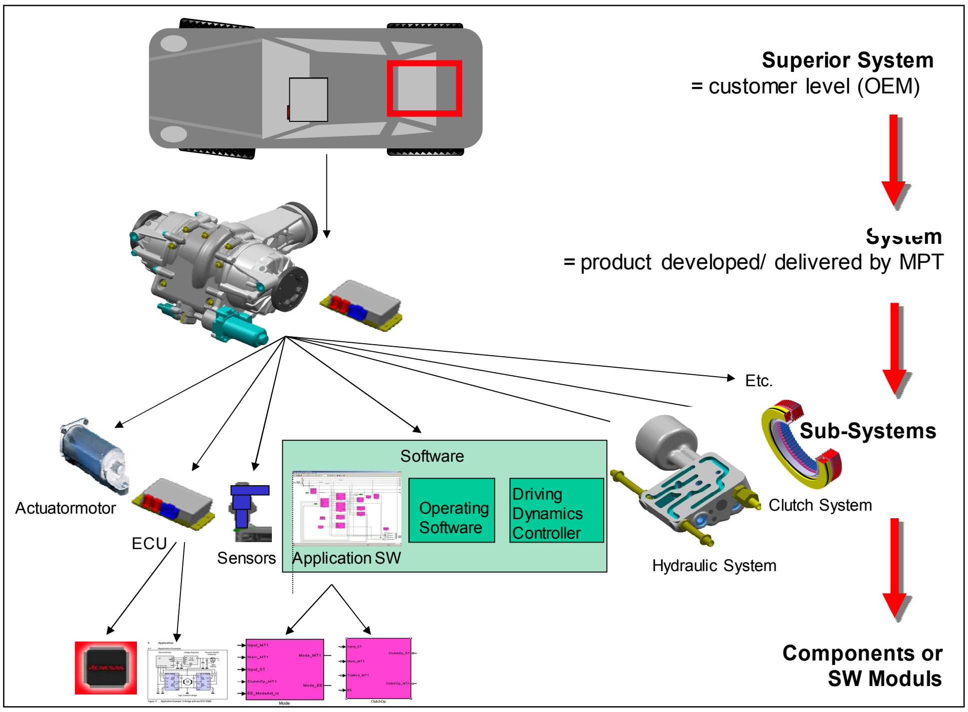

Figure 63: Requirements Levels[118]

The decision of one level of customer requirements and three internal levels arranged according the V-model leads to a graph like shown in Figure 65. The main states on different levels are already included and will be described in detail later.

Development itself is again a stage-gate process. Therefore the V-model should be supplemented with the according development gates and an indication for iterations should be added. Once the specific model is defined it has to be aligned with milestones from project management. A good synchronisation of development cycles and general control cycles is seen as success factor for systems engineering.[119] An project lifecycle model created that way is shown in Figure 64.

---

[118] Spork/Pichler (2007), p. 2.5

[119] Cf. Stelzmann et al. (2010a), p.6.13

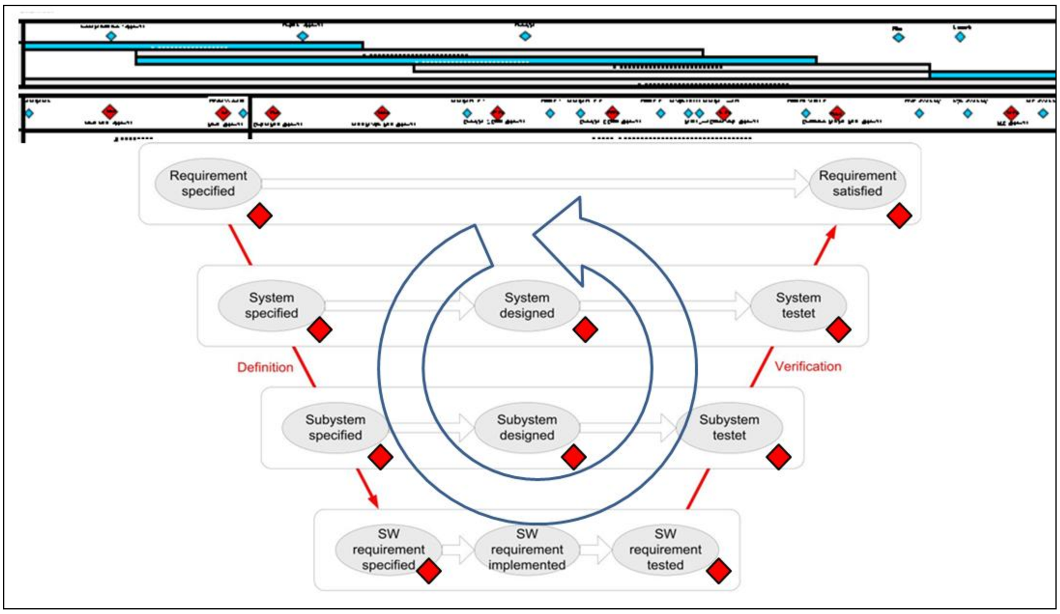

Figure 64: Project lifecycle model

To finalize the project lifecycle model a high level overview of project activities like in the Discipline Interface Chart in case study 5 should be created. In addition to the value that such a diagram provides for internal and external communication it can be used to identify the number of parallel streams and the main project interfaces. Already in 1968 the computer programmer Melvin Conway identified a close relationship between organizational structures and product design. According to Conway's law *'...organizations which design systems ... are constrained to produce designs which are copies of the communication structures of these organizations."*[120]The reason for this is the fact that communication is needed between designers of separate software modules in order to ensure that the interface works correctly. Therefore, the interface structure of a software system will reflect the social structure of the organization(s) that produced it. This characteristic was proven by a team of Harvard Business School researchers that revealed significant differences in modularity, consistent with a view that distributed teams tend to develop more modular products. If such a high level activity diagram displays a very high number of interfaces during the design phase it might be a warning sign for problems.

[120] Cf. Conway (1968), p.31

4.3 Structure of Requirements Management

It was already described in the case studies that the structure of requirements management has a big influence on effort and know-how protection. Any decision how the requirements management concept should look like has to be derived from internal goals. Currently available functionality of a specific tool can then be used for tool selection but should not be used as argument for structuring requirements management.

To limit documentation effort it is favorable to stick to three internal requirements levels. The level of customer requirements can be used for other requirements, too. This leads to the following structure as shown in Figure 65:

- Customer requirements
- System requirements
- Subsystem requirements
- Component requirements

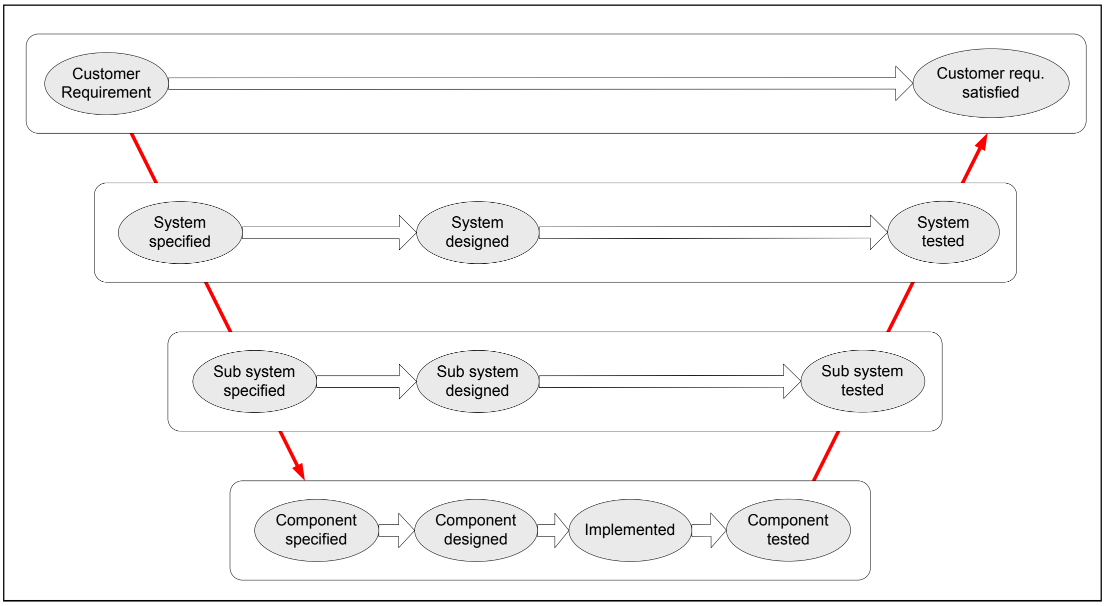

Figure 65: Simplified V-model[121]

The content of customer requirements can hardly be influenced. It is quite common that customer specifications are a mix of requirements for system, subsystem or software. In many cases customer specifications include at least some specific design solutions as requirements. Anyway it is important to consider careful which content has to be entered as customer

[121] Spork (2008), p. 13

requirements. This applies in particular if a customer has excessive specifications with redundancies or even contradictions and import is done automated. As each requirement in the requirements system has to be processed and thus creates effort it might be better to limit the number of customer requirements and to handle non functional content outside the requirements management system as far as possible.

Besides the customer requirements other requirements which are obligatory for the project can be specified here. This implies adherence to standards or laws as well as internal requirements like common parts with other projects.

All customer requirements should have derived system requirements, even if their content is clearly focused to a subsystem. On one hand this is required by the standards on the other hand it is the only way to prepare requirements management for re-use. A compromise to ensure increasing level of detail by stepping down requirements management could be, that a more general requirement is linked on system level and an optional link can be drawn to the subsystem or component level as add-on. One example for this strategy is to link a bus-message on system level and the affected signal on software level to a customer requirement including information about the possible range of values for a specific measurement. System requirements should describe functional properties and exclude any design solution if possible.

Subsystem requirements are either derived from system requirements or from system design. As already defined a direct link to a requirement on customer level is just available as option. The description should again be based on functional properties and exclude design solutions. The subsystem requirements for a specific subsystem can be used as specification for a supplier, too.

Component requirements are derived from subsystem software requirements and the software design. The guidelines for subsystem requirements apply again. If these requirements need an extra specification in a requirements management tool depends on the product and the coding strategy. In some cases it will be sufficient to use the lowest level of an UML-design as detailed software specification. If coding is done with automatic conversion of the UML-model to source code a separate specification is needed for all safety critical software modules. Component requirements are not mandatory in other areas beside safety critical electronical hardware requirements. As hardware is often purchased those requirements are usually found at the supplier.

At all requirements levels the specification should be arranged according features and functions. Requirements classified as safety critical should get the appropriate ASIL level as attribute but separate safety chapters should be avoided. This is also valid for system and software design. Experience shows that this separation creates problems during development because safety aspects have to be an integral part of the design. It is not possible to attach safety to an existing design if it was not considered adequate right from the start.

There is a close interaction between requirements and design. Most changes of the design will induce change to subsystem requirements even if the system specification stays the same. Requirements should specify functionality and design should describe a technical feasible solution. Requirements could be seen as black box description of a technical system whereas a design is the white box description. Hence know-how protection can be achieved by keeping design documents in the company

The creation of a single design for system and software level in UML with different abstraction layers is beneficial. This approach ensures that contradictions and redundant information keeping are avoided. The same time consistency of interfaces is assured. The UML model can be utilized for simulation in addition. A link between design and customer requirements should be avoided.

The workflow of requirements should be kept simple and the states of different levels should be identical as far as possible. The same name and equal expectations from the content supports common understanding in all departments and makes monitoring of progress easier.

The main states and their characteristics could be[122]:

Specified

- The requirement is distinct and testable
- The preliminary architectural design is created
- Main function flows are analyzed
- Interfaces are defined
- Verification criteria are defined
- Traceability of requirements is established

Designed

- The architectural design is fixed

[122] Cf. Spork (2008), p. 13

- All function flows are defined
- Detail design is established

Implemented

- Source code is generated (just relevant for software component requirements)

Tested

- Requirements are tested against verification criteria

Some other states like rejected, postponed, released, etc. might be implemented as needed. For monitoring of the development progress the above mentioned states are sufficient.

For safety critical projects it is necessary to create additional documentation. Most of it will lead to further documents, some can be integrated in existing work products. The definition of project scope according Automotive SPICE can be used as item definition as well. The safety requirements specification can be an excerpt of all requirements with a corresponding attribute.

The functional safety concept is based on hazard analysis and risk assessment. Functional safety requirements are derived from the functional safety concept and can be seen as an additional source for system requirements. Preliminary architectural assumptions serve as input for the system design. In automotive engineering the states of operation needed for the risk assessment and ASIL rating are identical for most functions. It is common to separate driving on highways, driving in rural area, driving in cities and special modes like service, parking or car wash. This leads to big table with different ASIL ratings for different functions in different operation modes. As each function has to be developed according the highest ASIL rating this table can be condensed to a small one which can be integrated to the system requirements specification, giving reason for the ASIL rating of the system requirements.

Another important work product is the HSI (Hardware Software Interface) specification. It is an interface specification which is clearly on system level according Automotive SPICE because it describes the relation between two subsystems. As it is a specification of a solution and not a functionality it should be part of the system design. The overview on interaction with HSI in ISO/DIS 26262 defines specifications of subsystems safety requirements as additional input, see Figure 66. The HSI is an interface which has unstable parts at the beginning of a project. Especially diagnosis management is usually established during the last development iterations. The majority of this content is in the area of software development but it is also a topic which is relevant for the customer as he gets signals from this interface and might deliver inputs for diagnosis. Therefore it is necessary to share the HSI with the customer which is in conflict to know-how protection of the system design.

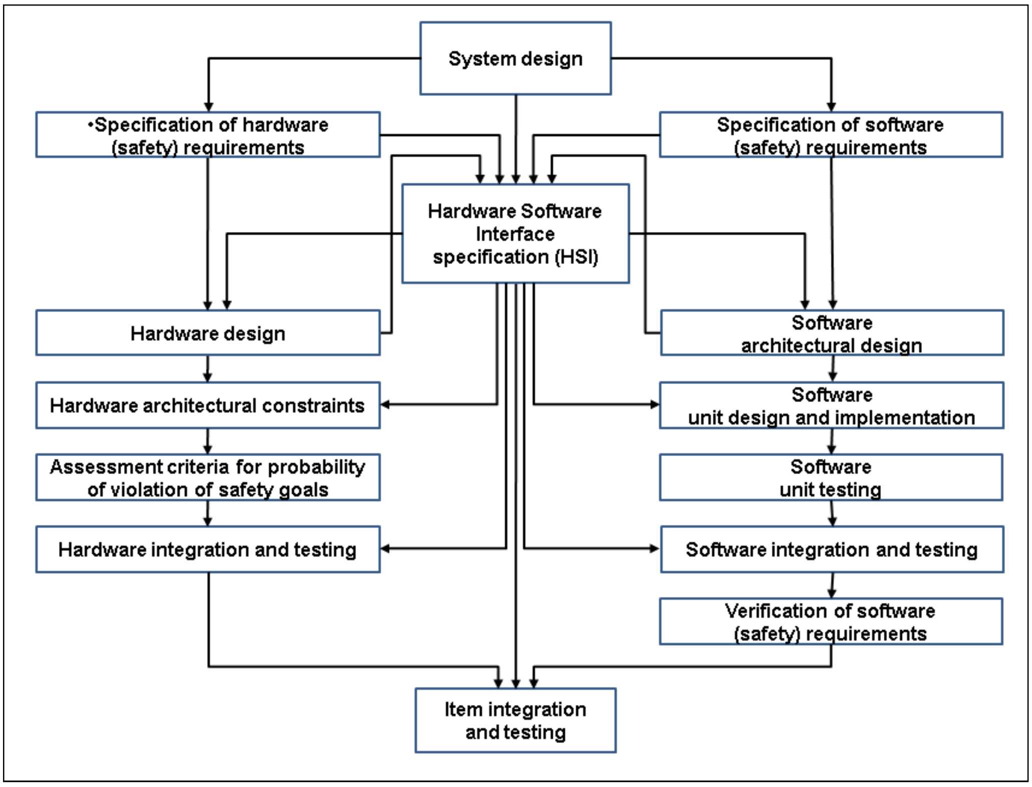

Figure 66: Overview on interaction with the hardware software interface[123]

A suitable solution could be an extra document or special marked requirements which replace this specific interface description of the system design. If the content of the HSI is separated it is possible to share it with the customer, enable links to customer requirements, install additional iterations and define an other responsibility than for the rest of the system design.

4.4 Functional Re-use

In competitive industries with similar products it is important to increase efficiency and reduce cost. One way to achieve these targets is the re-use of parts. This strategy is wide spread but no longer sufficient in automotive development. A more sophisticated way with higher benefit is functional re-

[123] Cf. ISO/DIS 26262-4, p.33

use. In this case the re-use is not limited to parts but includes also re-use of specification and testing. Although many times it will be necessary to make product specific adaptations – even to the parts - the effort can be reduced significant. In order to apply functional re-use it is necessary to consider this also in the establishment of requirements management.
For safety critical projects it starts with the hazard analysis and risk assessment. As functional capabilities of vehicles of different OEMs are similar within the same vehicle class it is possible to make the hazard analysis generic and create a standard risk assessment with generally valid ASIL levels for one product. Due to the fact that the system design is depending heavily on the safety classification a stable risk assessment is a prerequisite for the re-use of system functionality. A persistent system design with unchanged interfaces is also basis for the ability of easy adaptation of single components. This strategy is also applied in telecommunication industry where the product lifecycle of smartphones has dropped to a few month.
The next important point is to separate functional description of requirements from project specific acceptance criteria. If this can be coupled with a system architectural design which lasts for several projects one can create a generic system requirement specification with minimal effort. The requirement descriptions and links to the system design and related subsystem requirements can be re-used, acceptance criteria and other project specific attributes have to be added. More often than not the test specification will only need an adjustment of the acceptance criteria, too.
The chances for re-use are even higher on software level. E.g. the pulse width modulation of an electric motor could be realized the same way independent of the application. No matter if the motor is driving a set of levers or a pump for a hydraulic system the control can be re-used. If the software is created in a way that the project specific adjustment can be made by exchange of some parameters and the functional flow through the software modules stays the same the software can be called functional re-useable. Again a basic condition for that is stability of design and interfaces. In contrast to system level can software requirements be described oriented towards a specific solution. If software is re-used the reduction of effort is usually very high, not only in software creation. Testing of software is about as time consuming as development. By using already existing software modules the need of software module testing is eliminated and for software

integration testing the change of parameters is sufficient. Figure 67 shows this concept schematic.

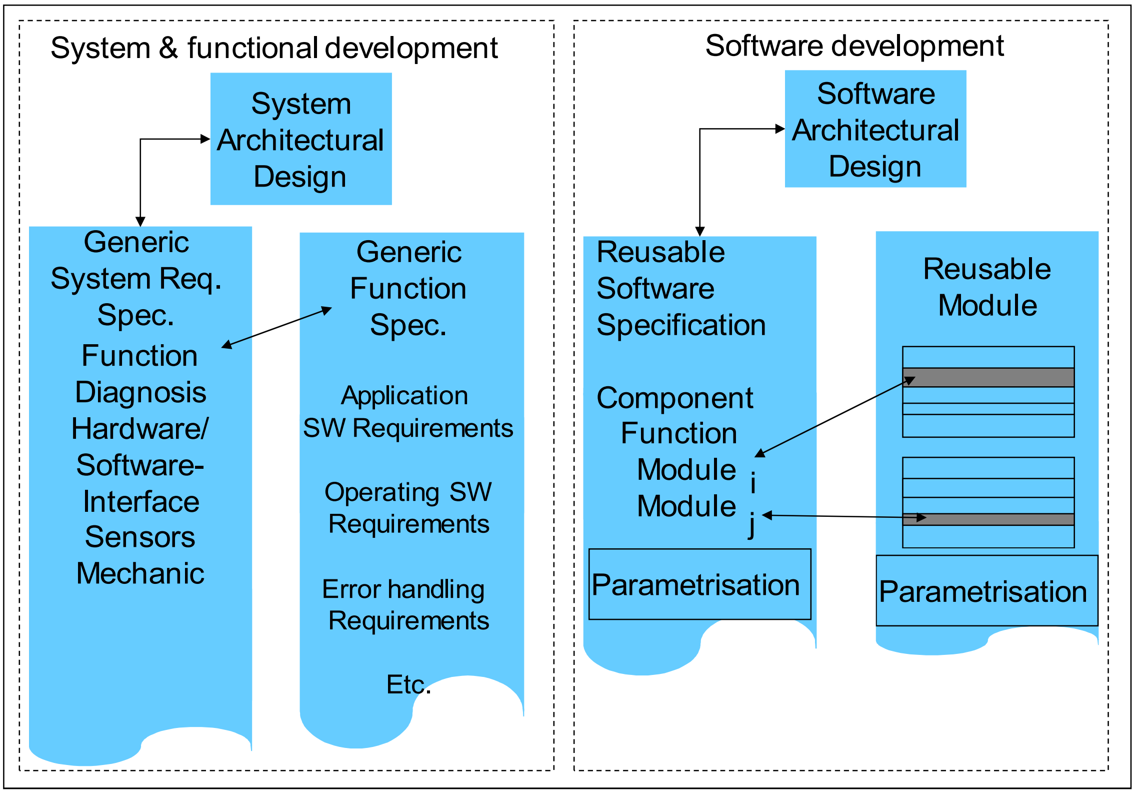

Figure 67: Schematic concept for functional re-use of requirements[124]

4.5 Relationship between requirements management system and other processes

Out of the 15 processes of the HIS scope 9 are directly dedicated to requirements engineering and testing. The 6 remaining processes have also a close relationship to requirements. Because of this interaction it is important to establish a requirements management system that is well interlocked with the whole organization and a variety of tools is avoided. Every additional tool goes along with license fees, establishment and maintenance of interfaces, administration effort, user training and support, etc. But the biggest advantage of an integrated system is probably that information is kept within one system. The establishment of an integrated engineering database reduces both misunderstandings and redundancies. At

[124] Spork (2008), p.14

the same time it increases efficiency and supports the attainment of higher capability levels according Automotive SPICE.

4.6 Summary

For the establishment of an effective requirements management system it is insufficient to concentrate on the requirements of standards like Automotive SPICE or ISO/DIS 26262. Engineering processes have to be aligned with project milestones and the structure has to be derived from internal targets. It is necessary to think about a strategy for protection of intellectual property and functional re-use as well. For many reasons the architectural design is one of the crucial factors for a successful and lean implementation of requirements management. If possible it should be created as one model with different abstraction layers for different requirement levels. A structure with three levels of requirements and full integration of safety critical aspects should be preferred. An overview of a possible requirements management structure including links and major safety related documents is shown in Figure 68.

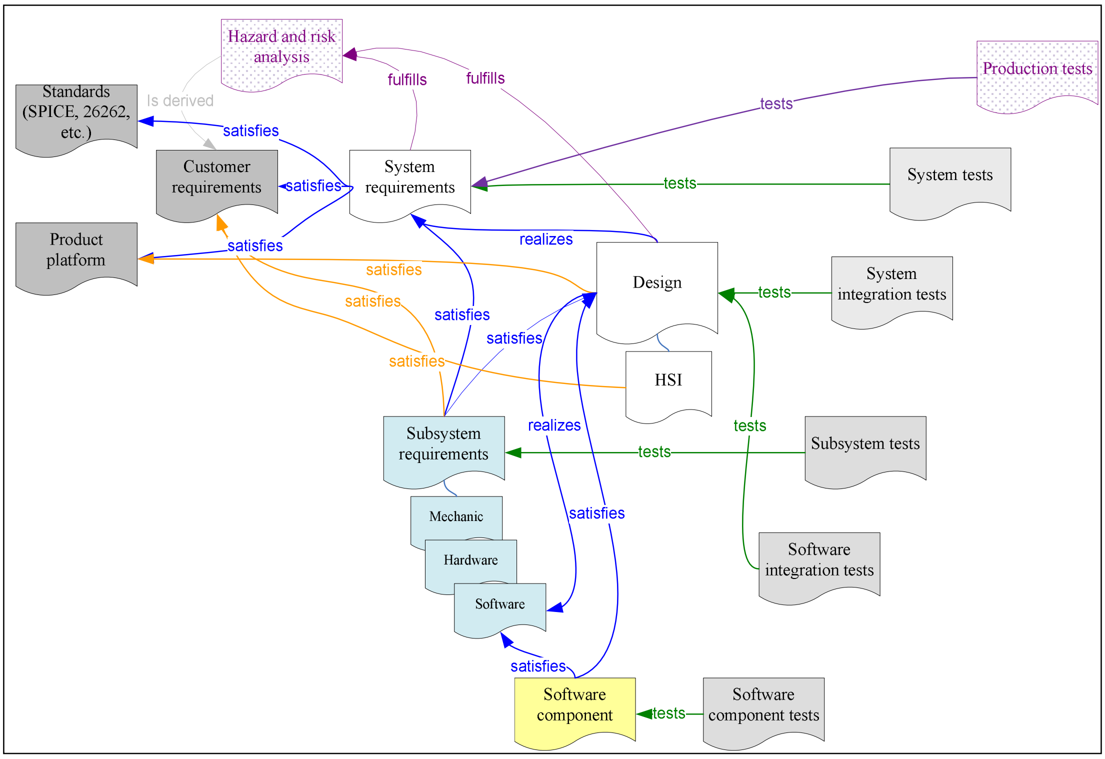

Figure 68: Requirements management structure

Requirements engineering has a close relationship to other processes. For en efficient solution it is essential to limit tool variety and the number of interfaces. This helps to save license fees in addition.

5 Conclusion and Outlook

Requirements management is a key success factor for any company developing complex mechatronic systems. It is well established in software engineering since decades and is currently in the focus of automotive industry. The spread in a flash in Europe was caused by an integration of SPICE in contracts and a strict enforcement by the German OEMs. The number of registered Automotive SPICE assessors grew from approximately 170 in the second quarter of 2008 to 442 in the third quarter 2010[125]. Although the majority of assessors is still based in Germany there is a trend towards Asia. Most of the Japanese assessors were trained during the second half of 2010. The regional distribution of VDA registered Automotive SPICE assessors is shown in Figure 69.

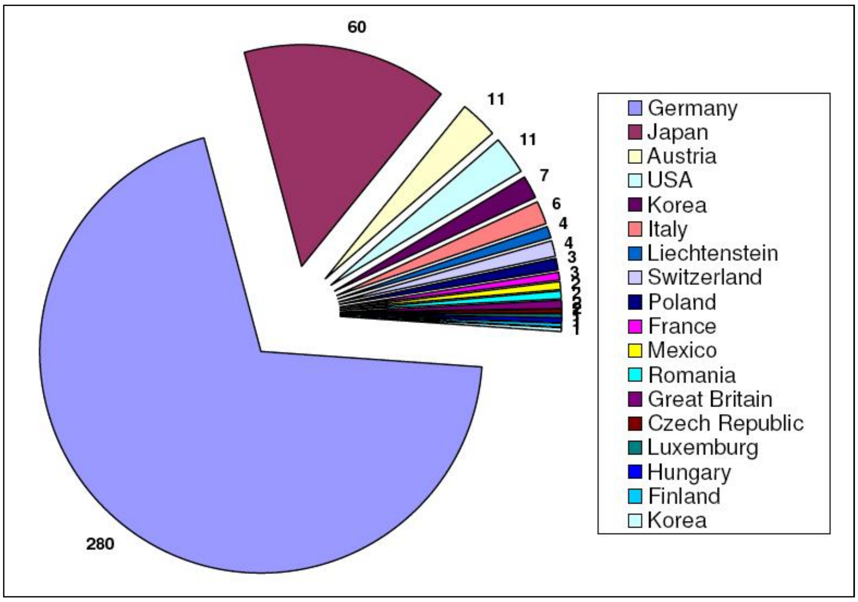

Figure 69: VDA registered Automotive SPICE assessors per country[126]

One can expect that other industries will follow, especially as the concept of ISO/IEC 15504 is open for further domain specific process assessment models and some are in preparation, e.g. Spice for Space and Finance Spice.

More and more mechatronic functions will become safety critical. Therefore many projects will have the obligation to follow SPICE and an appropriate

[125] Cf. Intacs (2011), p.12

[126] Intacs (2011), p.13

safety standard. At the same time ISO/IEC 15504 part 10, the safety extension to SPICE, is in the state final ballot and short before publication. As already practiced in some automotive projects SPICE assessments will be combined with safety audits if safety is required. The rating of one will influence the rating of the other. The big advantage of this combination is the reduction of effort for the OEM and the supplier.

SPICE itself is also developed further. ISO/IEC 15504 will be replaced by a series of standards. For this purpose the number range 33001-33099 has been reserved by the ISO and was split into logical blocks, see Figure 70. The first block contains the normative key elements. All other blocks are grouped by their content and contain informative elements. This restructuring of the standard allows even industry-specific process reference and assessment models to be established as ISO standards.

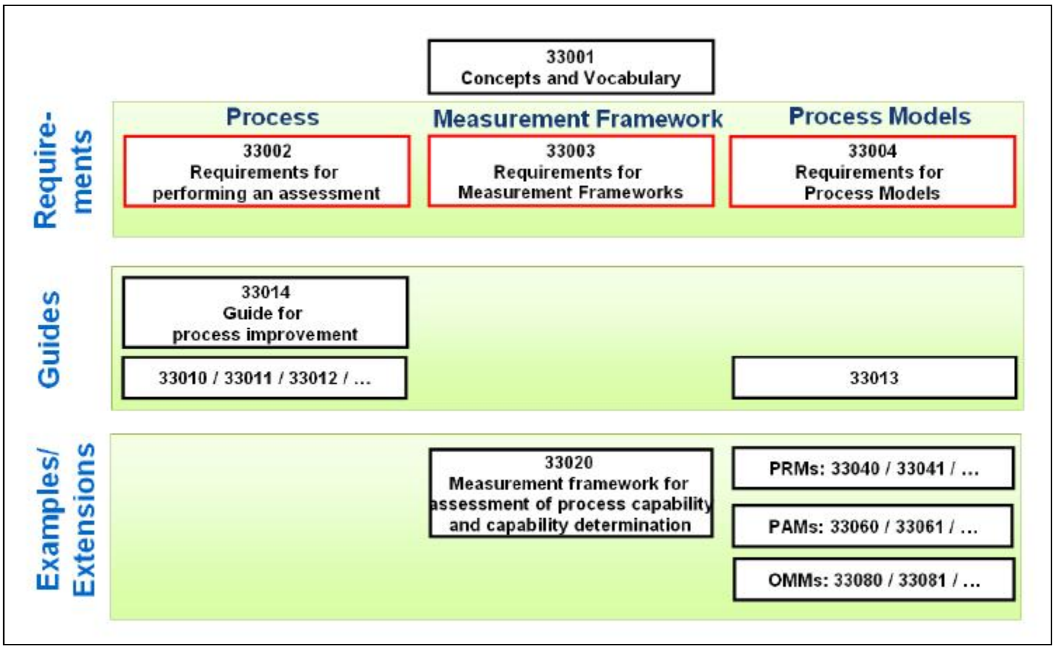

Figure 70: Structure of the ISO/IEC 330XX series[127]

Although one can assume that SPICE will gain additional importance due to the restructuring the spread outside automotive industry will be slower. The development of specific standards takes time and without a powerful lobby the dissemination might need years.

The market environment for system developing companies is challenging, see Figure 71. As markets tend to be less stable than in the past it is very important to shorten time to market. At the same time it is expected that

[127] Dussa-Zieger (2010), p. 27

development can react fast to changing requirements, even if products get more and more complex. To be able to handle this situation efficient it is inevitable to establish appropriate processes and supporting tools. Requirements management is one of the key functions effecting the whole product development. Standards deliver guidelines what should be kept in mind but the usually do not describe how requirements management should be done. Still they can serve as a checklist if the development process is state of the art or if improvements are necessary.

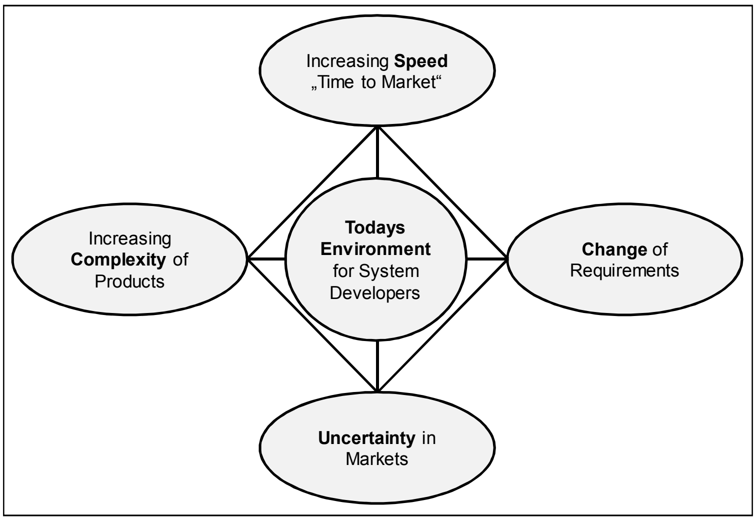

Figure 71: Market Environment for system developing companies[128]

In automotive industry the compliance with Automotive SPICE and ISO/DIS 26262 became a prerequisite to award contracts for the development of complex mechatronic systems from the major European OEMs recently. Assessments are made by OEMs and poor results can lead to new business hold.

Some of the leading German automotive suppliers are running pure mechanic pilot projects according Automotive SPICE. These companies recognized the benefits of a tailored implementation and can avoid differences between mechanic and mechatronic projects in product development and ambiguity for employees working for both. Even first voluntary customer assessments of mechanic projects took place demonstrating competence and thus creating trust. This could be also an indication that aspects of Automotive SPICE will be included in ISO/TS 16949 in the long run.

[128] Stelzmann et al. (2010b), p. 1

To stay in business it is no longer enough to fulfil the requirements of Automotive SPICE and ISO/DIS 26262. Today it is necessary to establish internal processes which facilitate efficient product development. One key element to reach this goal is adequate requirements management. To gain competitive advantage it is even necessary to create a requirements management system which is capable of supporting strategic aspects like protection of intellectual property and re-use of product features.

To support these targets all participating engineering sites will get this thesis to identify improvement potentials. By comparing the existing requirements management system with the other case studies and the requirements management business model each site should be able to implement some improvement measures.

For MAGNA Powertrain it is planned to have a workshop with representatives of the three engineering sites involved in the case studies as next step. The output of this workshop will be a recommendation to management what aspects of the requirements management business model should be integrated in the existing solution. Furthermore it shall be analysed if a harmonization of methods, templates and even tools is seen as technical and economical beneficial and how interfaces for distributed development projects should be defined. The results of this workshop will be presented to top management to reach decision for further proceeding.

In addition the requirements business model will be presented to MAGNA International together with the suggestion to implement an expert network regarding requirements management with periodic workshops for experience exchange and elaboration of synergies.

List of Abbreviations

ALARP: As Low As Reasonable Possible
ASIL: Automotive Safety Integrity Level
BP: Base Practice
CL: Capability Level
DIS: Draft International Standard
DOORS: Requirements management system by IBM
DOXYGEN: tool to extract text and generate documentation
ECU: Electronic Control Unit)
EU: European Union
HIL: Hardware In the Loop
HIS: Hersteller Initiative Software
HSI: Hardware Software Interface
HTML: Hyper Text Markup Language
ID: Unique Identification number
IEC: International Electrotechnical Commission
ISO: International Organization for Standardization
MKS: Requirements management system by MKS
OEM: Original Equipment Manufacturer
OLE: Object Linking and Embedding
PAM: Process Assessment Model
PRM: Process Reference Model
QM: Quality Management
RIF: Requirements Interchange Format
SAP Easy DMS: Easy Document Management System, configuration management system by SAP
SE-Team: Simultaneous Engineering Team
SPICE: Software Process Capability dEtermination
TR: Technische Richtlinie
TS: Technical Specification
UML: Unified Modeling Language
US: United States of America
VDA: Verband der Deutschen Automobilindustrie

List of References

Automotive SPICE: Prozessassessmentmodell, 2008

BERGMANN, B.; KLEFSJÖ,B: Quality from Customer Needs to Customer Satisfaction, Lund 1994, cited in: LEITNER, W.; VALASTIAK, V.: Quality Management Lecture Notes, 2. Edition, Graz 2009

Conway M. E.: How do Committees Invent?, in: Datamation 4/1968

DANZER, H. H.: Qualitätsmanagement im Verdrängungswettbewerb, Wuppertal 1995

DUSSA-ZIEGER, K.: SPICE: The Next Generation, International SPICE Days, Stuttgart 2010

EBERT, C.; LEDERER D.: Funktionale Sicherheit - Das Gesamtsystem Fahrzeug in: Automotive Engineering Tools 8/2008

HÖHN, H. et al.: Software Engineering nach Automotive SPICE, Heidelberg 2009

HÖRMANN, K. et.al.: SPICE in der Praxis, Heidelberg 2006

HRUSCHKA, P.; RUPP, C.: Agile Softwareentwicklung für Embedded Real-Time Systeme mit UML, München/Wien 2002

INTACS: Intacs newsletter 2011-01 en

ISO 9000:2005: Quality management systems – Fundamentals and vocabulary

ISO 9001:2008: Quality management systems – Requirements

ISO/DIS 26262: Road vehicles — Functional safety

ISO/DIS 26262-1: Vocabulary

ISO/DIS 26262-2: Management of functional safety

ISO/DIS 26262-3: Concept phase

ISO/DIS 26262-4: Product development: system level

ISO/DIS 26262-6: Product development: software level

ISO/DIS 26262-9: ASIL-oriented and safety-oriented analyses

ISO/DIS 26262-10: Guideline

ISO/IEC 15504-2:2003: Software engineering — Process assessment — Part 2: Performing an assessment

ISO/IEC 15504-4:2004: Information technology — Process assessment — Part 4: Guidance on use for process improvement and process capability determination

ISO/IEC 15504-5:2006: Information technology — Process assessment — Part 6: An exemplar system life cycle process assessment model

ISO/TS 16949:2009: Quality management systems — Particular requirements for the application of ISO 9001:2008 for automotive production and relevant service part organizations

JURAN, J. M.: Juran’s Quality Handbook, Fifth Edition, Singapore 2000

KIEDAISCH, F. et al.: Experiences On Outsourcing Requirements Specifications, in: MESSNARZ R. (editor): 8th EuroSPI Conference Industrial Proceedings, Limerick 2001

LEITNER, W.; VALASTIAK, V.: Quality Management Lecture Notes, 2. Edition, Graz 2009

LÖW, P.; PABST, R.; PETRY, E.: Funktionale Sicherheit in der Praxis, Heidelberg 2010

MAGNA Powertrain, MAGNA Electronics Requirements Engineering Policy

MASING, W.: Handbuch Qualitätsmanagement, München/Wien 1999

METHODPARK: VDA certified provisional assessor training 2009

METHODPARK: VDA certified provisional assessor training 2010

MESSNARZ, R.: Software in der Automobilbranche, in HINDEL, B.(Editor): Software im Automobil, Düsseldorf 2007

MÜLLER, M. et al.: Automotive SPICE in der Praxis, Heidelberg 2007

PABST, R.; SEEGERS, B.: Safety in Automotive SPICE Assessments, in: Gate 4 SPICE, Kornwestheim 2010

POHL, K.; RUPP, C.: Basiswissen Requirements Engineering, Heidelberg 2009

POTH, A.: SPI of the Requirements-Engineering-Process for Embedded Systems Using SPICE, in: MESSNARZ R. (editor): 13th EuroSPI Conference Industrial Proceedings, Joensuu 2006

SPORK, G.; PICHLER U.: Establishment of a Performance Driven Improvement Program, in: MESSNARZ, R. (editor) 14th EuroSPI Conference Industrial Proceedings, also published in Wiley Interscience Journal, SPIP Proceeding in June 2008, Potsdam 2007

SPORK, G.: Strategies for Platform Development, Key Note 15th EuroSPI Conference, Dublin 2008

STEFFAN, H.: Vehicle Safety 1, Graz 2009

STELZMANN E. et al.: Agility meets Systems Engineering: a Catalog of Success Factors from Industry Practice, in: MESSNARZ R. (editor): 17th EuroSPI Conference Industrial Proceedings, Grenoble 2010

STELZMANN E. et al.: Practical Use of Agile Methods within Systems Engineering, International SPICE Days, Stuttgart 2010

THORSEN, H. K. et al.: Evaluation of using a UML methodology in a large embedded development project, in: MESSNARZ, R. (editor): 11th EuroSPI Conference Industrial Proceedings, Trondheim 2004

WALLMÜLLER E.: Software Process Improvement mit CMMI, PSP/TSP und ISO 15504, München/Wien 2007

WLOKKA, A.: CMMI / Automotive SPICE Mapping, International SPICE Days, Stuttgart 2009

Weblinks

MAGNA International: http://www.MAGNA.com/MAGNA/en/about/ 31.01.2011

MAGNA International: MAGNA Overview A4, Version 5, Aurora 2010: http://intranet.magnapowertrain.com/info/Corporate%20Presentations/Forms/documents1.aspx, 20.02.2011

Qualitäts Management Center im Verband der Automobilindustrie: http://www.vda-qmc.de/zertifizierung/iatf, 26.02.2011

Hersteller Initiative Software: http://portal.automotive-his.de/index.php?option=com_content&task=view&id=7&Itemid=18, 26.02.2011

Automotive Special Interest Group: http://automotivespice.com/web/Introduction.html, 17.01.2011

The SPICE User Group: http://www.spiceusergroup.org/forum2/topics/isoiec-1550410-safety, 17.01.2011

U.S. Department of Transportation, Federal Highway Administration: http://ops.fhwa.dot.gov/publications/tptms/primer/index.htm, 22.01.2011

List of figures

Figure 1: Area of Best Practices 2
Figure 2: Quality of engineering sites 2
Figure 3: MAGNA's global capabilities 3
Figure 4: MAGNA's operational structure 4
Figure 5: The Juran Trilogy diagram 6
Figure 6: Overview about main automotive standards 8
Figure 7: Model of a process-based quality management system 9
Figure 8: Percentage of manufacturing cost for passenger vehicles 10
Figure 9: Increase of functionality in passenger vehicles 11
Figure 10: Innovative safety related mechatronic functionality 12
Figure 11: Stability of requirements during project start 12
Figure 12: Paradigm shift in development 14
Figure 13: History of standards 15
Figure 14: Members of the Automotive Special Interest Group 16
Figure 15: Structure of ISO/IEC 15504 16
Figure 16: Normative elements of ISO/IEC 15504 17
Figure 17: Assessment indicators 18
Figure 18: Process Assessment Model relationships 19
Figure 19: Capability levels 20
Figure 20: Example assessment output set of process profiles 22
Figure 21: SPICE processes 24
Figure 22: Structure to establish a Process Assessment Model 25
Figure 23: Key concept schematic 26
Figure 24: Bi-lateral traceability 28
Figure 25: Process areas and capability levels of CMM 29
Figure 26: CMMI / Automotive SPICE-mapping 30
Figure 27: Overview of ISO 26262 32
Figure 28: Concept for risk minimization 33
Figure 29: Safety lifecycle 35
Figure 30: Software safety requirements process 36
Figure 31: Classification scheme of ASILs when decomposing safety requirements 37
Figure 32:Typical recursive specification of a system on different abstraction levels 38
Figure 33: V-Model 40

Figure 34: System decomposition 41
Figure 35: V-Model of Continental AG 42
Figure 36: V-Model of ABB 43
Figure 37: Engineering Process Landscape 47
Figure 38: Structure of requirements management 49
Figure 39: Requirements frame 51
Figure 40: Traceability between MSWord and MKS 52
Figure 41: Requirements workflow 53
Figure 42: Traceability between requirements and tests 54
Figure 43: Overview of activities and related tools 56
Figure 44: Project lifecycle model with freeze points 59
Figure 45: Process model 60
Figure 46: System and system context 61
Figure 47: Requirements structure and possible traces 62
Figure 48: Planed requirements structure 64
Figure 49: Level 2 process "Create Products and Processes" 67
Figure 50: V-model 68
Figure 51: Turtle diagram of test management 69
Figure 52: Structure of requirements management 70
Figure 53: Excerpt of UML model 72
Figure 54: Structural overview about linked elements in software development 74
Figure 55: Traceability and tools 75
Figure 56: Requirements analysis workflow 79
Figure 57: Project lifecycle model 83
Figure 58: Discipline interface chart 84
Figure 59: System requirements baselines 85
Figure 60: Lifecycle of requirements 86
Figure 61: Integration of engineering activities and requirements lifecycle 86
Figure 62: Link structure on system level 87
Figure 63: Requirements Levels 92
Figure 64: Project lifecycle model 93
Figure 65: Simplified V-model 94
Figure 66: Overview on interaction with the hardware software interface 98
Figure 67: Schematic concept for functional re-use of requirements 100
Figure 68: Requirements management structure 101
Figure 69: VDA registered Automotive SPICE assessors per country 103

Figure 70: Structure of the ISO/IEC 330XX series.....................................104
Figure 71: Market Environment for system developing companies 105

List of Tables

Table 1: Related processes for Process Attributes..23
Table 2: ASIL determination 34