Dipl.-Ing. Markus Ernst, MLBT

KPI-related analysis methods to optimise mechatronic product development processes

Doctoral thesis

to achieve the university degree of
Doktor der technischen Wissenschaften
(Dr.techn./PhD)
submitted to
Graz University of Technology

1st evaluator: Assoc.Prof. Dipl.-Ing. Dr.techn. Mario Hirz
Institute of Automotive Engineering, Graz University of Technology
2nd evaluator: Prof. Dr.-Ing. Su Zhou
School of Automotive Studies, Tongji University Shanghai

Graz, November 2016

Affidavit

I declare that I have authored this thesis independently, that I have not used other than the declared sources/resources, and that I have explicitly indicated all material which has been quoted either literally or by content from the sources used. The text document uploaded to TUGRAZonline is identical to the present doctoral thesis.

.... Date

.... Signature

Acknowledgment

First and foremost, I would like to thank my advisor Assoc. Prof. Dipl.-Ing. Dr.techn. Mario Hirz for his scientific guidance. Our interesting discussions, common publications, and his excellent knowledge helped me a lot to finish my thesis. Special thanks to Prof. Dr.-Ing. Su Zhou for giving me the opportunity to research abroad at the Tongji University Shanghai in China and his interest in being the second evaluator of my doctoral thesis.

Thanks to Assoc. Prof. Dipl.-Ing. Dr.techn. Sebastian Fischmeister for enabling a research stay abroad at the University of Waterloo in Canada and the great discussions with all the colleagues and scientific support.

I would like to thank all my workmates at the Institute of Automotive Engineering at Graz University of Technology under the lead of Prof. Dipl.-Ing. Dr.techn. Peter Fischer for the great working atmosphere.

Special thanks go to Dipl.-Ing. Dr.techn. Jürgen Fabian for the supporting work during my whole PhD thesis. The successful collaboration in master and bachelor theses, publications, and project management are only an excerpt of his work.

I would like to thank all master and bachelor students for the great common scientific evaluations and assistance during my doctoral thesis.

I would like to express my appreciation to MAGNA Powertrain AG & Co KG, especially Dipl.-Ing. Andreas Münzer and his team as well as all colleagues from the locations Lannach and Ilz, for the opportunity, support, and trust regarding my doctoral thesis.

I sincerely want to thank my parents Margit and Bmst. Ing. Markus Ernst for giving me the opportunity to finish three studies and for their encouragement to reach all my goals. I would like to thank my sister OA Dr. med. univ. Barbara Ernst for the permanent and special support during all my studies. Special thanks to my girlfriend Mag. iur. Katharina Suk, BSc, MSc for her encouragement and help in all different situation during my studies. I owe my family a lot and I am very thankful for all their engagement.

I dedicate my doctoral thesis to my parents.

Kurzfassung

Die Industrie ist mit steigender Komplexität von Systemen und zunehmendem Qualitätsbewusstsein der Kundeninnen und Kunden konfrontiert. Derzeitige automotive Entwicklungstrends erfordern einen erhöhten Einsatz mechatronischer Systeme, z. B. für die Vernetzung von alternativen Antriebssystemen und erhöhte Sicherheitsmaßnahmen. Die unterschiedlichen Komponenten mechatronischer Systeme (z. B. Elektronik/Elektrik, Software und Mechanik) und deren Vernetzung stellen eine große Herausforderung für Entwicklungsprozesse und Feldeinsätze dar. Diese Umstände beinhalten ein erhöhtes Risiko für Ausfälle der Systeme im Einsatz bei Kundeninnen und Kunden, welche zu Rückrufen mit hohen Garantiekosten führen können. Aufgrund dieser Einflüsse müssen industrielle Herstellerinnen und Hersteller sowie deren Zuliefererinnen und Zulieferer eine kontinuierliche Reduktion von Fehlern in ihren Produkten sicherstellen. Die Analyse von komplexen Systemen, wie z. B. mechatronische Produkte, erfordern neue, innovative Analysemethoden, um den gesamten Produktlebenszyklus effizient untersuchen zu können. Unternehmen sammeln umfassende Informationen ihrer Systeme im Zuge der Entwicklungsprozesse und des Einsatzes bei Kundeninnen und Kunden, jedoch wird das Potential dieser Daten oftmals nicht vollständig genutzt. Im Rahmen dieser Doktorarbeit wird eine Methode präsentiert, welche den Einsatz sogenannter Key Process/Performance Indicators (KPIs) mittels neu entwickelter Analysemethoden auf Basis von Entwicklungs-, Test- und Felddaten ermöglicht. Das Ziel ist es, optimierte, zuverlässige und prädiktive Entwicklungsprozesse sowie die Kontrolle über den gesamten Produktlebenszyklus zu ermöglichen. Die präsentierten, generischen Methoden erweitern und optimieren das Qualitätsmanagement, Prozesskontrollen sowie Risikoanalysen und unterstützen Entscheidungsfindungen im Freigabeprozess mechatronischer Systeme. Der Einsatz stochastische Methoden zur Zuverlässigkeitsschätzung ermöglicht die Verbesserung der Qualität während der Entwicklung und im Feldeinsatz. Des Weiteren unterstützen die entwickelten Methoden die Einführung eines selbstlernenden Prozesses als Frühwarnsystem während der Entwicklung und im Feldeinsatz. Die generische Methode der Doktorarbeit wird in Serienprojekten eingesetzt. Durch die Verbindung der entwickelten Analysemethoden zum Serienbetrieb ist eine direkte Verwendung in der Automobilindustrie und in weiteren industriellen Einsatzgebieten möglich.

Abstract

The industry is faced with both an increasing complexity of systems and rising quality awareness of customers. Current automotive development trends require more application of mechatronic systems, e.g. for connectivity of alternative propulsion systems, and enhanced safety measures. The different components of mechatronic systems (such as electronics/electrics, software and mechanics) and their interaction represent a huge challenge for development processes and field uses. These circumstances bear a higher risk of possible malfunctions during field uses and can lead to product recalls with high warranty costs. Due to these influences, industrial manufacturers and their suppliers have to ensure a continuous reduction of faults of their products. The analysis of complex systems, such as mechatronic applications, requires new and innovative analysis methods to effectively investigate the whole product life cycle. During development and field use, companies collect data with comprehensive information. The potential of these data is often overseen.

In this PhD thesis, a methodology introduces so-called key process/performance indicators (KPIs) by developing new analysis methods based on development, test, and field data. The goal is to enable optimised, reliable, and predictive development processes and control during the whole product life cycle. The presented generic methods improve and optimise quality management, process control, risk analysis, and decision-making during the release process of mechatronic systems. In addition, stochastic analysis methods are used to estimate the system reliability for facilitating the improvement of quality in development and field. Furthermore, the methods enable a self-learning process to introduce an early warning system during development and field use. The PhD thesis delivers a generic approach which is used in series development projects. By connecting the developed analysis methods with series development, direct use in automotive industry as well as other industrial areas is possible.

Acronyms and Abbreviations

A/D	Analog digital converter
CAN	Controller area network
CC	Component controller
CCB	Change control board
CDD	Complex device driver
CFR	Constant failure rate
CI	Change issue
CIP	Continuous improvement process
CSF	Critical success factors
DC	Diagnostic coverage
DeCoDe	Demand compliant design
DFR	Decreasing failure rate
DoE	Design of experiments
DTC	Diagnostic trouble code
E/E	Electronics/electrics
ECU	Electric control unit
EDM	Engineering data management systems
EEPROM	Electrically erasable programmable read-only memory
EOL	End-of-line
EOP	End of production
EPROM	Erasable programmable read-only memory
ERM	Entity relationship modelling
FiL	Function-in-the-loop
FMEA	Failure mode and effects analysis
FTA	Fault tree analysis
GLM	Generalised linear model
GM	General management
HiL	Hardware-in-the-loop
HW	Hardware
ID	Identity
IFR	Increasing failure rate
IT	Information technology
KPI	Key performance/process indicator
KRI	Key result indicators
LM	Linear model
MiL	Module-in-the-loop

MLM	Maximum-likelihood method
MR	Module requirement
MS	Milestone
MTBF	Mean time between failure
MTTF	Mean time to failure
MTTFF	Mean time to first failure
MTTR	Mean time to repair
NHPP	Non homogenous poisson process
OEM	Original equipment manufacturer
OLS	Ordinary least square
PCP	Product creation process
PDP	Product development process
PI	Performance/process indicators
PLC	Product life cycle
PLM	Product life cycle management
PM	Project management
PrM	Process management
QA	Quality assurance
QFD	Quality function deployment
QIM	Quality information management system
QMS	Quality management system
RAM	Random access memory
RPM	Revolutions per minute
RPN	Risk priority number
RTE	Runtime environment
SCR	System component requirement
SiL, SIL	Software-in-the-loop, safety integrity level
SOP	Start of production
SPC	Statistical process control
SQMA	Situation-based qualitative modelling and analysis
SW	Software
TC	Test case
TQM	Total quality management
VC	Vehicle controller
VDS	Vehicle description section
VIN	Vehicle identification number
VIS	Vehicle indicator section
WMI	World manufacturer identification
XT	Extreme tailoring

Table of Signs and Symbols

Symbol	Description
$D_{(t)}$	Amount of failure points
$P_{(B_i\|A)}$	A-posteriori probability
$P_{(B_i)}$	A-priori probability
θ	Confidence estimator
$H_{(x)}$	Cumulated hazard rate
$Y_{(x)}, Y_{\mathrm{i}}, \widehat{Y_{\imath}}$	Dependent variable
$\varepsilon, \varepsilon_{\mathrm{i}}, \widehat{\varepsilon_{\imath}}$	Error rate
E_i	Expected value
$F_{(t)}$	Failure function, probability distribution function
$\lambda_{(t)}$	Failure rate
a	Form parameter
b	Function parameter
$h_{(x)}, Z_{(t)}$	Hazard rate
x_i	Independent variable
$\lambda_{i(t)}$	Instantaneous failure rate
$L(\vartheta)$	Likelihood function
P_ϑ	Maximum-likelihood function
$\Lambda_{(t)}$	Mean function of point process
μ	Mean value
$\hat{\beta}$	Non homogeneity parameter
$N_{(t)}$	Number of failures
O	Occurrence parameter
$\beta, \hat{\beta}, \beta_0, \vartheta$	Parameter
P	Probability
X	Random variable
$\mu_{(t)}$	Rate of occurrence of failures
$R_{(t)}$	Reliability function
RPN	Risk priority number
$\hat{\theta}$	Scale parameter
S	Severity parameter
σ	Standard deviation
$X_{(t)}$	State variable
$\overline{F_{(t)}}$	Survival function
t, T, T_i	Time

Contents

1 Introduction

Nowadays, industry is faced with a steadily increasing complexity of products and customers raising demands for quality. Furthermore, current industrial development trends require more application of mechatronic systems. In automotive industry autonomous driving, connectivity, alternative propulsion systems, and enhanced safety measures represent main drivers for the application of mechatronics. Developing mechatronic systems is a huge challenge in practice due to the different involved components such as electronics/electrics (E/E), software and mechanics as well as their interaction. In case of automotive applications, the increasing complexity, the number of electronic control units (ECUs), and the variety of desired, implemented software, E/E, or mechanical functions require additional effort in development and field uses, e.g. comprehensive processes or effortful warranty support. Hence, the development process of these systems and management of the systems' product life cycle (PLC) bear a higher risk of possible malfunctions. Functional deviations are related to mechanical, electrical, and computational subsystems as well as the interaction of these components. In cases of faults occurring in field use, warranty leads to product recalls which are typically associated with high costs. Due to these influences, industrial manufacturers and their suppliers have to ensure a continuous reduction of faults in their products. However, to avoid failure occurrence mainly in field use at the customer faults of mechatronic systems have to be detected and prevented efficiently. This can be enabled by improving quality management and development processes. For this purpose, innovative analysis methods have to be developed to control process and reliability during the whole product life cycle ensuring an effective quality management. An objective quality and process control can be enabled by using key process/performance indicators (KPIs). These metrics are the result of comprehensive analysis or evaluation methods and have to be introduced as soon as possible during the product development process. The goal is to support quality evaluation and risk analysis for the release process of mechatronic systems by using stochastic analysis methods to gain an estimation of system reliability, [12], [45], [49], [50], [52], [149], [156].

1.1 Motivation

"If anything can go wrong, it will."

Referring to Murphy's law, [63], [173], the development of complex systems always includes insecurity and a residual risk to humans or machines. Increasing competition, shortened development durations, and cost pressure during all phases of the PLC represent risk factors for industrial companies. In case of the automotive industry, recall rates have increased over the past years which lead to additional costs for OEMs and suppliers due to warranty, [52], [103], [187], (see Figure 1-1). The industry has the duty by law to ensure reliability of their products to prevent humans and their environment from damage because of malfunctions, [54], [108], [187].

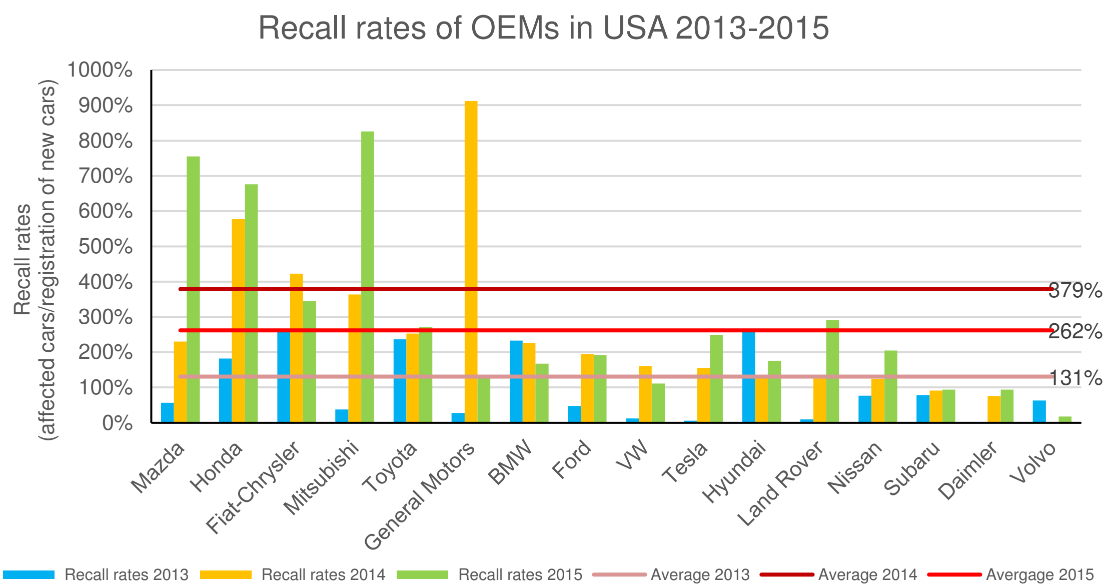

Figure 1-1: Recall rates of OEMs in the USA from 2013 to 2015, c.f. [20], [46], [176]

Product recalls financially impact OEMs and suppliers, because – following the rule of ten – in later phases of the PLC costs of faults are increasing stepwise ten times, as shown in Figure 1-2. To handle these challenges and to fulfil these additional requirements, different types of analysis methods based on objective quality evaluations and specific metrics have to be introduced in early phases of the development process. The aim is to support fault prevention as well as to minimise the residual error rate in customer use by optimising workflows and processes. Early fault prevention and detection saves warranty costs in later phases, [50], [52], [187].

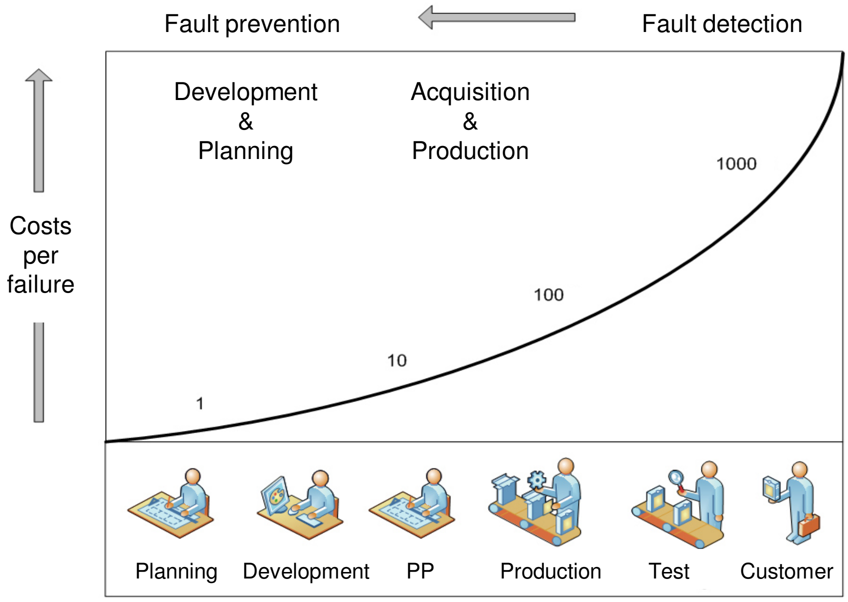

Figure 1-2: Increasing costs following the rule of ten, [44], [46], c.f. [162]

1.2 Initial situation

During development several methods, processes, and quality management tools for different industrial areas are available. Due to the increasing application and need for complex products, these methods often reach their limits or do not satisfy requirements. The analysis of complex systems, such as mechatronic applications, require new and innovative analysis methods to investigate the whole PLC effectively. During development and field use, companies collect data with comprehensive information. The potential of these data is often overseen. An efficient and effective management and quality control to deliver reliable products needs an enhanced control system of all phases of PLC. Furthermore, the implementation of stochastic methods increases analysis possibilities to enable improvement of quality in development and field.

1.3 Structure

The structure of this PhD thesis is divided into seven parts. After an introduction to the topic, general definitions and characteristics of the product life cycle (chapter 1), standards as well as different development processes are discussed (chapter 2). Chapter 3 is concerned with quality management, the definition process of key performance indicators and methods of reliability investigation. Chapter 4 contains the generic methodology of the PhD thesis presenting analysis methods using KPIs in different phases of the product life cycle as well as the stochastic reliability investigations to compare failure rates. Chapter 5 shows the automotive application of the methodology by the example of automotive software within all-wheel drive systems. Chapter 6 discusses potentials and results of presented analysis methods followed by a conclusion in chapter 7.

1.4 Project management

In the course of this PhD project, several papers have been published, awards and scholarships have been won, and Master and Bachelor theses have been supervised and conducted:

Papers:

- Ernst, M.; Hirz, M.; Fabian, J.: The Potential of Key Process/Performance Indicators (KPIs) in Automotive Software Quality Management, 2016-01-0046, SAE World Congress, SAE, Detroit, Michigan, U.S., 2016, [52]
- Ernst, M.; Hirz, M.; Fabian, J.; Fehrer, F.; Wolf, J.; Steinkellner, M: Key Process/Performance Indicators (KPIs) in der automotiven Softwareentwicklung, research report FTG-2015/10, Institute of Automotive Engineering, Graz University of Technology, Austria, 2015, [45]
- Ernst, M.; Erlachner, S.; Hirz, M.; Fabian, J.; Wotawa, F.: Analysis Methods in the Development Process of Mechatronic Drivetrain Systems with Special Focus on Automotive Software, The 19th World Multi-Conference on Systemics, Cybernetics and Informatics: WMSCI, Orlando, Florida, U.S., 2015, [50]
- Ernst, M.; Erlachner, S.; Hirz, M.; Fabian, J.: Optimisation of Automotive Software Quality Management by Use of Analysis Methods in the Development Process of Mechatronic Systems, International Conference on Advances in Software, Control and Mechanical Engineering, ICSCME, Antalya, Turkey, 2015, [49]
- Ernst, M.: Marktpotentiale eines innovativen Analysetools für erzeugende Branchen am Markt Österreich und Deutschland, Institute für Handel Absatz und Marketing, Johannes Kepler University, Linz, Austria, 2015, [42]
- Ernst, M.; Dallinger, P.; Fabian, J.; Hirz, M.; Schnellbach, A.: Innovative Analysemethode in der Entwicklung mechatronischer Systeme am Beispiel elektrifizierter Antriebsstränge, Elektrotechnik & Informationstechnik 132/3: 134–141. DOI 10.1007/s00502-015-0292-7, Springer, Germany, 2015, [47]
- Dallinger, P.; Ernst, M.; Fabian, J.; Schnellbach, A.: Development of a Data Model for an Innovative Analysis Method for Mechatronic Systems in the Field of Automotive Engineering, Proceedings of 2015 International

Conference on Advances in Software, Control and Mechanical Engineering, Antalya, Turkey, 2015, [26]
- Ernst, M.; Dallinger, P.; Hirz, M.; Fabian, J.; Schnellbach, A.: Potentiale einer innovativen Analysemethodik und deren Datenmodellierung, FMEA KONKRET, 02/2014, ISSN 2198-6851, FMEA plus Akademie, Germany, 2014, [48]
- Ernst, M.; Fabian, J.; Hirz, M.: Innovative Methoden zur Analyse elektrifizierter Antriebsstränge zukünftiger Fahrzeugkonzepte, 13. Symposium Energieinnovation, Graz University of Technology, Austria, 2014, [51]
- Ernst, M.: 1. Zwischenbericht zur Analyse und Prognosemodellierung mechatronischer Fahrzeugkomponenten in der Entwicklung und im Feld, research report Nr. FTG-2014/01, Institute of Automotive Engineering, Graz University of Technology, Austria, 2014, [43]

Awards and scholarships:
- 1. place innovation award for Software development, ti&m and SoftNet, ASQT, 2016
- KUWI scholarship 2016
- FSI scholarship 2016
- Best Paper Award, Universal Researchers in Science and Technology, International Conference on Advances in Software, Control and Mechanical Engineering, 2015
- Julius Raab scholarship 2013-2016

Master theses:
- Erlachner, S.: Einsatz von Entwicklungs- und Testdaten zur Bewertung automotiver Software, Institute of Automotive Engineering, Institute of Software Technology, Graz University of Technology, Austria, 2015, [41]
- Dallinger, P.: Entwicklung eines Datenmodells einer innovativen Analysemethode für mechatronische Systeme im Fahrzeug, Institute of Automotive Engineering, Graz University of Technology, Austria, 2015, [25]
- Fehrer, F.: Entwicklungs- und Testdatenanalyse mechatronischer Komponenten im Automobilbau, Institute of Automotive Engineering, Graz University of Technology, Austria, 2016, [57]
- Passegger, W.: Datenmanagement für Reklamationen und Befundungsinspektionen in der Automobilindustrie, Institute of Automotive Engineering, Graz University of Technology, Austria, 2016, [130]

Bachelor theses:

- Dernoscheg, S.: Konzeptionelle Risikoanalyse für Produktfreigaben in der Automobilindustrie, Institute of Automotive Engineering, Graz University of Technology, Austria, 2014, [30]
- Perner, P.: Bewertung von Zuverlässigkeitsmodellen für automotive Software, Institute of Automotive Engineering, Graz University of Technology, Austria, 2014, [134]
- Pickl, T., G.: Untersuchung und Evaluierung von Analysetools, Institute of Automotive Engineering, Graz University of Technology, Austria, 2015, [137]
- Ebner, C.: Rückkopplung von Erkenntnissen aus dem Einsatzverhalten von Fahrzeugen auf die Entwicklung von Automobilkomponenten, Institute of Automotive Engineering, Graz University of Technology, Austria, 2014, [37]
- Manko, J.: Konzeptionelle Vergleichbarkeitsanalyse von Entwicklungs- und Felddaten, Institute of Automotive Engineering, Graz University of Technology, Austria, 2015, [113]
- Prodinger, M.: Klassifizierung und Anwendung von Lizenzmodellen für Analysetools, Institute of Automotive Engineering, Graz University of Technology, Austria, 2015, [139]
- Wilfinger, M.: Interaktion mechatronischer Antriebsstrangkomponenten, Institute of Automotive Engineering, Graz University of Technology, Austria, 2015, [191]
- Kügler, P, M.: Analyse von Methoden für Entwicklung, Produktion und Einsatz von technischen Produkten, Institute of Automotive Engineering, Graz University of Technology, Austria, 2016, [107]
- Steinkellner, M.: Entwicklungs- und Testdatenanalyse automotiver Software - Mitentwicklung eines Analysetools, Institute of Automotive Engineering, Graz University of Technology, Austria, 2016, [177]
- Wolf, J.: Entwicklung eines Analysetools in VisualBasic, Institute of Automotive Engineering, Graz University of Technology, Austria, 2016, [193]
- Pferscher, A.: Data storage, management and analysis for complaints and teardown inspections, Institute of Automotive Engineering, Graz University of Technology, Austria, 2016, [136]
- Brodtrager, C.: Entwicklungsprozesse und Eingliederung von Analysetools, Institute of Automotive Engineering, Graz University of Technology, Austria, 2016, [16]

1.5 Objective

The aim of this doctoral thesis is to develop methods and strategies for the reduction of failures of mechatronic systems in the customers' field use in order to decrease threats to humans and machines. This should be enabled by optimised, predictable, and reliable development processes as well as an integrated approach to reduce failures in mechatronic systems. The development of innovative analysis methods is intended to help to detect, evaluate, and predict failures in mechatronic systems during all product life phases, such as requirement phase, development phase, and field use. Figure 1-3 shows relevant parts of the PLC to introduce these analysis methods effectively and efficiently. For this purpose, the first relevant part represents the phase of "Requirement specification". The introduction of a comprehensive analysis method has to ensure that mechatronic systems can be analysed and evaluated to detect all possible faults resulting in a preventive measure planning included in the development process. In the next phase "Development", new analysis methods have to consider development and test data to enable an objective and comparable investigation of development processes by introducing KPIs. In the next phase "Field phase at customer use" the development and application of new analysis methods are necessary to investigate the field use behaviour of products to provide feedback in development phases. Finally, stochastic methods have to be developed to allow a comparison of development and field data resulting in a self-learning prognosis system to estimate residual error rates to support release decisions in development.

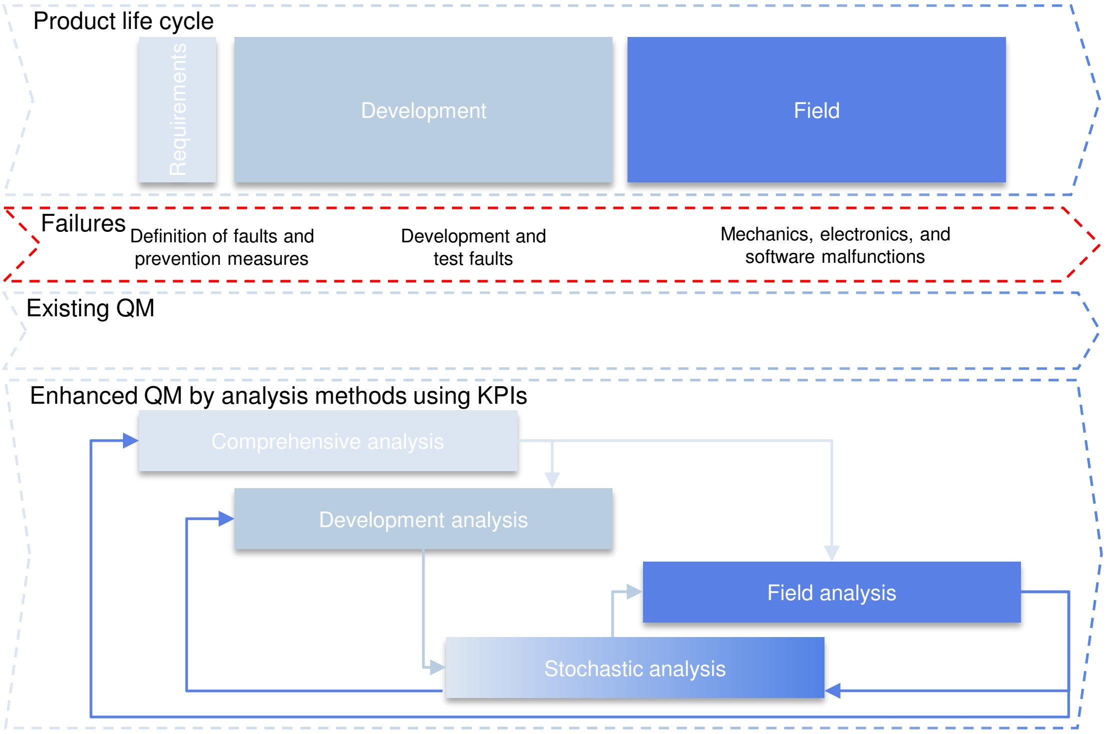

Figure 1-3: Introducing new analysis methods into relevant phases of product life cycle, c.f. [50]

According the requirement for the thesis, the main research question of this thesis is:

- Which analysis methods and KPIs applicable during different phases of the product life cycle have to be developed to enhance quality management and reduce faults of mechatronic systems in field use?

To define the main question more precisely related to different parts of product life cycle, the specific research questions are:

- How can the requirement specification phase and development processes be optimised by new analysis methods?
- How can development and test data be used to develop new analysis methods delivering relevant KPIs for different interest groups and division involved into the development process?
- What stochastic methods are useful for the reliability investigation and prognosis of residual error rates?
- How can field data be analysed and fed back to development?
- How can stochastic methods be used to compare development faults and failures in field use?

2 Product life cycle, development processes, and standardisation

2.1 Phases of the product life cycle

Today's industry is faced with the development and production of innovative products and/or complex systems. Furthermore, shorter processing times, cost pressure throughout the whole development process, increasing quality requirements due to product liability as well as dynamic market behaviour are additional challenges for companies in order to be competitive and successful. For this purpose, the interaction of OEMs and suppliers is increasing, resulting in the need for completely new methods to manage business processes in all phases of the product life cycle (PLC) – from planning, through development and production to customer use, service, and recycling. To handle these industrial challenges, mainly an optimisation of the product creation process (PCP) including the product development process (PDP) is required, [40]. The aim is to optimise the whole process by introducing new methods and solutions in early phases. For this purpose, an efficient product lifecycle management (PLM) has to focus on the data management to get correct information. Furthermore, a development of new analysis methods provides the investigation of processes during the whole PLC. PLM includes strategic, organisational, business-related, and technical measures for several phases, such as development, production, marketing, selling, service, and recycling, [40], [143], [178], [190]. The duration of PLCs depend on the industry and degree of innovation of the product. In comparison to the software industry, which is driven by innovation resulting in a short life cycle, the aerospace industry has a long life cycle due to high reliability, costs of production, and investment. During the PLC, various terms and conditions have to be considered, such as governmental requirements, quality improvement, measures for competiveness, product improvement, and cost reduction. After development, the production phase normally extends from the start of production (SOP) to the end of production (EOP), as shown in Figure 2-1. The length of the period between SOP and the product launch depends on the product complexity. In automotive industry this time lag is about six months. After production, the market phase starts with the introduction of a product on the market and ends with its removal from the market. The length of the customer field use depends on the type of product and industry. In the automotive industry, this phase is about 15 years and can extend up to 22 years including the production phase. The competition in the industry is driven by speed of innovation, reduction of development time, and increased requirements resulting in the so-called magic triangle consisting of quality, price, and performance. Thus, nowadays management of development and manufacturing represents a huge challenge. For this purpose, several tasks have to be considered which require a control system to ensure quality. In case of the automotive industry, the long PLC of 25 years represents an additional challenge, consisting of approximately three years of development, production of seven years, and 10 to 15 years of operation and service. Thus, OEMs and suppliers are faced to provide service after delivering the product to the customer to ensure the functionality over the product lifetime, [94], [143], [156].

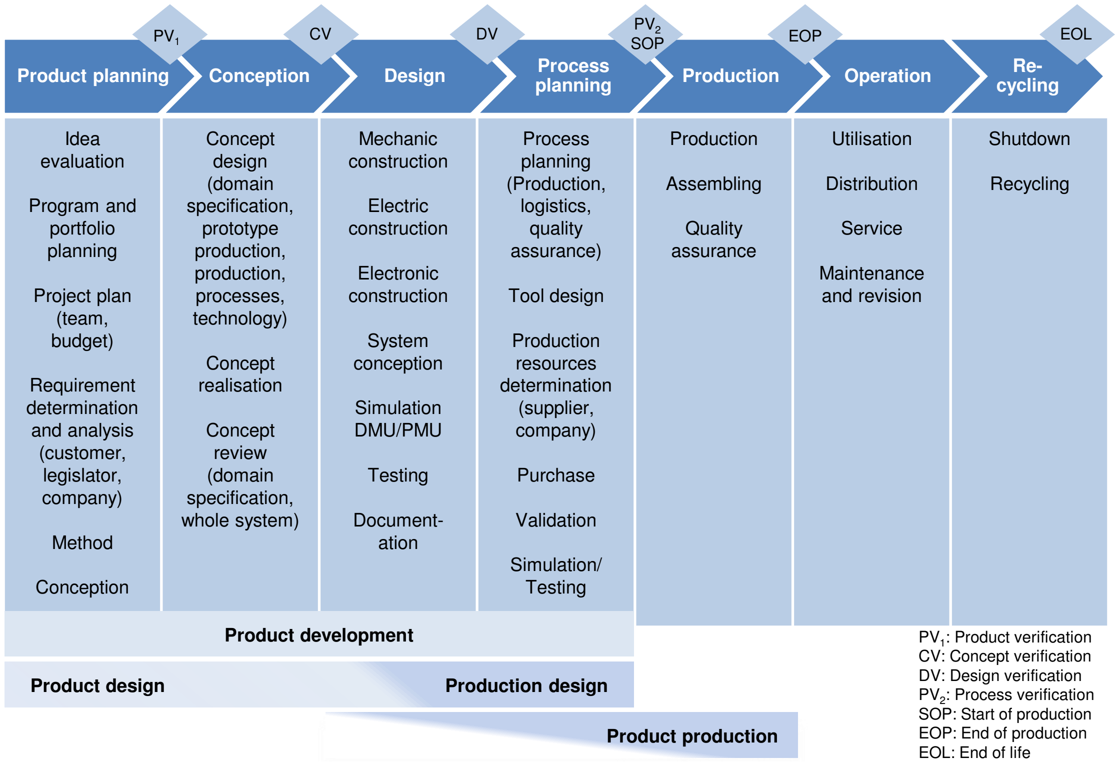

Figure 2-1: Phases, relations, and milestones of PLC, c.f. [40], [94]

2.1.1 Influences on PLC

During the lifetime of a product, various factors influence the companies and their products, especially in development process. Internal and external customers can influence requirements on PLC. External customers are society, employees, consumers, shareholders, and legislation. Internal customers are the departments like production, development, finance, and sales. The impact of these different requirements or stakeholders effects the following tasks and divisions, [143], [187]:

- Production planning
- Starting strategy
- Management of variants

- Commonality of series
- Component life cycle
- Continuous maintenance of products
- Market demand for innovative products
- Reaction on competition situations
- Optimisation of contribution margins
- Legislative requirements
- Process and organisational structure
- Quality optimisation

Reliability, availability, maintainability, and safety are characteristics with huge impact on costs and market acceptance. These influences result in different requirements on the entire PLC, such as safety-relevant, reliability-relevant, economic, social, human, and environmental requirements, [94]. The influences and the fact that the different disciplines, such as mechanics, electronics and electrics (E/E), and IT are getting more and more connected resulting in overlapping areas, are drivers for the improvement of mechatronic product development processes, [40]. Changes in development always represent big challenges on processes and tasks. The increasing product maturity over development phases includes decreasing possibility of reaction to changes. Late modifications always cause cost efforts that create unforeseeable additional impacts on development and management. Changes concerning hardware components are often more challenging than software adaption. After the SOP, changes are often done by software adaption and further releases, because hardware modifications are more expensive, [94].

2.2 Standardisation

There are various standards and guidelines addressing different phases of PLC and quality management. General basic standards represent guidance and tools for companies to ensure products meeting customer's requirements and high quality, such as, [27], [80], [188]:

- ISO 9000, [88]:
  The aim of the standard ISO 9000 is to enhance quality management systems to support continuous quality improvement by fault avoidance. For this purpose, this document includes basics and requirements due to customer needs to improve OEM and supplier connection.

- ISO 9001, [89]:
  This international standard includes requirements for a quality management system to ensure that companies meet customers' and governmental requirements.
- ISO 9004, [82]:
  The ISO 9004 focuses on how to make a quality management system more efficient and effective.
- ISO/TS 16949, [83]:
  For automotive applications, the standard ISO/TS 16949 represents a guideline for requirements for a quality management system considering the whole supply chain including design, development, production, and service. The main goals are avoidance of failures and decrease of waste.
- ISO/IEC 90003, [87]:
  The standard ISO/IEC 90003 considers software-related services including acquisition, supply, development, operation and maintenance of computer software, and related support services.
- ISO/IEC 25000ff, [86]:
  The ISO/IEC 25000 represents guidance for the use of various standards related to the so-called Systems and Software Quality Requirements and Evaluation (SQuaRE). The aim is to provide a general overview of SQuaRE contents, definitions, and relationships.
- ISO/IEC 15504, [81]:
  The standard ISO/IEC 15504 is an international initiative to support the development of an international standard for software process assessment, so-called Software Process Improvement and Capability Determination (SPICE). It includes three main aims to develop a consistent and validated framework for software process assessment, to conduct industry trials of the emerging standard, and to promote the technology transfer of software process assessment into world-wide software industry.
- ISO 26262, [85]:
  The standard ISO 26262 considers safety-related systems including electrical and/or electronic (E/E) systems. These systems are concerned to be within series passenger cars up to a maximum mass of 3500 kg. The standard addresses systems with a release for production as well as modifications after the publication date. In detail, it considers possible hazards due to faults of E/E safety-related systems and their interactions.
- VDI 2221, [181]:
  The VDI 2221 represents a general abstract approach and divides the development process into seven parts including additional iterations. The

first step of the approach is to clarify and define the tasks, followed by determining functions and structures of both integrated systems as well as subsystems. Next, the principles as well as their combinations are analysed and divided into releasable modules to enable following concurrent engineering. In the next part, the development of layouts for key modules as basis for different variants and the connection to the overall layout solution is necessary. The last tasks are to prepare product and operating instructions for final documentation. Advantages of this approach are high flexibility and development in modules which can be included into a platform [76].

- VDI 2206, [183]:
  The VDI 2206 represents a standard for the development of mechatronic systems. It includes a general problem solving cycle on micro level. On macro level, the V-model represents the basis of the development approach for mechatronic systems. Furthermore, process modules are used during the problem solution cycle to plan and process the various tasks. The problem solution cycle on micro level starts with analysing the situation and adapting goals. The next step is to search for solutions by a looped process delivering a basis for the situation analysis and objectives. After this, the defined solution variants are analysed in detail compared to the basis of the requirements and assessment criteria. Finally, the decision delivers the statement: Does the current result of the problem solving cycle satisfy the needs to go on planning or has it to be looped starting with the situation analysis and formulation of a goal? At macro level this standard follows the V-model (see chapter 2.3.1.1). The pre-defined process modules are recurrent steps in development of mechatronic systems. They enable an optimised process by the reuse of problem solutions and development procedures, [43]. The VDI 2206 delivers an efficient development approach for mechatronic systems considering all phases and sub systems. A disadvantage is that this approach stops with end of development and does not consider the field uses, [1].

2.3 Development processes

Due to the several requirements, development processes have to be plannable, flexible and controllable, [128]. There are different processes and methods ensuring a structured approach for development. These processes have to fulfil requirements, such as target-oriented processes, definition of responsibilities, use of milestones, use of interim results, use of tools, standardisation, documentation, [125].

Development processes often are influenced by special use cases and industrial areas like software engineering, mechatronic systems, or risk analysis. Development processes are used in various technical and business areas depending on their characteristics or applicability for other use cases. Furthermore, the preferred use of modified development processes also depends on territorial differences and standards. The following content incorporates and extends research of [41], [57], [137] and [16].

2.3.1 Development process of mechatronic systems

2.3.1.1 V-model

The V-model is an approach to plan and execute development projects. Besides defining tasks to reach the planned results, this method includes the specification of responsibilities during the whole development process. This method enables the development of complex systems and enhances planning as well as quality. The V-model was first published in 1997 by various German ministries [55], [56], [184] which developed this method at first for civil and military federal agencies. It is a combination of a top-down method as system/component design path and a bottom-up method as integration path (see Figure 2-2). Whereas the left side represents the specification and design of the system (later divided into sub systems and areas), the right side includes testing and integration (starting at component level, over hardware tests, resulting in an acceptance test of the whole system). The resulting V form enables an interaction of planned and fulfilled tasks using verification as well as validation. Both sides are connected to improve the product and development process by feedback of information, [56], [71].
The V-model represents a method for the development of complex systems. Several industrial applications use this approach. Due to the higher effort of documentation, small projects or products with less complexity are not the main target groups of this development process, [9].

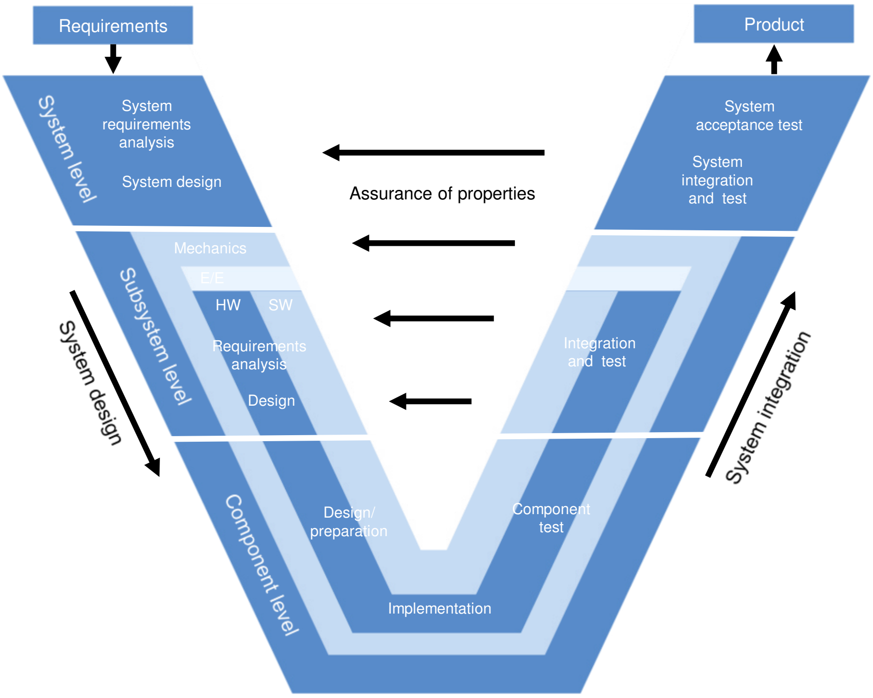

Figure 2-2: V-model, [44], [183]

2.3.2 Development processes of software

2.3.2.1 Waterfall model

The waterfall model is a sequential approach often used in software development. The approach is divided into several phases, such as system and software requirements, analysis, design, coding, testing, and operation. At the end of each phase, comparing the current result to defined criteria enables a check if the next step is possible or a loop of the previous phase has to be done, [153].

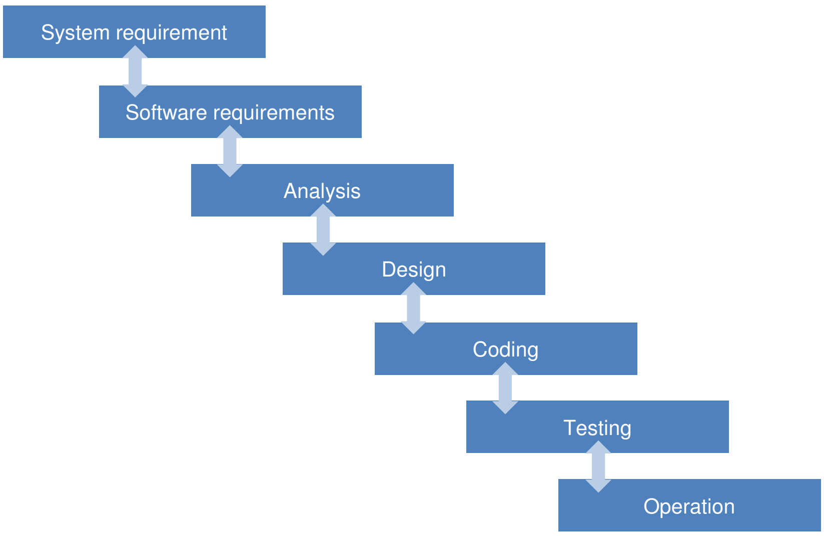

Figure 2-3: Waterfall model, c.f. [153]

One advantage is the easy use with less management and control effort. Furthermore, the waterfall model is commonly known and fits with milestone planning. Disadvantages are the decreased flexibility and need of cancelling as well as restart due to extensive changes. Thus risk management is difficult and most of the time not all requirements are known right from the start of the development process. Introducing iteration loops and validation can improve the model, [115], [153], [163].

2.3.2.2 Spiral model

The spiral model represents an enhancement of the waterfall model considering iterative processing. Focussing on software development, it delivers a risk-driven approach of a software development process rather than a primarily document- or code-driven process. Hence, risk analysis takes place at every stage. The approach starts at the centre of the spiral and the process is looping outwards by evaluating work by each loop. The spiral model uses different types of attributes, such as objectives to increase productivity, constraints representing limitations, alternatives as possibilities to achieve objectives, risks of different phases, risk resolution strategies, risk resolution results, planning next steps, and commitment, [14] , [163].

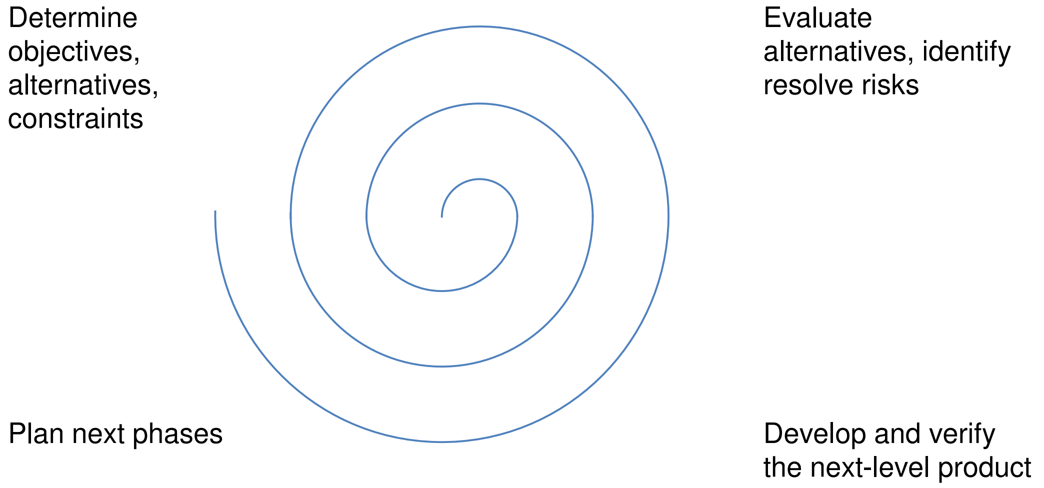

Figure 2-4: Approach of spiral model, c.f. [14]

One advantage of the spiral model is the risk analysis at each stage which enables an early recognition of deviations resulting in cost efficiency. Further advantages are the realistic approach and flexible use by reducing risks. A disadvantage is the huge control effort and expertise to use the model regarding risk analysis, [115], [163].

2.3.2.3 Scrum

Scrum is a development framework based on the main idea of agile and flexible project development of complex systems, normally used for software development. Some of the main targets are reducing risk as well as increasing quality, control, and customer satisfaction. The approach divides the system into subcomponents. Iterative-incremental steps, so-called sprints, are used to enable iterative development with short feedback loops. Main points are transparency, inspection, and adaption. The scrum team, which operates self-organised and interdisciplinary, consists of three roles: Product owner, development team, and scrum master. The basic workflow of scrum is the sprint with a duration of up to one month including all tasks that have to be fulfilled. The tasks of a sprint are planning, daily scrum, sprint reviews, and sprint retrospectives to enable control and adaption. Scrum artefacts are values to enable transparency and control to ensure right decisions and reduced risk, [164], [165].
Advantages of scrum are fast results due to short iterations in teams. Disadvantages can appear because of less involved persons outside of scrum teams resulting in less communication with the customer or management, [67].

2.3.2.4 Extreme Programming XP

Extreme Programming XP represents an agile software development method, which aims at high efficiency, quality control, and responsiveness to customer requirements. For this purpose, this method includes iterative development loops and freeze points to improve productivity. The development teams continuously communicate with customers to get feedback by testing the product as early as possible. The customer chooses stories for the next development phase which is divided into several tasks for the team. The implementation status is checked by test cases and releases are a constitution of various stories developed by iterations, [7].

Advantages of this method are the flexibility and feedback loops with the customer. It is an iterative- and test-based process to meet requirements and quality assurance. The self-organisation of the teams and open communication can lead to disadvantages caused by lacking experience. Large projects are difficult to handle due to documentation and test effort, [115].

2.3.3 Development processes for general use

2.3.3.1 Stage-Gate®-process

The Stage-Gate®-process represents a business process and risk model to improve development as well as innovation including the targets product innovation excellence, product leadership, accountability, market-relation, high-performance teams, customer focus, robust solutions, alignment, speed, and quality. The aim is to enable correct information with the appropriate level of detail to support decisions, allocate capital, and operate resources. This process includes pre-development, development, and commercialising activities. The main idea is to break the complex approach down to smaller stages for tasks and introduce gates for decision-making. Stages represent simultaneous or cross-functional activities and tasks to gather information resulting in a risk reduced decision point. In general, there are five stages: Idea discovery, scoping, build the business case, development, testing/validation, and launch. Gates represent quality-control points for decision-making to define if a project is allowed to pass through following three main targets: Ensuring quality of execution, evaluating business rationales, and approving project plan as well as resources, [24], [175].

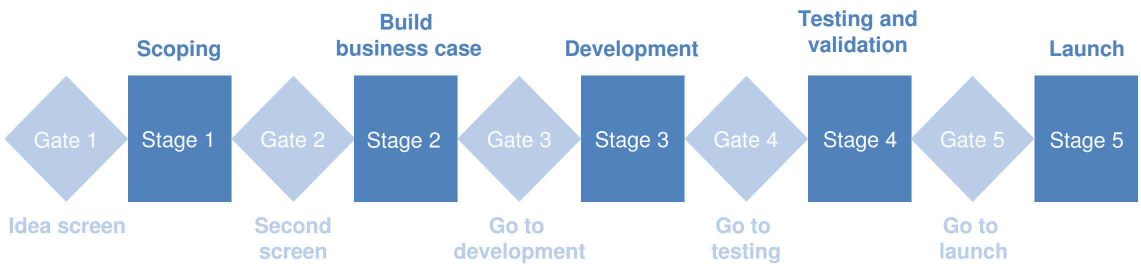

Figure 2-5: Approach of the Stage-Gate®-Process, c.f. [24], [175]

The Stage-Gate®-process is suitable for every discipline and easy to proceed. However, the sequential approach leads to decreased flexibility and the possible risk of tardy project cancelling, [24], [175].

2.3.3.2 Kanban

Kanban is a method for process improvement based on the just-in-time principle and lean production. Kanban method tries to enable flexibility and the use of existing capacity. Using Kanban in development durations of tasks is a key metric for teams to deliver an optimised development planning by minimising durations. Kanban development teams work self-organised on working packages during a development phase. A so-called Kanban board enables splitting development projects into different phases or working packages to ensure process flow and avoid

overloading of teams. Using work in progress definitions, simultaneous working of teams in different tasks is avoided to increase efficiency, [126].
Advantages of Kanban are efficient workflows and teams as well as the control of progress using metrics. Disadvantages are less connection of processes or tasks and the difficult application on development of mechatronic systems, [126].

2.3.3.3 Munich Procedure Model

The Munich Procedure Model is a method to support product development processes by using principles of system engineering. Furthermore, it delivers a guideline for problem solving processes. The Munich Procedure Model contains seven steps structured in a net matrix to enable different ways throughout the approach if a path cannot be passed. Thus, this method delivers a flexible approach. The seven steps are target analysis, target planning, problem definition, checking solutions, evaluate solution characteristics, decision-making, and target hedging, [109].

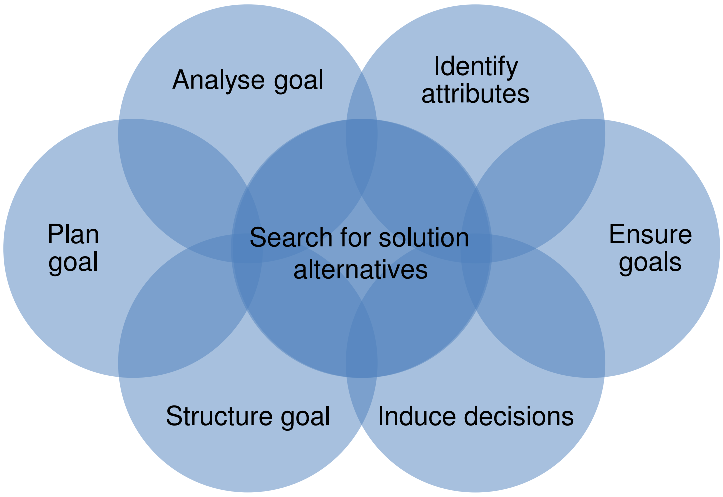

Figure 2-6: Munich Procedure Model matrix, c.f. [109]

The Munich Procedure Model is an approach to support development processes rather than a development method. On the one hand it is a flexible problem solving method, on the other hand there is a risk of losing the main objective and progress of development, [109].

2.3.4 V-model in a detailed view

The V-model satisfies the conditions and needs of mechatronic development. Besides the efficient design and specification phase of different subsystems, the testing and integration path enables effective verification and validation. The V-

model is commonly used in Europe and parts of Asia. Hence, the further investigations are based on this development process.
The following content incorporates and extends the research of [41], [57] and [113]. The development of mechatronic systems represents a challenge for process, approach, and development teams. Besides the integrated system, different subsystems such as mechanics, E/E, and software have to be considered during development. Furthermore, the integration of the different domains needs several types of testing procedures. For this purpose, quality assurance for validation and verification have to take place in the development process. The V-model enables the development of such complex systems and leads to increasing planning and quality. It is an approach to plan and execute projects during development. In addition, it defines the results and tasks to reach this results as well as the specification of responsibilities during the whole development process, [56], [71]. The comparison of the different development processes shows a good fitting of the V-model for development of mechatronic systems.

2.3.4.1 V-model XT

The V-model XT is an enhancement of the V-model due to new technologies and development requirements. “XT” stands for “eXtreme Tailoring” and represents the adaptability in different development circumstances. The aim is to present a guide to plan, execute, and realise projects considering the following topics: Minimisation of project risks, improvement and guarantee of quality, reduction of total cost over the entire project and system life cycle, as well as improvement of communication between all stakeholders, [56], [71]. The V-model is suitable for automotive mechatronic systems, especially those requiring high safety and reliability. Due to the project conditions and circumstances, V-models can differ from each other. The common denominator are the approach and regulation of “what”, “who” and “when”. With these different terms and conditions, projects can be classified and divided into project types, such as projects to award a contract to a supplier, projects to develop a system as supplier or as acquirer/supplier, as well as the introduction and maintenance of organisation-specific process models. Due to these project types, the project and relationship to stakeholders differ. The project type variants result in a so-called project execution strategy. This project execution strategy leads to decision gates indicating a project progress state to evaluate the current project status. This approach supports making decisions during development processes. Furthermore, a process module is used to define the actual task which has to be accomplished during the project. These modules specify the so-called work product which has to be developed, the activity to develop the product, and the roles representing responsibilities in scope of tasks, [71].

2.3.4.2 Specification and design process

The input of the V-model are requirements of customers – both technical and non-technical – representing the functionality, behaviour, properties, and quality of a system that have to be provided by the development process, [79], [111]. In this context, technical requirements are related to properties of products or services and non-technical requirements affect the product, service, or development and are not properties of the product. Requirements management is a dynamic process and ensures that changes to requirements fit project plans or activities to control the engineering process. The successful implementation of requirements challenges suppliers because customers' expectations can differ from the delivered product. While suppliers focus on a clear definition and declaration of requirements, customers demand unrestricted requirements, [21], [172], [185]. For this purpose, requirements engineering is a very important part and covers several activities, such as, [35], [111]:

- Requirements elicitation:
  Requirements elicitation represents the process of discovering, reviewing, documenting, and understanding customers' needs and constraints for a system.
- Requirements analysis:
  Requirements analysis is the process of refining customers' needs and constraints.
- Requirements specification:
  This phase is the process of documenting customers' needs precisely.
- Requirements verification:
  Requirements verification ensures that the system requirements are correct, complete, consistent, and clear.
- Requirements management:
  In this part, engineering activities are scheduled, negotiated, and coordinated to ensure requirements.

The left side in the V-model represents the specification and design process of the system and subsystems. Thus, the development process is specified by differencing so-called subjects. This supports a concrete framework of the development approach for software (SW), hardware (HW), complex or embedded hardware and software systems, as well as the system integration. Due to the use of the V-model as a process model for development, planning, and executing various basic management divisions and mechanisms are required, such as project management, quality assurance, configuration management, and change management. For this purpose, the system can be divided into subunits during development. These subunits are subdivided into segments, hardware units, software units, external units, hardware and software components, hardware and software modules, and external hardware and software modules. Due to a uniformed pattern, every subdivision step permits a complete tracing of the requirements. Thus, the requirements of the higher system elements are taken into account during the designing and specification process of lower level system elements, [71].

2.3.4.3 Test process and environment

In case of realisation and integration of the system, the process conducts into the reverse direction, from hardware/software modules and more complex system elements to the system integration. This process represents the right side in the V-model. When using the V-model, a very important part in the development process is to ensure verification and validation at every design or realisation level. This enables testing of the realised to specified system. For this purpose, every validation and verification level maps to an accompanying level on the design side. The validation and verification steps or methods are defined during specification phases, which enable target-oriented development, [71].

The verification process is used to check if the system requirements and specifications are fulfilled or implemented during development. The validation process is used to evaluate the system, which shows if the developed system satisfies the needs and fits its mission. Verification means “doing things right” and validation means “doing the right things”, [13], [110]. For this process, the following different types of tests are used in the special case of software development, [15], [156], [180]:

- Module or component tests:
  Module or component tests are carried out at module level to test functionality, correctness, and fulfilment of specification. At this certain point of time and due to cost management the hardware components are not or not completely available. For this purpose, often simulation substitutes expensive hardware or parts to test modules and components. To get an efficient test coverage and costs/effort balance, test methods (white box tests) are necessary to maximise test coverage, such as checking special parameter or critical parts.

- Integration tests:
  Integration tests enable the check of interactions and interfaces of different components or modules. These systematic tests use black box or white box testing methods.
- System tests:
  System tests are used to check if all functional and non-functional requirements are included in the product. The aim is to control if the system delivers the right output depending on specified inputs, usually carried out as black box-tests. Many different test scenarios can be considered, such as robustness (investigating utilisation and real-time capability), recovery (checking correct behaviour after disturbances), configuration, usability, safety.
- Acceptance tests:
  The customer usually carries out these tests to control if the system fulfils the product requirements. In this phase, a close interaction and feedback loop between the supplier and the customer is advisable.

Considering the knowledge of the internal structure of the system, the following types of tests are classified, [6]:

- Black box test:
  This testing method enables checking the functionality of a system without knowledge about the internal structure, functions, or the source code. Thus, the review is limited to functionality and correctness of external behaviour due to input of defined (random) parameters. To handle huge test effort, methods such as classes of equivalence or boundary value analysis (fail-safe-test) are used. This testing method is used in every level of automotive development, such as modules, hardware, integration, or system tests.
- White box test:
  The white box test allows checking the functionality of a system including internal and external behaviour. Thus, it requires knowledge about the internal structure or functions as well as the source code. Particularly during testing subcomponents or modules the information about the internal circumstances is needed to optimise bug fixing and further development by locating faults in the system.
- Grey box test:
  During development, it can happen that black box tests are insufficient to get a significant evaluation of the system. Thus, the grey box test is used as a compromise between black and white box test because in addition to

the input/output interaction they deliver access to a set of internal status parameters.

Due to time and cost pressure companies are faced with the challenge that module, component, or system tests often cannot be carried out including all necessary hardware and software components. To provide quality and safety of the produced system and the entire development process, modelling and virtual interfaces are introduced to replace the physical equipment by a virtual environment, so-called in-the-loop-tests. This enables early, repeatable test environments at lower costs. It has to be considered that these testing methods are always performed in a so-called protected environment and cannot include all real life conditions and circumstances. For this purpose, in comparison to field use or prototype tests deviations can occur. In-the-loop-tests are used for validation and further development of software and hardware components. The following types can be differentiated, [147], [156], [180]:

- Model-in-the-loop (MiL):
  The component, module or system to be tested is integrated as a function model in a virtual test environment which represents an interface delivering input signals as close to reality as possible. The aim is to control and guarantee the functionality of the module. Advantages of this test method are less time and cost effort as well as the possibility of early changes in development. Disadvantages occur because of the protected test environment. Hence, there is always a risk of deviations to reality.
- Software-in-the-loop (SiL):
  SiL tests usually do not use special or terminal hardware, instead the compiled code is tested by using a virtual simulation model. The virtual test environment is similar to the MiL test stage, usually carried out at the development CPU. An interface enables the communication between the code and the virtual environment. Advantages of this test method are the possibility to verify the software system without the final hardware by considering internal code segments, which enables an exact test depth. Faults can be fixed early in the development process without huge costs effort. Disadvantages are similar to those of MiL.
- Function-in-the-loop (FiL):
  Like SiL, this test method tests software codes, but in comparison to SiL the terminal hardware is used. This method combines testing software on the original hardware while using the white box test approach.

- Hardware-in-the-loop (HiL):
  The HiL level extends from testing individual software and hardware to checking the integration of systems within the whole vehicle. At the beginning, some environment can be included through a virtual model. Besides the integration and interface check, real time capability, reliability, and functionality are the main points of this stage. To be as close as possible to reality more hardware and realistic environment should be included at the test bench, such as ECUs, actuators, sensors. Advantages of this testing method are the close simulation to reality combined with reproducible testing conditions by saving costs in comparison to real vehicle tests. Disadvantages are using black box tests without knowledge about the software code, the lower test depth, and the complex introduction of real-time environments.

2.3.4.4 Limits of V-model

The V-model is limited by different impacts such as, [56], [71], [156]:

- V-model does not regulate placing of contracts for services
- No distinction between acquirer and supplier due to the organisation-specific process model
- No consideration of different post processing phases, such as maintenance, repair, and disposal
- Missing feedback of earlier development phases which includes a late and complicated detection or consideration of faults due to early phases
- Requirements modifications and changes can be combined with high costs and risks
  →This circumstance leads to spiral looping of the V-model in practice

2.4 Management of development processes

To control and manage development processes different approaches and roles have to be defined.

2.4.1 Project management process

A very important point is to identify all relevant project requirements and address the various needs or expectations of stakeholders to deliver the best results. Due to different project circumstances and characteristics, project management has to focus on specific priorities. To balance project constraints many different factors have to be considered that influence each other, such as scope, quality, schedule, budget, resources, and risks. Due to the short time schedule, some of them may be left out. So, balancing the requirements and expected results is very important to deliver a successful project. Thus, project management is closely connected to operations management as well as the organisational strategy. Whereas operations management deals with directing and controlling operations to achieve strategic goals of business, organisational strategy considers changes and decisions of operations, development of products and systems, strategic direction, and performance parameters. Changes to project objectives affect project efficiency and success which always have to be aligned with business direction. To ensure the fulfilment of project requirements in time, project management process is necessary, including all relevant techniques, skills, and activities divided into five parts, [140]:

- Initiating:
  The project management process starts with defining the project by obtaining authorisation. Therefore, at first the initial scope and financial resources as well as the internal and external stakeholders are identified. The next steps are selecting the project manager and appointing project boundary conditions, such as start and completion date, project phases, always in compliance with the individual stakeholders.
- Planning:
  The planning process requires establishing the scope of objectives of a project by scheduling different actions and processes. Due to changes in the product life cycle and necessary feedback loops, this process is iterative (so-called progressive elaboration). During this planning process, a lot of project data and documents are generated including different information and aspects, such as time, scope, quality, costs resources, risk definitions. Thus, impacts, changes, or updates of this data are always combined with precise investigation of time schedule, resources, requirements, costs, and monitoring of tasks to meet the defined scope.
- Executing:
  In this phase, the project plan is completed by satisfying specifications and requirements. Hence, coordinating all involved divisions, resources and integrating all needs or views is important. This phase is also faced with impacts and updates which results in changes of tasks, resources, productivity, and risks.
- Monitoring and controlling:
  This part is very important to track and review the progress as well as the performance of a project. This ensures the identification of areas that require changes and enables control of the development process. The most important point in this phase is to evaluate and analyse the whole project performance: Check changes, monitor ongoing tasks in comparison to the

milestone or project plan and investigate the entire project effort. Besides controlling and monitoring, preventive measures are a powerful tool in this phase.

- Closing:
  This phase is used to finalise all activities and processes to close the project. Besides finalising completed projects, closing aborted or cancelled projects as well as considering critical projects takes place. In addition, many tasks like obtaining acceptance of the customer, documenting, recording impacts, lessons learned and conducing end reviews are conducted.

2.4.1.1 Different roles and perspectives

To ensure an efficient and effective project management process different roles and responsibilities have to be defined. Hence, a suitable environment in the development process considers the following roles, c.f. [21]:

- Management:
  The management ensures an efficient and effective working of an organisation. The responsibilities include review and management of performance, taking corrective and preventive actions, efficient resource management, as well as definition and improvement of organisational processes. One of the most important decisions is to release a system or product at a certain point of time.
- Project management:
  The project management leads a specific project and integrates the efforts of different parties to fulfil all requirements of customers on time. Furthermore, it ensures effective implementation of organisational processes, guidelines, and standards. Furthermore, project management is responsible for activities of project requirements engineering and schedules. One of the main points is the efficient planning of project-related resources like people, time, and money. Project management leads the team that is responsible for achieving project objectives and satisfying several needs. This division has to fulfil many skills, such as leadership, team building, motivation, communication, influencing, decision-making, negotiation, [140].
- Process management:
  The process management controls the definition and improvement of different processes in an organisation. Furthermore, this division is responsible for review and feedback to improve processes. It also induces quality control and change management of all processes.

- Quality assurance:
  The quality assurance includes various areas due to quality control. The tasks of this division range from failure prevention to defect detection. Quality control is important to prevent defects and deliver a nearly defect-free product. Some of the main responsibilities are improvement of quality, verification, validation, and defect reports.
- Business/system analysts:
  Business/system analysts are responsible for requirements engineering process and control. Primarily, they have to ensure the deliverables and best possible quality within an accepted schedule. Furthermore, they have to take care about analysis and establishment of requirements, traceability matrix, testing, efficient completeness, and quality control. These activities can affect one project or the complete organisational management.

2.4.1.2 Project data and information

During the life cycle of a system, large amounts of data can be collected which is a very important information source to analyse and control development as well as field uses. The storage of data as well as the connection and structure of databases has to be considered. Relevant data can be, [140]:

- Work performance data:
  This data includes observations and measurements during activities performed to carry out project work. Examples include reported percentage of work physically completed, quality and technical performance measures, start and finish dates of schedule activities, number of change requests, number of defects, actual costs and actual durations.
- Work performance information:
  The performance information is collected from various controlling processes, analysed in context and integrated based on relationships across areas. Examples of performance information are status of deliverables, implementation status for change requests, and forecasted estimates to complete.
- Work performance reports:
  Reports are a physical or electronic representation of work performance information compiled in project documents intended to generate decisions or raise issues, actions, or awareness. Examples include status reports, memos, justifications, information notes, electronic dashboards, recommendations, and updates.

2.4.1.3 Milestone planning and quality gates

To manage product development processes, especially for complex systems such as the automotive industry, using milestones is essential. Milestone planning enables coordination and control of all phases, processes, and divisions during the development as well as the entire PLC. The mapping of development process phases and milestones enables points of synchronisation resulting in so-called quality gates, [83], [88], [182]. The combination of structured development phases, defined roles, distribution of responsibilities, and milestone planning enables a controllable approach, [65]. These quality gates support management of development processes by delivering a method to measure and evaluate the progress. Quality gates are result-oriented points of evaluation controlling process-related metrics. They support release decisions and quality management as well as control by delivering early possibilities to introduce measures by project deviations. For this purpose, the basis for quality gates are quantitative metrics delivering an objective evaluation of the development process and progress, [77]. Thus, the difference between milestones and quality gates is that milestones represent certain points of time where results have to be fulfilled, whereas quality gates are content-related measures delivering synchronisation. Milestones can be driven over, while quality gates cannot, [138].

3 Quality management, KPIs, and reliability investigation

Quality management has to ensure quality over time and the so-called reliability of a product to enable sustainable success and customer loyalty, [8], [11], [121].

3.1 Quality management

3.1.1 Quality basics

Quality (lat. "qualitas"= inner structure, condition) represents the structure, value, and condition of a system or product, [18]. The definition of quality is changing and is interpreted in different ways, [17] , [27]:

- Sanctity of contracts = Conformity to specification (requirements)
- Customer-oriented = Fulfilment of requirements and expectations
- Competitive-oriented = Attractiveness
- Norm definition = Requirement fulfilment degree of a set of inherent characteristics [88]
- Management = Level of target achievement
- Amount of immanent properties of a product or process
- Scale to reach a customer buying decision
- Factor due to intensive interactions of a competitive situation and the capability of the supplier [135]

Besides representing product characteristics, quality affects basic requirements, processes, measurements, and methods. This includes functions of systems, reliability, support and service, resources management, design, warranty, and complaint management, [121]. Thus, quality represents a relative conformity of a system, process, or product with predetermined requirements and is connected to reliability representing the long-term quality behaviour, [32], [33].
The quality of products and processes represents the success of companies for the following reasons, [17]:

- Global competition:
  Many companies, particularly in high-wage countries, are faced with cost pressure due to decreasing geographic differences. Using high quality enables a differentiation between these companies.

- Increasing customer expectations:
  Customer expectations concerning quality are increasing. Particularly reliability, service, efficiency, environmental excellence, and handling represent crucial significances. In comparison to the buying decision where costs are the most important point, quality represents a sustainable meaning for customers including complaint management.
- High complexity:
  Complexity of products and processes is increasing due to diversity of variants and challenging development projects. Nevertheless, quality management has to ensure safe and reliable products, because deficits in quality implicate a loss of customers.

Quality is an important issue to sustainable success and customer loyalty. Furthermore, cost can be reduced due to avoidance of faults. An important value for customer decisions is their benefit calculated as quality divided by costs. In addition, this quotient represents a comparative value to competitors resulting in strategies to improve product quality as well as reduce faults and costs. These strategies can be, [121]:

- Improvement of end product quality:
  Continuous improvement as well as innovation delivers better customer satisfaction, including technical aspects, handling reliability and service.
- Improvement of process quality:
  The improvement of process quality enables reducing faults and costs.

Efficient and effective quality management represents the basis for high quality. Hence, this division has to focus on management tasks, such as, [121]:

- Planning and decision:
  - Determining quality
  - Determining quality strategies
  - Determining quality targets considering product and market strategies of the company
  - Defining of basic principles
  - Organising and creating of a quality management system
  - Controlling, approaching, and processing
  - Controlling resources and integrating QM in the company structure
  - Determining responsibilities and power
- Human relationships:
  - Supporting QM
  - Reviewing regularly

  - Informing and communicating internally/externally
- Controlling:
  - Comparing results to targets and deriving measures
  - Defining of indicators (KPIs)
  - Controlling efficiency and effectivity with internal audits and evaluation
  - Measuring customer satisfaction
  - Handling the complaint management
- Continuous improvement and innovation:
  - Determining targets and structures to support the continuous improvement process (CIP)
  - Training and sensitising employees and suppliers

In detail, quality management is divided into five divisions, [121]:

- Quality politics:
  Quality politics defines the framework of actions and targets considering company strategy. Thus, it is a part of the company's basic principle. Quality politics as part of integrated company politics defines how quality leadership is used for differentiation between other companies or products.
- Quality planning:
  Due to requirements of the customer, the division quality planning defines targets, characteristics, and requirements of processes and systems. To support decision-making in this division, the structure of quality politics in the company is defined. Relevant parameters are product, market, positioning, and market strategy of the company. Furthermore, superordinate quality targets, such as number of complaints, customer loyalty, costs of faults, are used to define target parameters. In addition, choosing the right quality strategy is very important, e.g. 0-failure-strategy, CIP, QMS, Six sigma.
- Quality assurance:
  Quality assurance supports control of reaching quality targets including quality control, primarily resulting in a basis of trust with the customer. Quality assurance is defined as a planned and systematic pattern including all actions which are necessary to provide adequate confidence of a product to fulfil all requirements. For this purpose, quality assurance consists of a constructive and an analytical part. The last one delivers the state of quality of a product or system and detects bugs, [84], [188].

- Quality control:
  This task includes providing resources as well as preventive, controlling, and correcting tasks to realise quality targets. Furthermore, carrying out fault correction and applying elimination of fault potentials are important steps, [84], [89]. Quality control influences both product and development processes by controlling quality parameters within their defined boundaries. Quality and process parameters are driven by various influence factors such as motivation, human qualification, overload, machine precision, stability of environment, long-term stability, material dimensions, chemical and physical properties, methods, processes, parameters, contemporaries such as temperature, air humidity, pollution, measurement (wrong or incorrect), management and decisions, [77]. Quality control has a lot of tools and tasks to ensure high quality and efficiency. In case of development processes, they are for instance evaluation of suppliers, entrance control, evaluation of risk, measures for process stability, preventive measures, complaint management. Comprehensive tools are for instance internal audits, improvement of QMS, support of CIP, development and introduction of QM-related methods, [89].
- Quality support (CIP):
  The Quality support uses CIP to improve products, systems, and structures of an organisation. This affects costs, time, and efficiency. Management has to ensure a continuous quality control to recognise deviations as soon as possible. Thus, control processes and methods have to be applied in the company. Thus, the development of an indicator system on basis of company data using KPIs is required, such as Balance Score Cards (BSC). In addition, regular internal audits or evaluations of processes are necessary to control internal and external circumstances. Customer satisfaction and combination of complaints, customer audits, as well as supplier evaluations are efficient tools to meet quality targets.

3.1.2 Quality and fault characteristics

Quality and product characteristics are the result of the customer requirements specification and deliver quality checks, [121]. A fault represents the cause of a deviation and a potential failure within the system or product, [188]. Thus, faults are a non-fulfilment of requirements, such as, [121]:

- Non-fulfilment of customer requirements
- Non-fulfilment of quality or product characteristics
- Process parameters outside of boundaries
- Badly performed tasks

Faults are always considered as decreased success of a company, increasing costs and bearing a risk of decreasing customer satisfaction. Thus, fault prevention is an important tool to provide quality. Furthermore, faults can be differed into the following groups due to their significance, safety, and influence, [121]:

- Critical faults:
  Critical faults affect dangerous situations for humans and machines combined with high damages and costs.
- Major faults:
  These faults probably lead to a failure or breakdown of the system/product.
- Minor faults:
  These faults do not influence the usability of the system/product.

High test coverage values are necessary to maximise fault detection. Sometimes tests are carried out as random samples to save costs, [121].

3.1.3 Improvement and optimisation of quality management

Quality represents the basis for successful products to satisfy requirements, enable customer loyalty, and ensure long-term success of companies. To meet increasing customer expectations to quality, decreasing complaints during the whole PLC, and ensuring reliability that results in long-term usability of products, strategic and continuous optimisation of quality management are necessary. Hence, new analysis and quality evaluation methods as well as continuous investigation during the whole PLC are essential. This avoids influence of a negative image or high warranty costs due to recalls resulting in possible business damage. In practice two different types of quality are differed, [62], [101], [143], [187]:

- Result quality:
  - Product quality:
    - Functional quality
    - Long-term quality
  - Service quality
- Process quality

The most important point is to satisfy customer needs by delivering a reliable product. This avoids complaints and substitution of a product combined with high warranty costs. To reach this quality level, a continuous optimisation of quality assurance during the whole PLC is necessary. For this purpose, two strategic approaches should be pursued, [143]:

- Short-term quality measures:
  Short-term quality measures are fast and target-oriented measures to restore product quality, primarily in case of safety-relevant systems which require immediate acts to avoid comprehensive recalls.
- Continuous quality improvement and stabilisation:
  Continuous quality improvement is related to non-safety-related functions and characteristics. The optimisation of quality in these areas supports the customer satisfaction and long-term business success.

The improvement of quality follows preventive and reactive quality measures, [69], [143]:

- Preventive quality measures:
  Preventive quality measures serve to improve quality and avoid weak points in advance. As part of the continuous quality improvement drivers for this tasks are optimisation measures due to competition, customer requirements, and complaints.
- Reactive quality measures:
  During operation, faults and deviations should be fixed as soon as possible depending on their criticality. Thus, continuous knowledge building and evaluation of faults are very important tasks.

During development both product effectivity to fulfil customers' requirements by optimising product value and product efficiency to ensure a customer-related realisation by controlling cost drivers have to be considered. A customer-related development uses traceability, testing strategy, and fault diagnostics, [159].

3.1.4 Costs of quality

The calculation of quality management costs is complex and often not comparable. Fault prevention costs represent the lower boundary of these costs but are often too little to get sufficient quality. Quality costs consist of the following categories, [121]:

- Fault prevention costs:
  These costs include expenses of all control measures, such as risk evaluation, trainings, audits and supplier evaluation.
- Test costs:
  Test costs represent efforts of processes and tasks to ensure product characteristics fulfilment of customer requirements. Examples for these tests are for instance entrance control and end tests.

- Fault costs:
  These costs appear because of a non-fulfilment of quality requirements. They can be divided into internal fault costs, such as waste, reworking, production stop, and external fault costs like warranty and complaints.

Following the “rule of ten”, fault costs later increase ten times in every phase during the PLC, as shown in Figure 1-2. Increasing effort of quality management decreases costs due to faults and malfunctions in the field use. For this purpose, a good balance between effort of quality and cost saving is important to ensure high quality as well as less warranty costs, [50], [52], [121], [187].

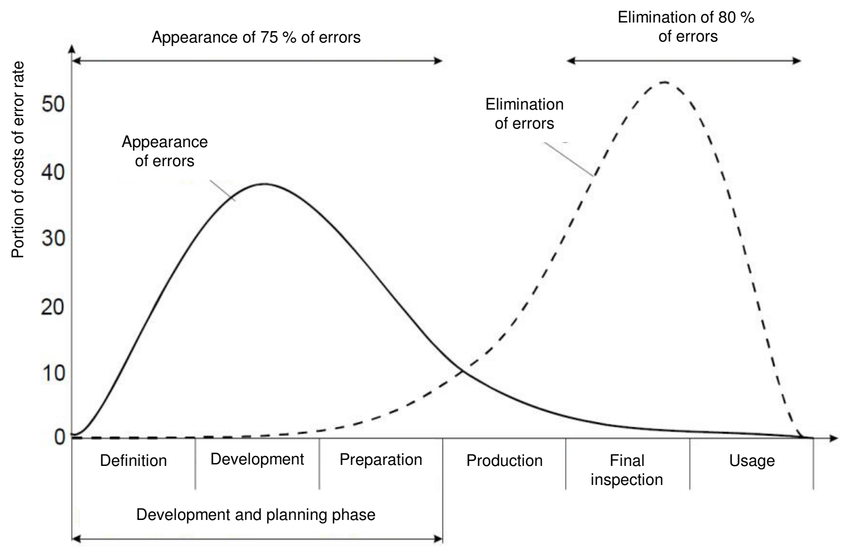

Figure 3-1: Fault evolution and removal c.f. [192]

3.1.5 Quality management methods and tools

Today’s quality management uses different methods and tools to ensure quality of systems considering planned approaches, preventive use, various effects and development phases. The state of the art shows methods and tools reaching from abstract approaches to precisely defined guidelines, such as, [17]:

- Problem solving methods:
  If quality management detects a deviation, this method is used to define the reason and solve the problem (such as the general problem solving approach). This approach includes target determination with boundary conditions, problem analysis as well as description of the fault/detecting and cause research, finding of solutions with research and evaluation, measures implementation, and standardisation. Similar to this approach, the PDSA cycle use the steps plan, do, study, and act to solve a problem,

[29]. Another method which is also common in automotive industry is the 8D method to evaluate and solve a problem, [93].

- Quality tools:
  Different quality tools are used to solve problems and improve quality in a company, such as flow charts, fault collecting cards, histograms, pareto analysis, correlation diagrams, Ishikawa diagram [91], and traffic light charts.
- Quality function deployment (QFD):
  QFD is a systemic quality planning method using customer orientation. "What" expects the customer and "how" are requirements fulfilled? The result is a relationship matrix, the so-called House of Quality, [119].
- Fault method effects analysis, FMEA (see chapter 3.3.1.1)
- Fault tree analysis, FTA (see chapter 3.3.1.2)
- Statistical process control (SPC), [167]:
  The production environment and circumstances are changing throughout the production timeline. The SPC delivers evaluation and control by use of quality regulation cards. This method depends on small fault rates, detects deviations, enables controllable and able processes, uses representative random samples and enables statistical safeness.
- Poka-Yoke, [168]:
  Poka-Yoke was developed for the Toyota production system and influences various processes based on an assumption of a zero-fault-strategy. Even if zero faults are not realistic, this method interprets a non-zero-fault strategy as expectance of faults. This method uses the approach to avoid random failures during production process, because human failures are natural and cannot be avoided. Poka means random unintentional failure and Yoka means avoidance or reduction.
- Total quality management (TQM):
  Total quality management (TQM) is an approach to ensure quality in a company, including internal and external customers, processing improvement, developing quality in the organisation, and developing a measurement analysis program. The idea is that everyone in the organisation is both customer and supplier. So the main target is to reach customer needs and quality expectations by a holistic approach to enable all functions of the organisation following high quality standards. TQM considers all areas within the organisation by identifying areas for improvement. The evaluation and analysis of these areas use collected data, [127].

- Stochastic quality methods (see chapter 3.3):
  During production system characteristics scatter due to various circumstances, such as vibrations and systematic deviations (like wear out effects). Hence, quality management uses a definition of boundaries to control production processes, [152]. Various stochastic analysis possibilities use descriptive statistics, quantitative and qualitative characteristics, medians, variances, and distributions to describe and evaluate development processes.

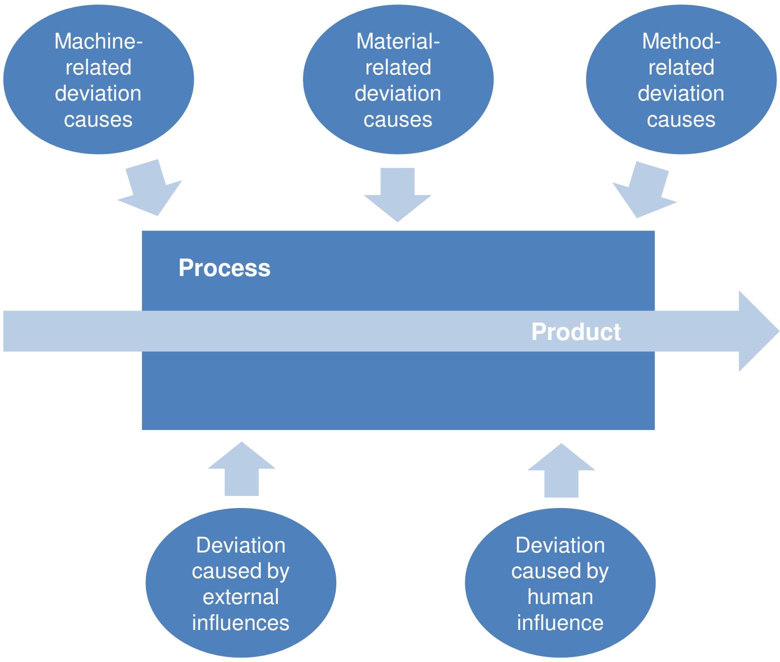

Figure 3-2: Possible deviation due to process and product, c.f. [17]

- Modern quality management tools:
  The later a defect is detected in the PLC, the higher the costs to fix it. Thus, it is advisable to detect a fault or deviation as soon as possible during development, preferably within the relevant phase in which it was produced. For this purpose, quality management has to introduce a detection process. To ensure high quality, many tasks should be considered, such as, [127]:
  - Project management
  - Estimation methodology
  - Risk management process
  - Requirements development and management
  - Design and development
  - Software development lifecycles
  - Quality assurance/management

- Software inspections
- Software testing
- Supplier selection and management
- Configuration management
- Customer satisfaction process
- Continuous improvement

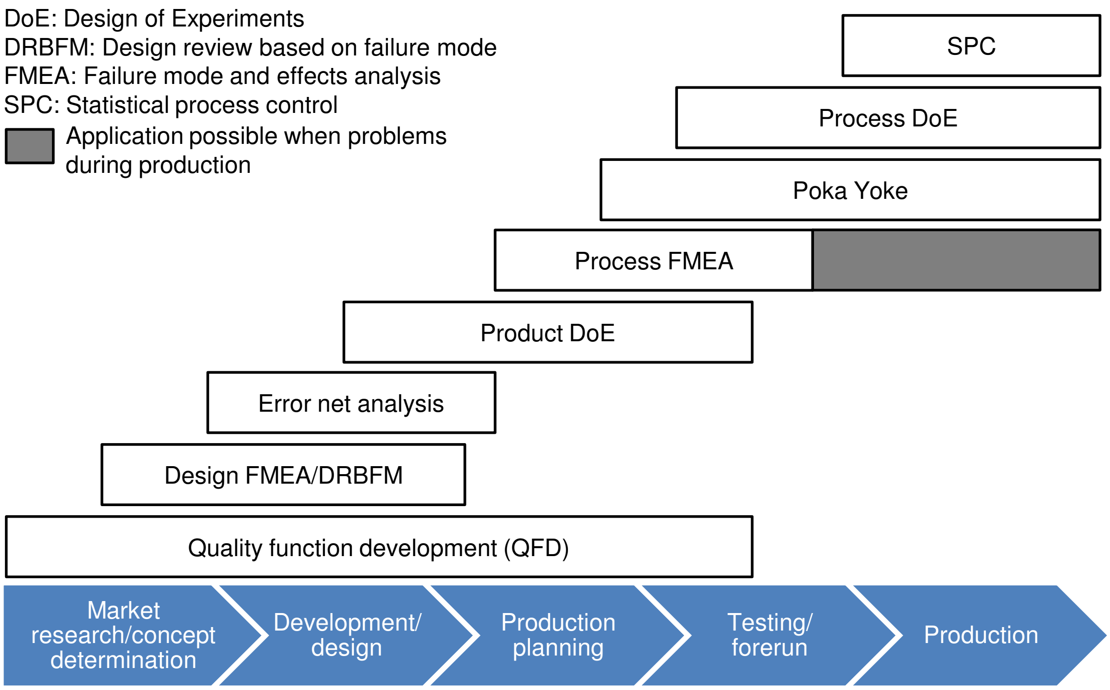

Figure 3-3: Introduction of common analysis methods during PLC, c.f. [17]

3.1.5.1 Using development data

The state of the art shows various analysis tools for different applications and goal achievements in development. A market overview shows an excerpt of available tools. The following contents in this subchapter and detailed investigation of tools were carried out by [16], [57] and [137].

- KPIFire [102]:
  This analysis tool represents an online application for management using changeable precast KPIs, such as costs, revenue and profit. The three items goals, projects, and metrics are used to define processes and can be connected to each other. Projects are related to users and workflows (plan, do, check, act). Metrics also allow a definition of effort and impact. The visualisation includes defined dashboards enabling an overview.

- SimpleKPI [72]:
  This tool enables a quick and easy realisation of KPIs over time lane using dashboards as visualisation platform for different graphs. The input data follow a schema including group, user, date, actual, target, etc.
- JIRA [2]:
  This application represents a project management tool for development, especially software development. Based on a defined approach, such as SCRUM, Kanban, or basic development, it includes issues as tasks or bugs, epics, and stories (including several tasks or bugs). Basic status elements are "to do", "in progress", "in review" and "done". In addition, dashboards enable visualisations.
- Datapine [145]:
  This tool uses an online application and connection to different databases to include information data from development. A flexible system allows a definition of approaches and metrics. Additionally, SQL queries can be introduced to generate KPIs visualised by different charts.
- Sisense [170]:
  This tool is an offline solution which introduces data from various databases to its own interface. Data can be connected and dashboards including different charts are available.
- Rally [19]:
  This tool allows different possibilities to investigate the development. Items are user stories (status idea, defined, in progress, completed and accepted), tasks, deflects, and test cases. These items are related to a release point including iterations. Dashboards are available.
- InetSoft [75]:
  After PC installation, this tool enables connection to databases. Visualisation of this data by dynamic dashboards and changeable user interfaces including filter function is available.
- Target Process 3 [179]:
  This tool uses the items "epic", "feature", "story", and "task" to define a project. The declaration of workflows, bugs, and test cases as well as time lane definition by releases including sprints is possible. Analysis methods are generally like flow diagrams or burn down diagrams.
- Wrike [194]:
  This tool uses the items "folder", "tasks", and "subtasks" to define a project including durations and due dates. Gannt charts are used to visualise the status of tasks. Approaches can be individualised.

- Kiuwan [97]:
  This tool is a static code analysis including a selection of different code evaluation methods. An action allows planning of tasks. This tool delivers code metrics and shows the status of software by the categories maintainability, efficiency, security, reliability, and portability.
- Imagix 4D [74]:
  This tool is a static code analysis including the definition of software architecture. Metrics and connections can be visualised. Changes in code and fulfilment can be evaluated.
- SQuORE Automotive [174]:
  This tool represents a static code analysis including different software standards. The specification of milestones, open issues, and test cases enable planning and controlling functions. The categories “community adoption”, “code quality”, and “team activity” show the status of software development by using different metrics.
- Axivion Bauhaus Suite [4]:
  This tool includes various software standards and enables a software structure and code metrics analysis.
- Software Diagnostics [171]:
  Due to structure and code investigation, a static and dynamic code analysis is possible.

3.1.5.2 Using field data

Field data represent a huge potential for companies to get information of product behaviour, as shown in Figure 3-4. These data enable an enhancement of knowhow, quality management, and reliability of further product generations. Hence, the feedback of this field delivers information of product behaviour for development processes.

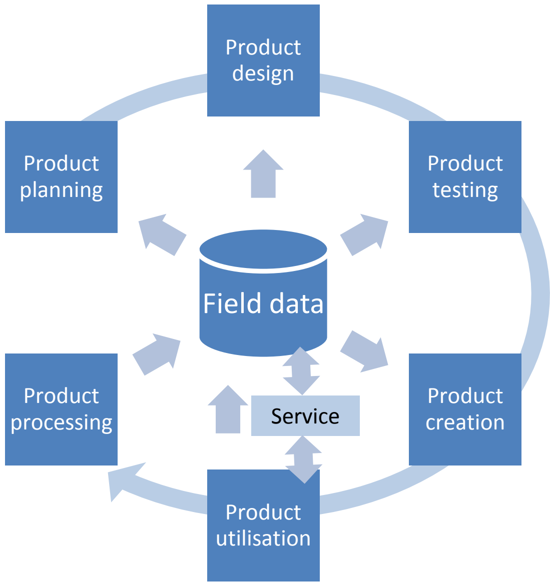

Figure 3-4: Extraction and use of field data, c.f. [39], [43]

Service, customer, and automatic detection devices are three main categories of field data sources, as shown in Figure 3-5, [43], [160].

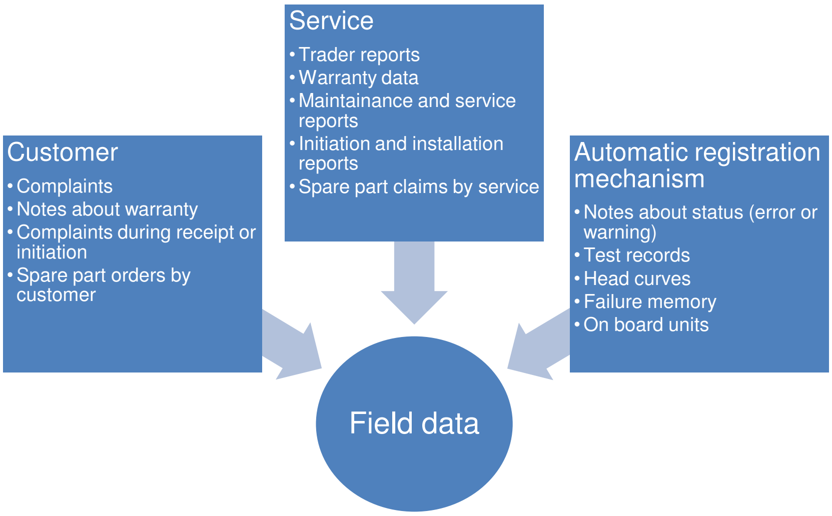

Figure 3-5: Categories of field data sources, c.f. [43], [160]

Determining field data can be done by the following sources, [39], [43], [160]:

- Service:
  Service inspections provide information due to malfunctions or faults in field use. Considering the quality and reliability of products, malfunctions and their data are interesting. For tracking information and measurements, normal service data can include important information. Systems with failures often induce complaint investigations and guarantee processes. Delay and falsification can influence this data.

- Customer:
  Data from customer occur in form of feedback, interviews, or questioning and usually represent qualitative information. The direct use or introduction in development can be inefficient. However, this type of data is also very important for a company to improve products and customer satisfaction. For instance, customer needs and requirements are useful inputs for fault fixing, product planning, and development.
- Automatic detection devices:
  Automatic detection or recording systems come from on board and off board diagnostics, such as sensors, ECUs, tachographs, or other measurement systems. This type of data delivers quantitative parameters which are normally not influenced by delay and falsification. OEMs and their suppliers need these data to enhance quality and processes. Hence, most of them use systems for their service to save and obtain information, such as engineering data management systems (EDM), quality information management systems (QIM), complaint management systems and quality function deployment systems (QFD).

Furthermore, recycling represents a possibility to generate information by analysing parts, components, or systems. An efficient complaint investigation system supports the recycling process and generates useful data, [39], [43].
Reliability investigation on basis of field data is a complex and not fully researched area. Nevertheless, the state of the art shows different approaches to enable feedback of field data. There are some studies which try to feed back field data, such as “Section of Safety Theory and Traffic Engineering” from the University of Wuppertal Study for Automotive Industry [166] or Bertsche et al. in [10] and [11]. The study from the University of Wuppertal deals with warranty and field data to enable lifetime distribution models. Field data represent information about the behaviour of systems due to real conditions and stress factors. During the warranty period various data are available but limited in time (type I censorship). To enable a satisfying statement on basis of field data, various information is required, such as, [43], [117], [132], [157]:

- Production date
- Registration date (considering registration delay)
- Breakdown date
- Registration date in database (considering report delay)
- Mileage at failure
- Amount of production
- Sold products

- Component number
- Failure Code
- Sub market factor (considering sales areas)
- Recall quote (considering relation of OEM and suppliers)
- VIN (Vehicle Identification Number) including World Manufacturer Identification (WMI), Vehicle Description Section (VDS, including production series, motor type) and Vehicle Indicator Section (VIS, including model year code, manufacturing plant and a continual number), [53]

Obtaining information on basis of field data is always combined with the risk of incorrect or inexact information, such as missing data or wrong information. There are various reasons for this problem in data quality. One reason is that service staff fills in wrong data which have to be considered by making statements on basis of these data. Controlling and defining approaches by generating or collecting this data are required but cannot guarantee a perfect condition of data quality. The combination of various databases with information from different parties enables an optimised field data information. This can be done by connecting all of the available information in a relational data storage system, [43], [157]. The so-called diagnostic trouble codes (DTCs) in combination with the diagnostic specification give the possibility to analyse the cause of faults. Depending on the failure type and failure cause knowledge further information, such as fault protocols, expert knowledge, or development information, can be required to enable cause investigation. The Wuppertal study includes three types, [43], [118], [132], [157]:

- Driving performance distribution:
  To model driving performance distribution on basis of field data a log-normal function or Weibull distribution is used over a time period. The aim is to calculate driving performances of different vehicle types. The maximum-likelihood method enables the parameter estimation.
- Km-related reliability parameter:
  Generating km-related reliability parameters based on warranty data. For this purpose, the mileage of breakdowns in warranty phases is reported representing a shortened population. The Weibull distribution is used to predict future breakdown behaviour.
- Time-related reliability parameter:
  Some industrial applications prefer statements of time-related behaviour. The combination of driving performance distributions and km-related prediction delivers a statement of time-related behaviour.

In [10] and [11] Bertsche et al. uses different qualitative and quantitative approaches to describe reliability of mechatronic systems. Schmitt and Peifer show in [160] a common approach for processing and using field data (see Figure 3-6).

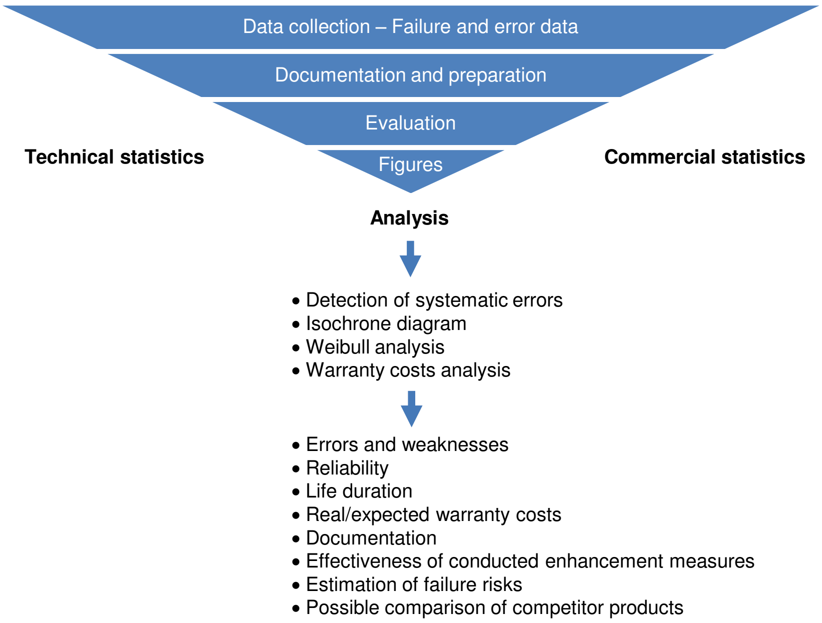

Figure 3-6: Common approach for processing and use of field data, c.f. [43], [160]

Other approaches try to enable a model-based feedback of field data by connecting process and product system models, such as Mamrot, Marchlewitz and Winzer in [112]. A few methods focus on improving the quality in development with minimum effort. Nevertheless, faults appear in field uses. The increasing complexity of products or systems with high functionality needs an evaluation of field and complaint data. There are also approaches trying to use system engineering to enable an interface for investigating field data of systems and their connection to development phases, such as Schlund and Winzer in [158]. These approaches try to represent the unstructured connection between PLC and product. Hence, the system includes elements with relations to each other and not to the environment. The system boundaries are related to the definition of the system and its subcomponents. Input and output of the whole system are defined as material, information, and energy flows. The process model describes the process system including work, equipment, and actors as well as their connection. For instance, a production system can be defined as work or equipment. Demand compliant design (DeCoDe) helps to describe the product model using requirements, functions, internal processes, and components. This approach differs by event, faults, and general field data, such as environmental conditions. This supports the definition of process-related attributes, such as environment, actor or purpose, and product-related attributes like the problem function, relation, and structure-giving statements or causes. Process-related and product-related attributes can be connected to enable cause research due to field data, [112], [158], [160].

3.1.5.3 Big data

Big data include huge amounts of complex, enhanced, and changeable information with challenges for capacity of database or storage systems. This data is generated by various possibilities including all kind of data. Using big data requires approaches and methods of data processing. For companies big data enables various options to improve processes and product quality as well as precise definition of customer needs and use. To enable efficient big data management four kinds of attributes have to be controlled, [98], [186]:

- Volume:
  Increasing data flows, for instance due to social media or connected driving
- Velocity:
  Increasing importance of data processing to enable real time investigations
- Variety:
  The variety increases due to all kinds of information sources
- Value:
  New analysis methods increase value and opportunities

3.1.5.4 OpenMDM

The automotive industry needs structured and transparent generating and investigation of measurement and field data. Thus, openMDM represents a measurement data management system as open source framework including planning, test, import, publishing, retrieval, evaluation, and import of evaluation results. Companies are able to change systems due to changing requirements. OpenMDM enables better administration of measurement data by providing interfaces for different systems to use these data, such as Matlab, Diadem, and Test-Lab. The collaboration of different OEMs and their suppliers enable advantages of test and measurement data management to reduce effort and cost, [38], [104], [105].

3.1.5.5 Other data management systems

Engineering data management system
Engineering data management systems (EDM) enable an exchange of data and information in companies, primarily in development phases. Thus, enabling data information during the whole product life cycle of another target. Including field data would provide continuous improvement of products and processes, [39].

Quality management systems (see Chapter 3.1.5)
Different quality management systems take place in product development processes. The use of field data or information of the whole product development process is not solved sufficiently at this point of time, even if there are some theoretical approaches, [17].

3.2 Indicators (KPIs) definition process

To ensure efficient project and quality management using so-called key performance indicators is an important process to control and monitor the whole PLC. For this purpose, introducing analysis methods based on metrics is necessary to gain quantitative and qualitative information of the progress. Using indicators is an efficient and objective way to evaluate performance and productivity of a project or a company. While metrics are quantitative measurements of inherent attributes of a system or process, KPIs are measurements of performance in terms of aims and goals of a company. Thus, these analysis methods are delivering indicators to support an efficient project and quality management. Four types of parameter can be differed, [45], [50], [52], [73], [78], [129]:

- Critical success factors (CSFs)
- Key result indicators (KRIs)
- Performance indicators (PIs)
- Key performance indicators (KPIs)

The definition of these indicators is a process as shown in Figure 3-7. Depending on the project or company preferences, values, and targets, different indicators deliver the right information and statement to satisfy various needs. During the first step of the KPI definition process, the company has to define the mission, vision, and important values. After this goal and strategy decision, CSFs are specified. Finally, the determining significant indicators deliver an objective evaluation basis to control the PLC.

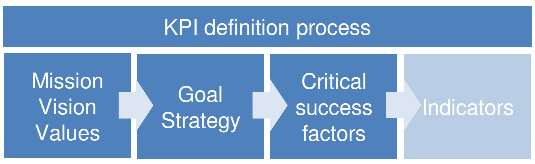

Figure 3-7: Indicator definition process, c.f. [52], [129]

There are three types of indicators: KRIs, PIs, and KPIs. Figure 3-8 shows the relationship between these three types of performance and progress metrics. These indicators are used simultaneously, but deliver different statements. KRIs represent the outside layer as so-called overall conditions. PIs are various indicators within different layers under outside skin. KPIs represent the core of the process and progress performance in the middle circle, [52], [129].

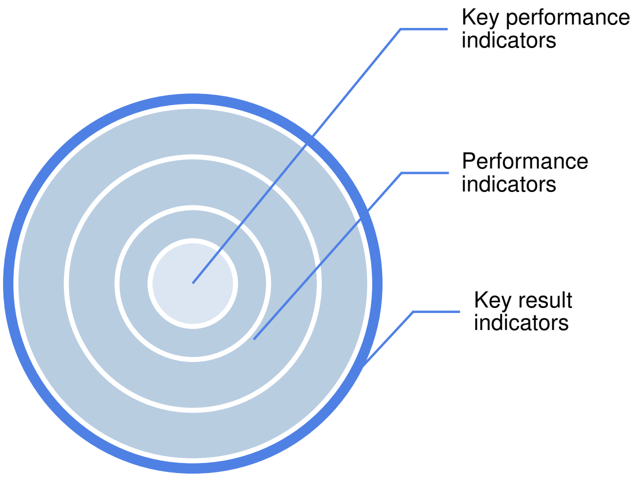

Figure 3-8: Relationship of indicators, c.f. [52], [129]

3.2.1 Critical success factors

As representatives of business success CSFs answer the question "What has to be done to be successful?" (e.g. to increase quality, to attract new customers). Hence, CSFs are crucial determinations to ensure the success of a company or project. Thus, various objectives include some basics for the definition process of CSFs to ensure an efficient and successful strategic management. For instance, defining CSFs uses practical logic, heuristics, or rule of thumb by brainstorming rather than elaborating procedure or theoretical models. Additionally, managerial experience over the years supports decision-making based on CSFs as well as the analysis process based on expert statements and past organisational success. Furthermore, CSFs enable a distinction of successful and unsuccessful companies as well as the definition of key point areas to determine objectives and devising measurements of performance. CSFs and KPIs are interacting as effects of a company's tasks and performance, [52], [96], [129].

3.2.2 Key Result Indicators (KRIs)

KRIs deliver information about how the project or progress is done in a perspective. These indicators are used to define if the process or project heads towards the right direction. KRIs enable monitoring throughout a longer time period and include monthly or quarterly information sampling, such as, [52], [129]:

- Customer satisfaction

- Profitability
- Employee satisfaction
- Net profit before tax
- Return on capital employed

3.2.3 Performance Indicators (PIs)

PIs give information about what to do and are various parameters between KRIs and KPIs, for instance, [52], [129]:

- Process time
- Ratios
- Various metrics
- Profitability of the top 10% of customers
- Net profit on key product lines
- Percentage increase in sales with top 10% of customers
- Number of employees participating in the suggestion scheme

3.2.4 Key Performance Indicators (KPIs)

KPIs deliver information about how to increase performance and are a set of parameters which represent the most important measures for the current and future success (e.g. the residual error rate or fulfilment degree of requirements). During the whole PLC, especially during development, KPIs are permanently used for analysis of the daily or weekly progress and performance. These indicators are very important measurements to avoid risks and complications at the end of development or at least at the customer stage. Nevertheless, this generated information is the basis to develop statements about necessary actions represented as KPIs. These metrics can be used to evaluate and compare performance and progress during the PLC. KPIs are objective, quantitative measurements to handle the CSFs enabling control of status, progress, and performance trends. The combination of CSFs which are requirements to reach a business target, and KPIs as quantitative measures for the achievement of long-term targets, enable an efficient company and project control. There are different options to gain indicators, such as past-, current-, or future-focused measures. The relation of the different indicators should follow the 10/80/10 rule to deliver a controllable set of KPIs (10 KRIs, up to 80 PIs, and 10 KPIs), [52] , [70], [129].

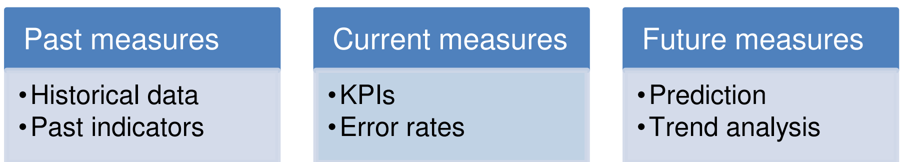

Figure 3-9: Options to gain indicators, [52] c.f. [129]

Efficient and effective analysis methods connect KPIs to business strategies and quality management. For this purpose, the following four main objectives have to be considered by determining KPIs, [52], [129]:

- Partnerships (e.g. to suppliers, customers or unions):
  This objective is important for a uniform communication and transparent decision-making. Thus, all stakeholders have to be informed and have to be considered if a change is required. In addition, the information should include a common development strategy, involvement of suppliers and customers, as well as the status of progress of implementation.
- Transfer of power to the front line:
  An important precondition for successful performance and process improvement is the empowerment of the operational front line. Effective communications, enabling KPIs handling, sensitisation for understanding of KPIs and CSFs, as well as the definition of responsibility of teams are efficient ways to transfer the power.
- Implementation and integration of reporting, measurement, and improvement of performance:
  Developing strategies to improve performance and generating performance measurement are iterative, time-depended processes. To support these approaches a consistent and continuous reporting supports efficient and focused decisions.
- Connection of KPIs to strategy:
  KPIs should be linked to strategy following the process of indicator definition, as shown in Figure 3-7. This connection of indicators to the mission, vision, and values of a company or project deliver efficient measurement or evaluation possibilities.

Management of a company has to ensure that strategies and targets are fulfilled to reach profitable results. The used KPIs to control these processes have the following characteristics, [121]:

- KPIs describe the process due to effectivity, efficiency, and quality.
- KPIs detect deviations and changes in processes.
- KPIs are measurable and objective.
- KPIs have a target value (part of the company's target).
- Reaching KPI targets equals reaching company's targets.
- The defined set of KPIs has to be consistent with the company's target.

3.3 Reliability investigation

The following content incorporates and extends the research of [25], [43] and [134]. Reliability is defined as the ability of the product or system to meet the defined requirements during a given period. It can also be interpreted as a time function of quality and can be evaluated following qualitative or quantitative analysis. Qualitative analysis defines reliability as the quality to meet requirements and deliver weaknesses of the system. Quantitative analysis delivers a statement of reliability by using survival probability, [8], [11].

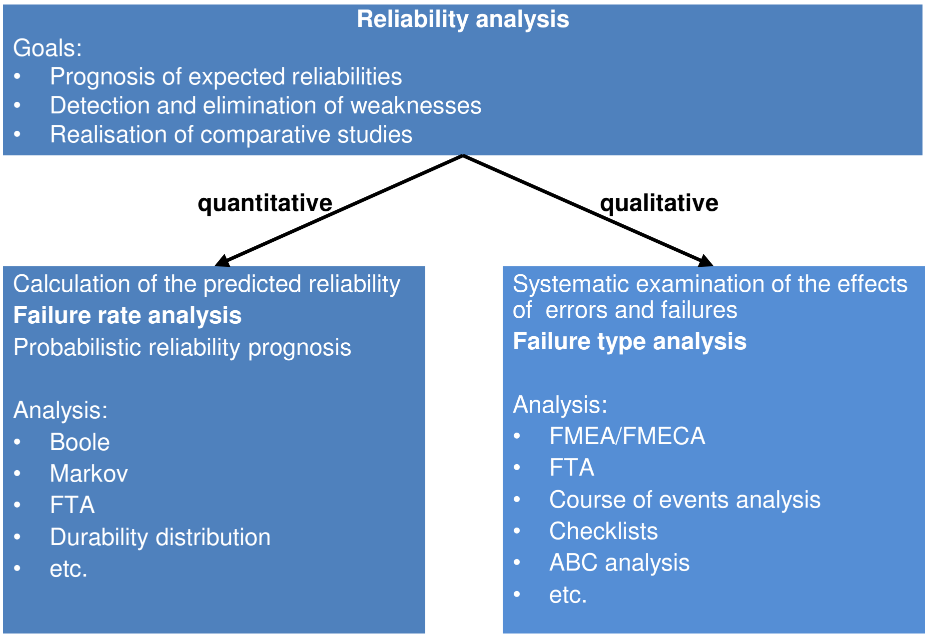

Figure 3-10: Types of reliability analysis, c.f. [11]

The area of reliability investigation includes different parameter and definitions, such as, [8], [11]:

- Parameter of functionality:
  - Fault:
    Faults are a status in which the functionality of a system is decreased or lost.
  - Defect:
    Defects occur when the system shows a deviation from requirements in at least one function.
  - Failure:
    Failures lead to a determination of correct functionality or breakdown of a system.

  - Required function:
    The requirement function represents a necessity of a system including target values and tolerances.
  - Mission profile:
    The mission profile depicts a specific task of system over specified time under predefined conditions.
  - Product lifetime:
    The product lifetime is the duration of an operating period of a system.
- Calculating values:
  - Survival probability:
    The survival probability is the likelihood that system reaches operating period.
- Parameter of status:
  - Availability:
    The availability is the probability that a system provides functionality at a certain point of time.
  - Non-Availability:
    The non-availability is the probability that a system cannot provide functionality at a certain point of time.
  - Safety:
    Safety is the absence of unacceptable risk.
  - Risk:
    Risk is a quantitative value rather than a combination of failure amount and severity of damage.
  - Deviation:
    Deviation is the difference between the investigated or measured value and the correct, specified value.
  - Danger:
    A danger occurs, if the risk is higher than acceptable risk.

3.3.1 Qualitative reliability analysis

Qualitative reliability investigations deliver a systematic evaluation of function faults and their effects on the systems. Using these methods as soon as possible in development process enables time and cost advantages. Analysis and evaluation by abstraction of both the whole system and subcomponents in detail supports an efficient reliability investigation of systems. Furthermore, incomplete and imprecise information have to be modelled ensuring consistency of PCP. To enable this analysis, different methods and development languages are used. Due to increasing complexity of systems, new enhanced methods are required, [11], [43].

3.3.1.1 Failure mode and effects analysis (FMEA)

The failure mode and effects analysis (FMEA) is a common qualitative analysis method and very often used in industry. This method enables an investigation of failure modes, their causes, and consequences. Furthermore, the FMEA delivers an evaluation of failure-related risks. It follows a five-step approach starting with a structure analysis of the system and their subsystems, resulting in a structure tree. The next step is the function analysis and description of functions as a function tree. After this, the failure analysis follows by investigation of all possible failures and their consequences resulting in a failure function structure. The following reliability investigation delivers the so-called risk priority number (RPN). The RPN is the product of the three parameters severity (S), occurrence (O), and probability of detection (P) calculated for each failure, [11], [43], [189]:

$$RPN = S * O * P \tag{3-1}$$

The RPN delivers a risk indicator which supports the decision for system optimisation or safety measures by avoiding fault and reducing risks during the final step of FMEA. Additionally, there are different types of FMEAs, such as construction FMEA (design), process FMEA (production), and system FMEA (product and functionality). Construction FMEA represents the design and layout of components to avoid constructive process failures. Process FMEA considers production-oriented dimensioning or processing to avoid planning and production failures. System FMEA delivers an investigation of function-oriented connection of system components and interfaces to avoid failures in system design and selection. Quality management uses FMEA to understand and investigate products and systems, but mainly to reduce risks in development of new products or processes, [11], [43], [47], [48] , [189].

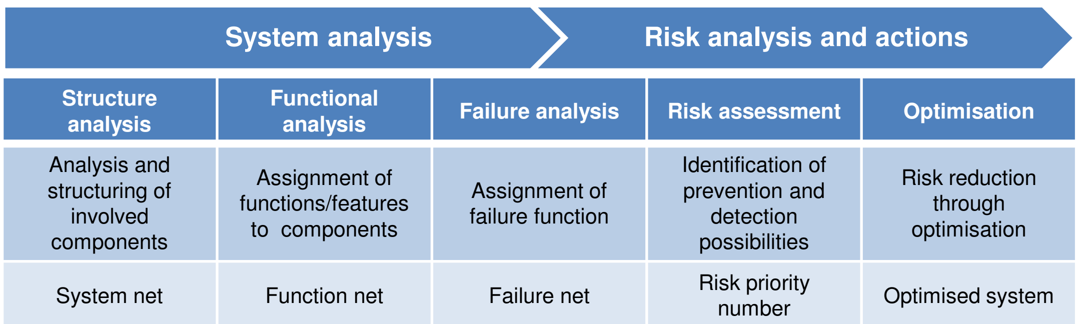

Figure 3-11: Approach of FMEA, c.f. [94], [189]

3.3.1.2 Fault tree analysis (FTA)

Fault tree analysis is a very common reliability analysis method which enables a qualitative and quantitative top down approach. FTA starts on the basis of a specific failure or breakdown, representing the primary event and evaluates different functions by use of Boolean operators to investigate their connections and the basis cause event. Thus, multiple faults can be investigated and reconstructed by this method, [11], [43], [47], [48], [189].

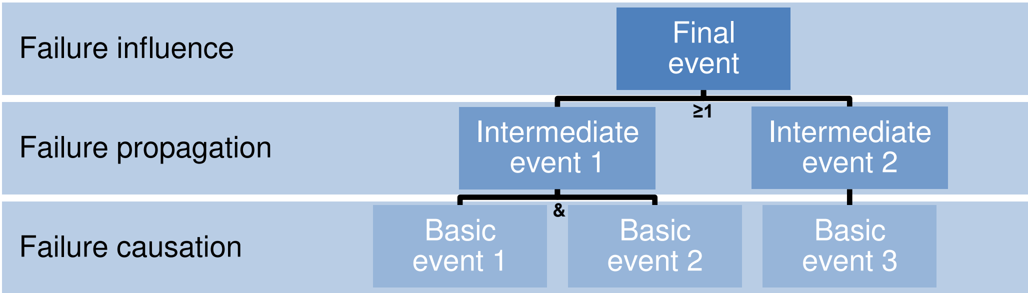

Figure 3-12: Approach of FTA, c.f. [31], [51], [94]

3.3.1.3 Situation-based qualitative modelling and analysis (SQMA)

The situation-based qualitative modelling and analysis (SQMA) is a qualitative reliability analysis method using hierarchical structures, interfaces, description of model parameters with intervals, behaviour determined by qualitative rules, as well as a computer-based connection between components and system. The approach of this analysis method starts from an overall system layer to a detailed layer to define components. Thus, the next step investigates interfaces (terminals) of communication of a component with the environment. Furthermore, quantitative variables (quantities) are used to represent physical parameters and so-called situations describe the behaviour of components using qualitative intervals. Situation rules define physical and functional relations or behaviour. Commentary rules enable classification of situation groups and parameter combinations. Transitions can also be described by the use of a matrix and rules. The next step includes the connection of components using a net list representing the whole system connected by knots and interfaces. This enables the consideration of the whole system and connection to the subcomponents by using different levels. Furthermore, this structure and connections can be visualised by a graphical representation, [11], [43], [160].

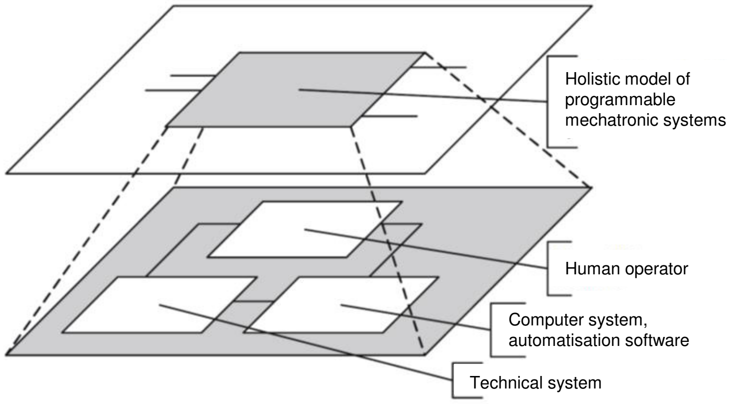

Figure 3-13: Visualisation of the system, subcomponents, and their connection using SQMA, c.f. [11], [43], [59]

3.3.2 Quantitative reliability analysis

Quantitative reliability analysis methods depend on stochastics including the sub-areas statistics and probability, and primarily the use of reliability models to describe lifetime distributions and failure rates as well as their prognosis, [11], [43].

3.3.2.1 Basics

The state of a system at a certain point of time t is described by the variable $X_{(t)}$. If this state variable is one, the system works, if it is zero, it doesn´t, [11], [43], [169]:

$$X_{(t)} \begin{cases} 1 \ldots working \\ 0 \ldots other \end{cases} \tag{3-2}$$

The time between commissioning until the first breakdown represents the lifetime of a system or a component. Therefore, the state variable can be interpreted as, [11], [43], [169]:

$$X_{(t)} \begin{cases} 1 \ldots t < T \\ 0 \ldots t \geq T \end{cases} \tag{3-3}$$

Failure probability is the distribution function of $F_{(t)}$ over lifetime T and the reliability $R_{(t)}$ (survival function $\overline{F_{(t)}}$) is calculated as its complement. The derivation of failure probability after time t is called failure density or probability density $f_{(x)}$. The expected value $E_{(t)}$ represents the mean lifetime of a system or product, [11], [43], [117], [151], [169]:

$$F_{(t)} = P(T \leq t) \tag{3-4}$$

$$R_{(t)} = \overline{F_{(t)}} = P(T > t) = 1 - F_{(t)} \tag{3-5}$$

$$F_{(t)} = \int_0^t f_{(x)} dx \tag{3-6}$$

$$\frac{dF_{(t)}}{dt} = f_{(t)} \tag{3-7}$$

$$\int_{t1}^{t2} f_{(x)} dx = F_{(t2)} - F_{(t1)} \tag{3-8}$$

$$\int_0^{\infty} f_{(x)} dx = 1 \tag{3-9}$$

$$E_{(t)} = \int_0^{\infty} x dF_{(x)} dx = \int_0^{\infty} x f_{(x)} dx = \int_0^{\infty} \overline{F_{(t)}} dt \tag{3-10}$$

The expected value depends on the type of the system. Hence, the MTTF (mean time to failure) is used for non-repairable systems, whereas MTTFF (mean time to first failure) represents the expected value for repairable systems. MTBF (mean time between failure) is the mean time between a repair until the next failure. MTTR (mean time to repair) represents the time from failure until repair. The failure or hazard rate $h_{(x)}$ depicts the limit of failure density function in a short time interval. While the failure density function is a conditional probability, the hazard function is not. The cumulated hazard rate $H_{(t)}$ represents the probability of a failure or breakdown at a certain point of time t, [11], [43], [117], [169]:

$$h_{(x)} = \lim_{\Delta x \to 0} \frac{P(x < X < x + \Delta x | X > x)}{\Delta x} = \lim_{t \to 0} \frac{1}{t} (1 - \overline{F}_{x(t)}) = \frac{f_{(x)}}{\overline{F}_{(x)}} \tag{3-11}$$

$$H_{(t)} = \int_0^t h_{(x)} dx \tag{3-12}$$

The hazard function is an important characteristic for the decision of reliability models. Systems with an increasing hazard rate, which means that the limit of failure rates in an interval increment increases, represent a growing probability of failures with the age of the system. This phase for non-repairable system is called wearing out. Shorter times between failures mean deteriorating for repairable systems. In comparison, decreasing hazard rates constitute the so-called burn-in effect for non-repairable systems, whereas repairable systems are improving, [151].

The so-called bathtub curve shows these three areas, called early failures, random failures, and wear out failures, [10], [52], [127].

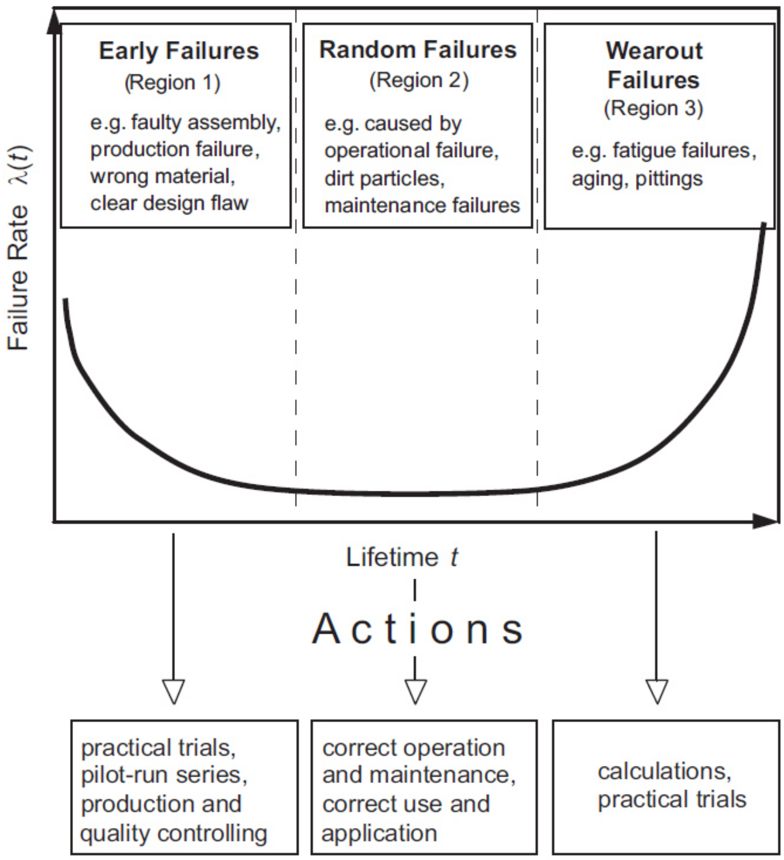

Figure 3-14: Bathtub curve, c.f. [52]

3.3.2.2 Reliability investigation of repairable systems

In case of repairable systems, an occurring failure can be fixed by repairing or changing the affected component to enable further operating conditions. Relating to behaviour, structure, functions, and characteristics of systems, different stochastic approaches deliver the correct description. For this purpose, macro models describe the outer behaviour and functions, so-called black box models (see Table 3-1), while micro models additionally consider inner structure and characteristics, so-called white box models (see Table 3-2). Macro models assume that faults from each component influence the system in a similar way and that the failure rate is equal to faults in the system. Statements due to macro models have less power and are related to the data basis quality, but there is less effort to use them. Micro models also describe the inner characteristics and need this information as data basis. Some models provide the time between failure detection (type I), others deliver the amount of failures as parameter (type II), [8], [11], [52], [151].

Table 3-1: Macro model examples, [8]

Model	Characteristics and assumption
Jelinski-Moranda-model	Assumes perfect repair, faults are random and influence reliability in similar way
Bayesian-Jelinski-Moranda-model	Improvement of Jelinski-Moranda-model using Priori-/Posteriori-reliability values
Schneidewind-model	Assumes that further test approach increases efficiency of fault detection resulting in higher importance of late faults
Geometric-model	Assumes a infinite number of failures with different probability of occurrence
Generalized Poisson-model	Failure process follows NHPP
Goel-Okumoto-model	Assumes a continuous improvement of the failure rate with increasing time
Musa-execution time-model	Describes the cumulative amount of failures using Poisson-Process and assumes exponential-distributed execution times between faults resulting in a constant hazard rate
Musa-Okumoto-model	Assumes that later fixing and corrections have less impact on failure behaviour of systems
Yamada Delayed S-Shaped-model	Considers learning process of tester
Littlewood-model	In comparison to Jelinski-Moranda-model it considers different weighting of failures and influences on reliability as well as early detection of faults and increasing time to failure
Littlewood-Nonhomogeneous-Poisson-Process-model	Assumes a continuous failure rate in comparison to Littlewood-model
Littlewood-Verallmodel	Acts on the assumption of random failure rate and uncertainty of failure weighting or correction
Duane-model	Assumes an early bug fixing and changing failure rate, usually developed for burn-in testing

Table 3-2: Micro model examples, [8], [36]

Model	Characteristics and assumption
Brooks-Motley-model	Assumes testing only on MiL level using code, variables etc. to calculate the first existing faults in the system
Shooman-model	Calculates the failure rate using process times, amounts, and probability of failures
Halstead-model	Uses different operators and metrics of code to calculate failures in system at the beginning
McCall-Richards-Walters-model	Reliability investigation due to metrics
Rudner-model	Injects the system to calculate first fault rate by division of injected faults to real faults

Point processes enable stochastic models to describe failure distributions over time. These failure points for repairable systems are independent and not identically distributed over time, such as time points T_1, T_2 . If $N_{(t)}$ represents the number of failures within the time interval $(0, t]$, the distribution shows a stochastic point process $N_{(t)}; t > 0$. The mean function of this point process is continuous, [116], [151]:

$$\Lambda_{(t)} = E(N_{(t)}) \tag{3-13}$$

The rate of occurrence of failures (ROCOF) represents the instantaneous fault rate, [151]:

$$\mu_{(t)} = \frac{\mathrm{d}}{dt}\Lambda_{(t)} \tag{3-14}$$

Various point processes differ in the increasing characteristics of events. The so-called self-exciting point process as special case delivers a hazard rate $Z(t, N_{(t)}, D_{(t)})$ which is more related to a certain point of time t, previous failure events $N_{(t)}$, and the amount of failure points $D_{(t)} = \{T_1, T_2, \dots, T_{N(t)}\}$ rather than to a constant parameter. The expected value of the hazard rate without previous event knowledge, the so-called failure rate, is defined as, [22], [61], [144]:

$$\mu_{(t)} = E[Z(t, N_{(t)}, D_{(t)})] \tag{3-15}$$

Due to these various factors and different characteristics, e.g. process status, previous history, or failure intensity, a further categorisation of reliability models enables the following classification, [11], [43], [52], [116], [151], [169]:

- Markov process models:
  The Markov process models go on the assumption that appeared faults are immediately fixed, a so-called perfect repair. Furthermore, these models assume that each fault appears only one time and cannot influence the further process. Perfect repair supposes that the hazard rate is related to previous failure rates but not their amount:

$$Z(t, N_{(t)}, D_{(t)}) = z(t, N_{(t)}) \tag{3-16}$$

  Models that depend only on the actual status or only on time t are subcategories of Markov process models. Binominal models represent a special subcategory which describes the characteristic of the tested system including a fixed number of faults with similar criticality, resulting in a similar time depended failure probability:

$$F_a(t) = 1 - e^{\left(-\int_0^t z_a(y)dy\right)} \tag{3-17}$$

Examples for this category are for instance the Jelinski-Moranda-model, the Moranda-model, and the Littlewood-model.

- Semi-Markov process models:
  Semi-Markov process models depend not only on the actual status and time, but also on the last point of failure $T_{N(t)}$:

$$Z\left(t, N_{(t)}, D_{(t)}\right) = Z\left(t - T_{N(t)}, N_{(t)}\right) \tag{3-18}$$

  T_0 equals zero representing the start of testing. The information of status at time points $T_0, T_1, T_2, \ldots$ is the only previous information for a further stochastic point process. Some of these models also consider the possibility of further faults due to bug fixing. One example is the Littlewood-Verrall-model.
- Non homogenous poisson process models (NHPP):
  Non homogenous poisson process models only depend on time and are not related to past events:

$$Z\left(t, N_{(t)}, D_{(t)}\right) = Z(t) \tag{3-19}$$

  These models represent a special case of Markov processes without the influence of status on the further process. These models consider repairing in form of minimal repair. This means that appearing faults are not immediately fixed and further faults due to bug fixing are considered. Examples are the Musa-Okumoto-model, the delayed s-shaped-model, and the power law model.
- Regression models:
  Regression models deliver a basis for lifetime analysis and reliability investigation of repairable systems. These models are related to real data and consider information loss due to censor or cuts. Co-variables consider characteristics of systems such as variants and loading levels. These parameters influence the lifetime with different intensity. Examples are the Cox-model and the Aalen-model.

3.3.2.3 Reliability investigation of non-repairable systems

Reliability investigations of non-repairable systems consider the product lifetime until first failure which causes a breakdown (no repair). The lifetime of non-repairable systems represents a random variable. One assumption is that one failure of a system does not influence other systems and their performance. Hence, the idea is that different systems have independent lifetimes. In comparison, systems with similar production, conditions, and requirements have the same lifetime distributions. The survival probability over the systems expected life stops when the first and only failure occurs. Hence, MTTF describes the failure rate (hazard rate) which can decrease, increase, or remain constant. Decreasing failure rates (DFR) represent the lifetime distribution of systems with fewer tendencies to fail over increasing lifetime, such as electronics during the early life or burn-in period. Systems with a constant failure rate (CFR) usually have loads at a constant average rate, mostly in design specifications or strength. Normally, faults with a constant rate are externally caused. An increasing failure rate (IFR) represents material fatigue, wear, or strength deterioration. Faults within this phase have an increasing probability of occurrence due to aging or cyclic loading, such as electrical or mechanical components (see Figure 3-14). Some of the most important distribution methods for modelling those characteristics are, [92], [144], [151]:

- Normal distribution:
  The normal distribution is represented by the following density function. If the mean value $\mu = 0$ and the standard deviation $\sigma = 1$, it is a standard normal distribution, otherwise there is a skew function. Usually, it is used to investigate random samples. For lifetime distribution or reliability investigations this method is not sufficient, because the variable can be negative with positive probability, [11], [43], [169]:

$$F_{(t)} = \frac{1}{\sigma * \sqrt{2 * \pi}} \int_{-\infty}^{x} e^{-\frac{1}{2}*(\frac{t-\mu}{\sigma})^2} \, dt \qquad (3\text{-}20)$$

- Log Normal distribution:
  Another form of normal distribution is the lognormal distribution which is used for lifetime investigations. This method represents an increasing failure rate until a maximum with further reduction to zero along the time lane. This characteristic is suitable for fatigue breaks or distribution of mileage and often used in mechanical engineering, [11], [43] , [118], [169]:

$$F_{(t)} = \frac{1}{\sigma * \sqrt{2 * \pi}} \int_{-\infty}^{x} \frac{1}{t} e^{-\frac{1}{2}*(\frac{\ln(t)-\mu}{\sigma})^2} \, dt \qquad (3\text{-}21)$$

- Exponential distribution:
  Due to a negative argument, the function of density of exponential distribution decreases monotonically following the exponential function. This

lifetime distribution can be used for components with little aging or wearing, such as electronical parts, [11], [43], [169]:

$$F_{(t)} = 1 - e^{-\lambda t} \tag{3-22}$$

- Weibull distribution:
  The Weibull distribution is a special form of exponential distribution. The characteristics of this allocation can be changed by the parameters $a > 0$ and $b > 0$. Parameter a represents a scale parameter and enables a dilation or compression of the time axis. As a form parameter, b influences the form of the function, [11], [43], [169]:

$$F_{(t)} = 1 - e^{-\left(\frac{t}{a}\right)^{b}} \tag{3-23}$$

  b < 1 Early failures, b = 1 Random failures, b > 1 Wear out failures
- Others: Depending on the characteristics various other methods are available like Gamma distribution, Birnbaum-Saunders distribution, or Inverse Gaussian distribution.

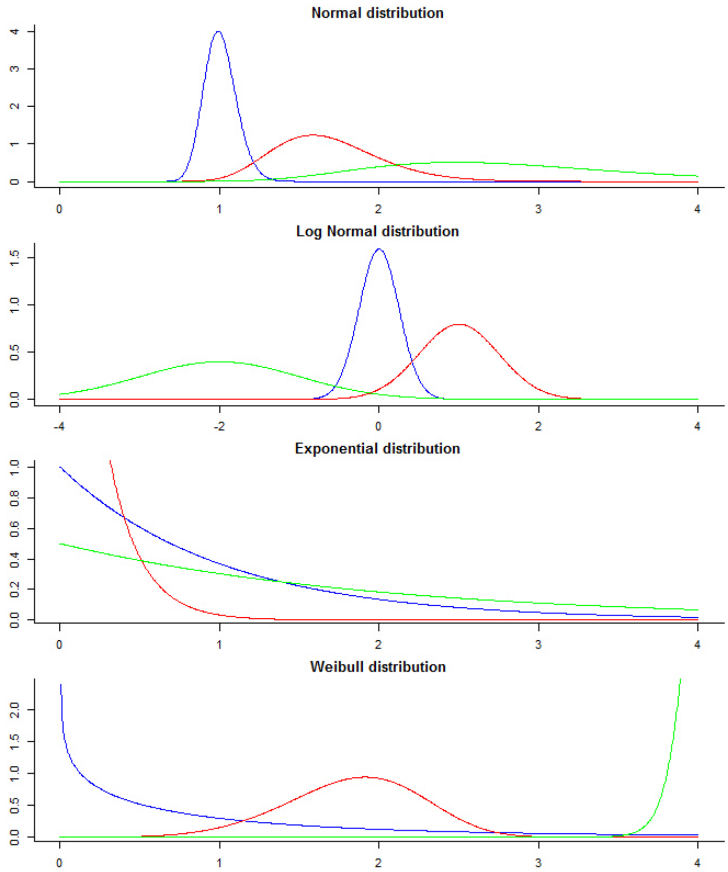

Figure 3-15: Examples for Normal, Log Normal Exponential, and Weibull distributions

3.3.2.4 Regression analysis

Regression analysis is a statistical method to define relationships between variables. For reliability analysis this method can be used to describe the relation of a dependent variable Y and one or more independent variables x_i with unkown parameters β, as well as an error rate ε, [154]:

$$Y(x) = \sum_{i=1}^{n} \beta_i * x_i + \varepsilon \tag{3-24}$$

3.3.2.5 Parameter estimation

Parameter estimations are used to deliver parameters for distributions as exact as possible, [154].

Point estimation

If the distribution of a random variable X is known except of parameter ϑ, the estimation of this parameter is called point estimation and defined as, [120]:

$$\hat{\vartheta} = \hat{\vartheta}_{(X)} = g(X_1, X_2, \ldots, X_n) \tag{3-25}$$

Requirements of an estimator are, [120]:

- Inbiasedness:
  Due to the unknown parameter ϑ in mean value, an acceptable value for $\hat{\vartheta}$ has to be delivered. $\hat{\vartheta}$ is inbiased for ϑ if:

$$E_{\vartheta}\left(\hat{\vartheta}(X)\right) = \vartheta \ldots for\ all \text{ allowable } values\ of\ \vartheta \tag{3-26}$$

- Consistency, [120]:
  If an increasing random sample delivers a variance tendency against zero, consistency is enabled fulfilling the law of large numbers. $\hat{\vartheta}$ is consistent for ϑ, if:

$$\hat{\vartheta}_N(X_1, X_2, \ldots, X_n) \to \vartheta \ldots for\ N \to \infty, for\ all \text{ allowable } values\ of\ \vartheta \tag{3-27}$$

Maximum-likelihood method (MLM)

The maximum-likelihood method is used to generate unknown parameters of a given distribution by estimation. The principle of MLM relies on finding a parameter constellation with the highest probability. N independent and similar distributed random variables $X_1, \ldots, X_n$ and their realisations $x_1, \ldots, x_n$ are given, with ϑ as the unknown parameter. If $f(x|\vartheta)$ is the density of probability function, the Likelihood function of parameter ϑ is the combined probability of occurrence of samples, [154]:

$$L(\vartheta) = f(x_1|\vartheta) * f(x_2|\vartheta) * \ldots * f(x_n|\vartheta) \tag{3-28}$$

The maximum-likelihood method maximises the Likelihood function to get the highest value, the so-called Maximum-estimator. As a strictly monotonically increasing function, the maxima of $L(\vartheta) = \ln L\,(\vartheta)$ are equal. Maximising delivers the estimator $\hat{\vartheta}$, [154]:

$$\ln L\,(\vartheta) = \sum_{i=1}^{n} \ln f(x_i) \tag{3-29}$$

$$\frac{d \ln L(\vartheta)}{d\vartheta} = \sum_{i=1}^{n} \frac{d \ln f(x_i|\vartheta)}{d\vartheta} = 0 \tag{3-30}$$

$\hat{\vartheta}_{(X)}$ is the maximum-likelihood estimation for ϑ if, [154]:

$$\begin{array}{l} \mathrm{P}_{\vartheta}(X_1 = x_1, \ldots, X_n = x_N) \text{ (discret case)} \\ \text{or } \mathrm{f}_{\vartheta, X_1,, \ldots, X_n = x_N}(x_1, \ldots, x_N) \text{ (stetic case)} \\ \textit{is maximal} \end{array} \tag{3-31}$$

Maximum-likelihood estimators normally are consistent but not always unbiased, [120].

Ordinary least square method (OLS)

The parameter estimation using least squares assumes the estimation of parameter β with a linear regression function Y_i by minimising the sum of distance squares, [66], [106]:

$$Y_i = \beta_0 + \beta_i * x_i + \varepsilon \tag{3-32}$$

$$\sum_{i=1}^{n} \hat{\varepsilon}_i^{\,2} = \sum_{i=1}^{n} [Y_i - \hat{Y}_i]^2 \rightarrow min! \tag{3-33}$$

Bayesian estimation

Bayes estimation can be used to generate unknown parameters of a given distribution using conditional probability and past events. This method enables a conclusion from effect A to cause B_i. The so-called a-priori probability $P(B_i)$ has no past information of A. The so-called a-posteriori probability $P(B_i|A)$ estimates the probability of B_i after information of A, [120], [169]:

$$P(B_i|A) = \frac{P(A|B_i) * P(B_i)}{\sum_{i=1}^{n} P(A|B_i) * P(B_i)} = \frac{P(A|B_i) * P(B_i)}{P(A)} \quad , i = 1, \dots, n \tag{3-34}$$

Interval estimation

The distribution of an estimation function of a distribution of a random variable can also be random. For this purpose, intervals are defined to enable a range of this variable with a given probability, [120]:

$$P(\hat{\theta}_n^1 \leq \vartheta \leq \hat{\theta}_n^2) = 1 - \propto \tag{3-35}$$

The confidence interval or confidence estimator is $[\hat{\theta}_n^1, \hat{\theta}_n^2]$ of the realisations $[\hat{\vartheta}_n^1, \hat{\vartheta}_n^2]$, [120].

4 Methodology

4.1 Introducing analysis methods to deliver KPIs during the different phases of PLC

Permanently increasing connectivity and complexity of products or systems are a huge challenge for industrial quality management. Only an efficient and effective control of development and field use provides high quality as well as reliability of products. The new methodology of this PhD thesis to enhance quality management and to reduce failures of mechatronic systems in field use is based on the development of KPIs delivered by new analysis methods during all product life phases (see Figure 4-1). Thus, developing and introducing new analysis methods during the PLC considers requirements, development, and field phases. This methodology enables an integrated approach for both optimised, reliable, and predictable development processes as well as detection, evaluation, and prognosis of faults in field use. Following the KPI definition process, the company has to declare relevant parameters during the whole PLC. The development of new, complex analysis methods based on a structured database is necessary to generate these indicators. The definition of relevant KPIs depends on the goals of a company, characteristics of the product, processes, and available data. Figure 4-1 shows a general concept overview of the methodology for mechatronic systems.

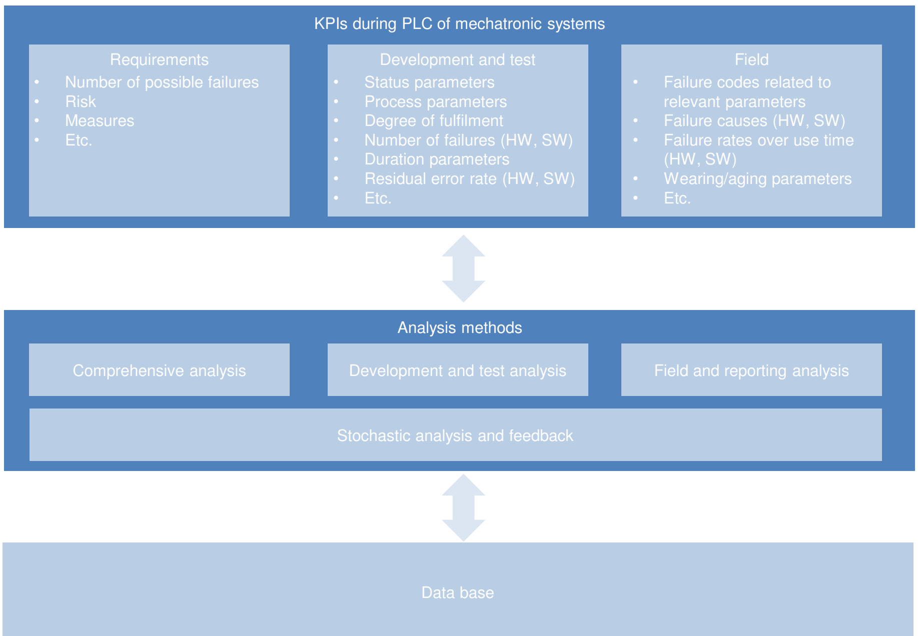

Figure 4-1: Methodology of introducing analysis methods to deliver KPIs during the PLC

4.2 Comprehensive analysis method in the requirement phase

The following content incorporates and extends the research of [25], [42], [47], [48], [51], and [26].

4.2.1 Theoretical introduction

Today's industrial trends show an increasing application of mechatronic or complex systems, e.g. electrification of powertrains in automotive industry or interaction in aerospace industry. The increasing complexity and connection of systems, various amounts of characteristics, and increasing requirements due to quality and safety standards such as ISO 26262:2011 („Road vehicles – Functional Safety“) [85], require innovative analysis methods to enable an exact and efficient evaluation of systems. Conventional analysis methods hold huge publicity and acceptance in various industries, but these challenging requirements and industrial development trends are pushing conventional analysis methods to their limits. Modern products need enhanced development processes using new analysis methods to investigate behaviour and malfunctions of components. The aim is to detect faults or danger situations and enable preventive measures in the requirement phase for further development to optimise systems, [11], [34], [47], [48], [51], [68].

4.2.2 State of the art and limits

The state of the art of evaluation methods in the phases of requirement analysis and design are failure mode and effects analysis (FMEA) or fault tree analysis (FTA). While FMEA enables determining all possible failure types, their causes and consequences resulting in a risk investigation, FTA allows the investigation of an individual failure and its cause by using Boolean operators (c.f. chapter 3.3.1.1 and chapter 3.3.1.2). The limits of conventional analysis methods depend on the complexity of the analysed systems. Conventional analysis methods are an effective way to analyse simple systems. The more complicated components and their connections are, the less these methods deliver efficient investigation possibilities, particularly when lacking the representation of multiple faults, connections, interfaces, and the evaluation of various flows such as data-, signal-, or fault-flows. In industry, combining different conventional methods is a way to face this problem resulting in insufficient redundancies and additional effort. Enhancing these methods is essential for the safety of humans and machines. The introduction of a new, innovative method is necessary and represents an influencing change in industry and processes. For this purpose, the focus of the following comprehensive method is on the realisation regarding a holistic tool, [47], [48], [51], [189].

4.2.3 Method

The comprehensive analysis method during the requirement phase is based on the concept of combination and enhancement of conventional analysis or modelling methods. It represents a connection and functional enhancement of FMEA, FTA, and block diagrams as used in sysML. This ensures complete new analysis possibilities considering complex functions and failure behaviour of systems. Using this method enables investigating and visualising the complete system including structure, function, and fault analysis, regardless of the degree of complexity. In comparison to standard representation of functions as a net, function block diagrams are used in modular approach using various levels with different depth of detail. This grade of detail is increasing with every lower level. Figure 4-2 shows the concept of the comprehensive analysis method on the application of a mechatronic system. The upper level represents the whole system with its environmental impacts and in-/outputs. In the next level 2, the mechatronic system is divided into processor, actuator, sensor, and basic system. Level 3 includes further details of subcomponents of the basic system, such as software, electronics/electrics, and mechanics. The signal flows and connection between all components can be visualised and investigated by a higher detail depth. At level 3, this example shows a fault propagation from SW as failure cause, E/E, and failure event at mechanics, e.g. breakdown. Due to the new approach and block diagrams, a method is enabled to represent and investigate different connections or relationships within systems, such as signal or data flows and fault propagation paths. The functions are varied, such as optimised visualisation of fault propagation paths enabling investigation of difficult connections, consideration of multiple failures, as well as detection gaps and differentiation between lateral or non-lateral faults. This comprehensive method enables potential to generate KPIs more precisely like number of possible failures, risk, diagnostic coverage and preventive measures, [47], [48], [51], [189].

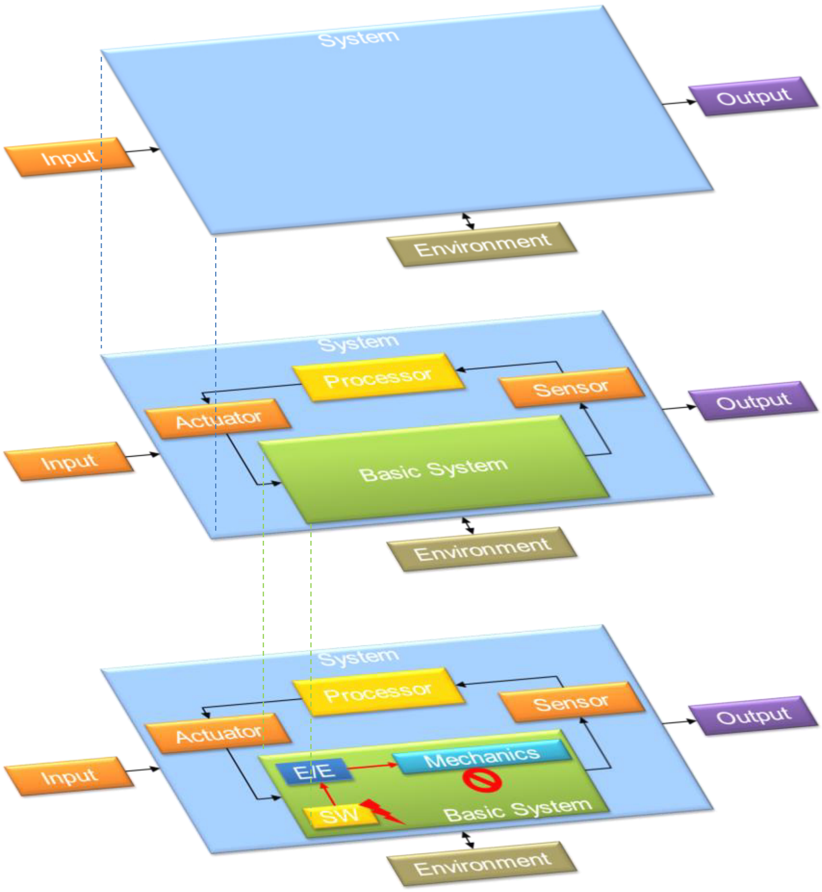

Figure 4-2: Modular concept of the comprehensive analysis method using different system levels, c.f. [47], [48], [51], [189]

The comprehensive analysis method represents an enhancement on basis of conventional evaluation tools to create an automated solution. Thus, data modelling delivers a template for programming and a data overview of the IT system. The often-used data modelling method ERM (Entity-Relationship-modelling) allows the development of a model including linking of elements (entities) and their connections (relationships), [47], [48], [51], [60]. Basic elements of this modelling language are shown in Figure 4-3.

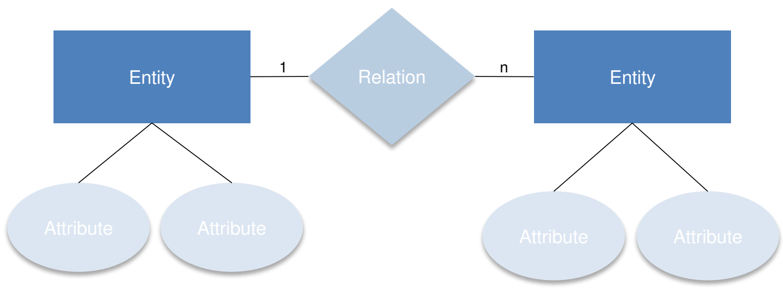

Figure 4-3: Basic elements of ERM c.f. [26], [60]

To enable a better acceptance of programming, the approach of data modelling using ERM starts with the conventional methods FMEA and FTA to enable comparability, acceptance, and feasibility, [47], [48].

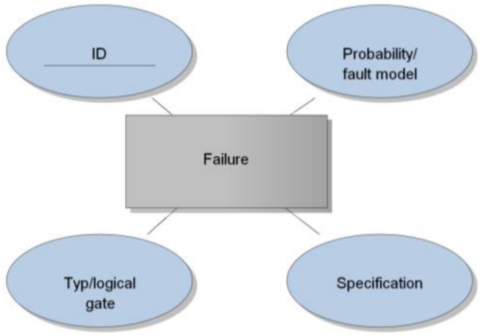

Figure 4-4: Example of FTA using ERM, [26], c.f. [25], [47], [48]

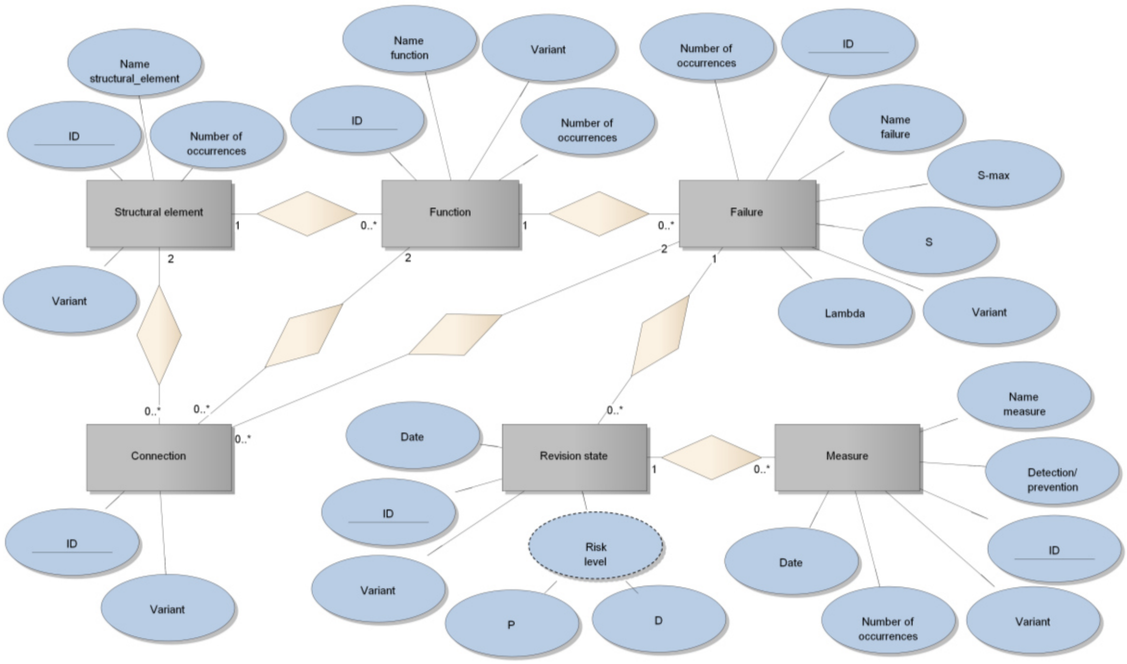

Figure 4-5: Example of FMEA using ERM, [26], c.f. [25], [47], [48]

(Further detailed information of common analysis tools and data modelling is included in [25].)

To enable better acceptance and understanding, the ERM model of FMEA serves as a basis for further development of a comprehensive method. Modelling the comprehensive analysis method is conducted in four steps, so-called expansion stages, [47], [48]:

- Expansion stage 1:
  Expansion stage 1 shows data modelling of a structure and function analysis of the standard FMEA overlaid by representation and connection pos-

sibilities. This is enabled by a modular structure with different levels. Furthermore, the connection itself and the linked ports as interfaces are shown as entities and represent the central main elements. Thus, this stage is an enhancement of FMEA by the sysML concept. For this purpose, new attributes are added to the ERM, as shown in Figure 4-7 (orange).

- Expansion stage 2:
  Expansion stage 2 enables the function of fault propagation paths by the functional structure. Besides the conventional direct connection of faults, it supports visualisation and analysis of fault propagation over various function paths delivering further failures at other locations of systems. Considering the differentiation of latent and non-latent failure types is very important for an enhanced analysis method, because latent faults do not immediately cause breakdowns of systems but can lead to critical system failures at other locations due to fault propagation (see Figure 4-6).

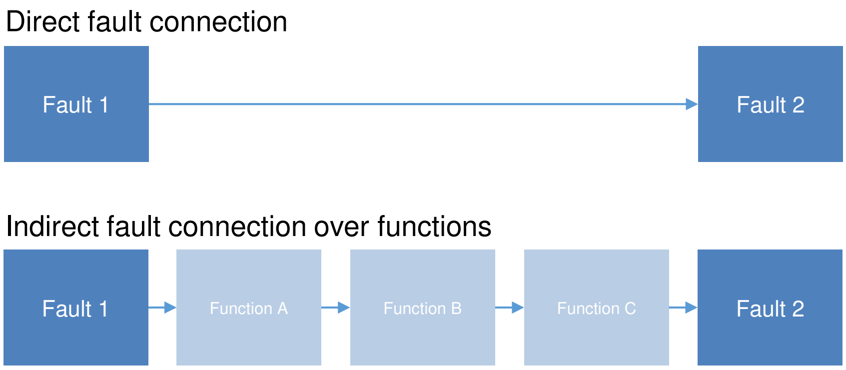

Figure 4-6: Direct fault propagation and fault propagation over functions, c.f. [47], [48]

  The developed ERM shows the implementation of fault propagation including signal and function paths, multiple fault connection due to logical gates, and investigation of failures due to latency to enable a precise analysis of the system, as shown in Figure 4-7 (green). Furthermore, the connection of fault functions by Boolean operators (logical gates) is included. This expansion stage 2 represents a combination of advantages of FMEA (function- and fault nets) and FTA (fault tree function) as well as an enhancement by additional functions.

- Expansion stage 3:
  The expansion stage 3 focuses on the implementation of so-called diagnostic coverage (DC) of safety-relevant systems. DC is a measurement for detection of dangerous or safety-critical failures. The combination of various safety systems and mechanisms can increase the detection of critical

failures, but overall DC cannot be calculated by summing individual DC values. The DC function is shown in Figure 4-7 (blue).

- Expansion stage 4:
  Today analysis methods are faced with dynamic development and processes resulting in a permanent change of systems and tasks. Hence, expansion stage 4 enables subsequent changes of structure, connections, faults and functions to be integrated in the system and development, as shown in Figure 4-7 (violet). This represents an enhancement of FMEA which considers changes only as measures and not as new elements like the comprehensive analysis method does.

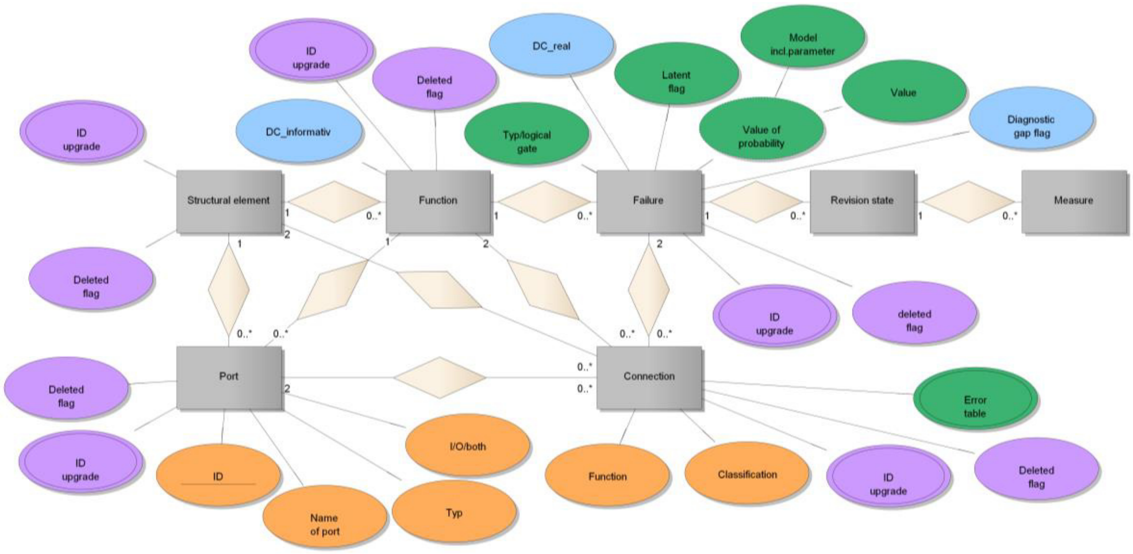

Figure 4-7: ERM of expansion stage 1-4, [26], c.f. [25], [47], [48]

Additional advantages are new possibilities of visualisation due to the modular structure and enhanced connection options of the comprehensive analysis method. Figure 4-8 shows an example of visualisation that enables the combined analysis of possible faults and fault net, fault paths over functions using Boolean algebra, safety functions, DC parameter of sensors, and safety mechanisms. Thus, these new visualisations are enabled by the new approach and do not need further effort, [47], [48].

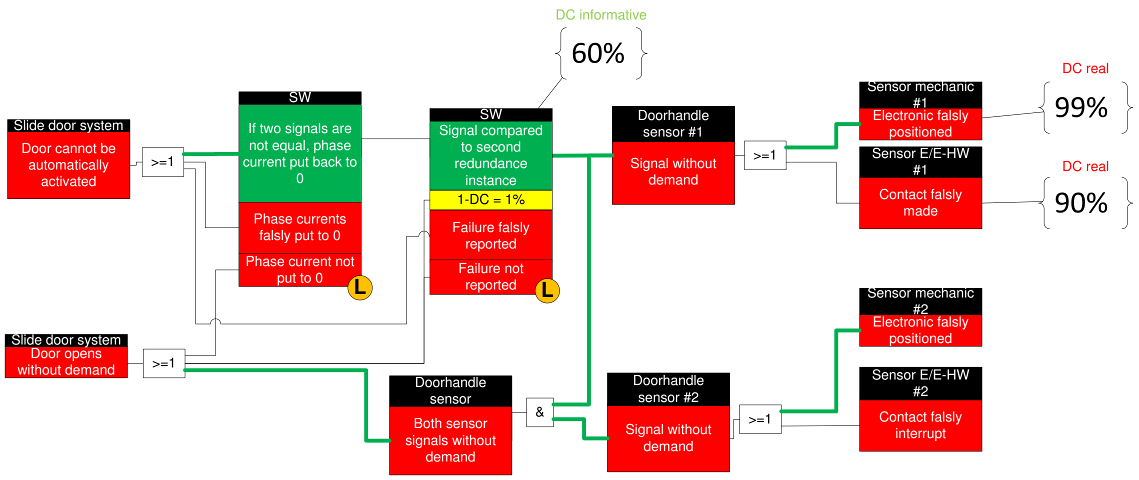

Figure 4-8: Enhanced visualisation of fault net and safety functions, c.f. [48], [161]

Additionally, further research investigations on this topic include a qualitative comparison considering data storage and possibility of evaluation of the new analysis method with common tools. Thus, decreasing data storage demands due to enhancement of features and the significantly increasing possibility of new visualisations confirm the realisation of comprehensive analysis method, see [47], [48], and [25]. Due to the target positioned new features and functions, this comprehensive analysis method can be realised on basis of the conventional tools enabling completely new analysis possibilities. Furthermore, additional research in [42] shows an attractive market potential and development information confirming this new analysis method.

4.3 Analysis methods using KPIs on basis of development and test data

The following content incorporates and extends the research of [30], [41], [45], [57], [177] and [193].

4.3.1 Theoretical introduction

Current industrial trends and the increasing challenges for development require new analysis methods including quantitative measurements to evaluate processes as well as progresses. For this purpose, the next step of the methodology introduces KPIs by development of new specific analysis methods to enable objective quality evaluations as soon as possible during product development processes. The aim is to avoid critical safety danger of humans and machines as well as warranty due to product recalls which are typically associated with high costs.

Furthermore, the support of risk-analysis for the release process is a very important issue. Hence, stochastic analysis methods are an efficient way to gain an estimation of product reliability. Modelling of the expected residual error rate based on development and test data provides basic information for decision-making during development, e.g. if further development and tests are necessary or if systems are already releasable. Quality improvement in industrial development requires various KPIs by using specific analysis methods due to different questions and perspectives of various involved parties in development, e.g. quality assurance, project management, process management, and general management. The challenge is to meet all needs by considering all of these perspectives, [45], [49], [52].

4.3.2 State of the art and limits

The state of the art shows many available tools (see chapter 3.1.5). The application of these tools in complex projects and processes can be difficult or even impossible. The limitation of the tools derives from unstructured or inefficient databases and unspecific metrics including the danger of too ordinary or even wrong KPIs. Due to the limited queries functions and the restriction of charts ordinates, such as count, sum, or average functions, analysis methods are not completely changeable and not effective for complex statements. Some limitations restrict the use of complex approaches, database connection, and metrics, e.g. connections are limited to normal forms of tables, only data with type integer on ordinates is allowed, relative frequencies or pre-defined approaches, and metrics are difficult to calculate. Tools considering code metrics are only interesting for use in software development and even not appropriate for generating enhanced KPIs. The application of complex development processes and different questions of various parties require customised tools and databases to provide expressive KPIs giving the right answers during development processes.

4.3.3 Method

The principle of KPIs generated by new developed analysis methods on basis of development and test data is shown in Figure 4-9.

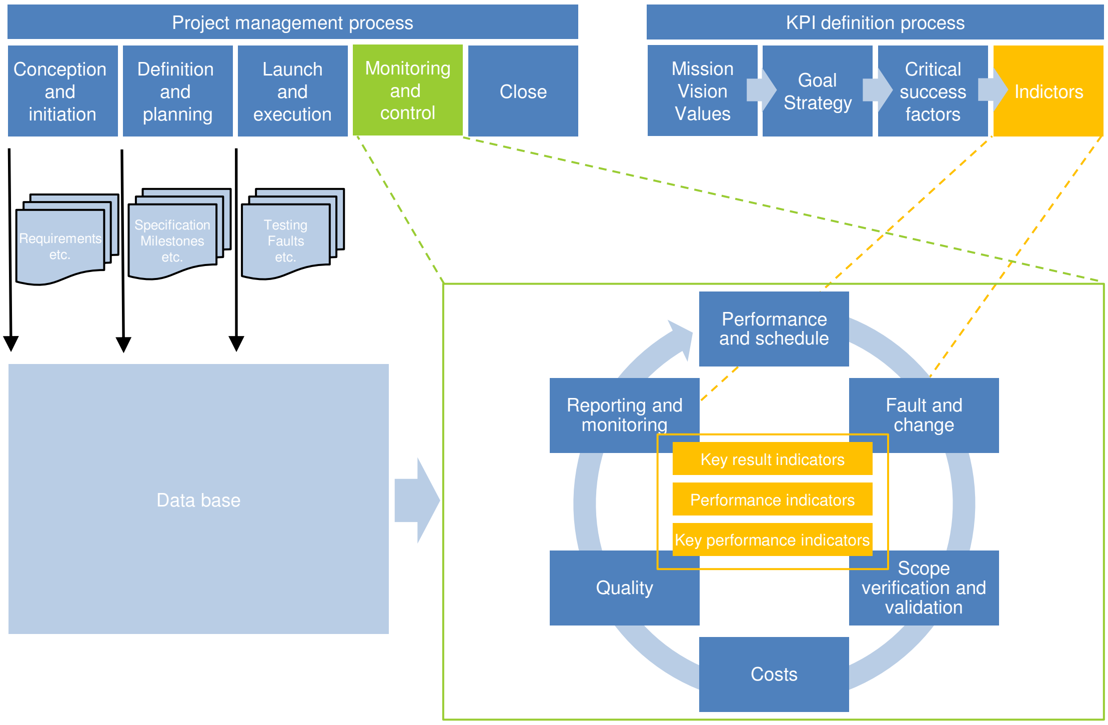

Figure 4-9: Principle approach of generating KPIs on basis of development and test data, c.f. [46]

During the whole project management process, the basis for enhanced analysis is collecting data and information, such as requirements, system and components specifications, tests, faults, and milestones. Thus, a database stores and connects development as well as test data to provide a prepared analysis platform. To enable the right statements, the KPI definition process has to be passed through considering various parties, views, and needs during development processes. Special attention has to be paid to the indicator definition for answering all questions efficiently. These questions depend on the company, product, development, and stakeholders. The introduction of KPIs by analysis method takes place in phase of monitoring, enabling several potentials such as investigations of performance and schedule, fault and change, costs, quality, as well as reporting and monitoring. This method considers different questions, perspectives, and views of the areas, e.g. general management (GM), project management (PM), quality assurance (QA), and process management (PrM). Development processes use different items to handle process and progress. Hence, requirements are usually a result of project specification serving for communication between customer and supplier. Related to these requirements so-called planning elements are used to regulate and handle internal processes or tasks. Thus, planning elements represent the operational realisations of requirements and are connected to them. Furthermore, test cases define the whole testing procedure, which are usually also connected to requirements and/or planning elements. The result of tests is faults representing deviations in process which can trigger planning elements or requirements in a dynamic development. To control the whole development process, an enhanced quality management continuously monitoring of relevant items, such as requirements, planning elements, tests, and faults has to be ensured. Due to various development phases and project maturity, specific KPIs delivered by differently developed analysis methods enable control during every step. Questions during development processes and the specific KPI-related methods can be classified by the following categories, [45], [52]:

- Status and progress:
  This analysis method enables the possibility to check the actual projects status, tasks, and progress by using descriptive statistics.
  Related question categories:
  - Control of progress and process
  - Comparison of status and project maturity
  - Evaluation of deviations, time lags, and rescheduling
  - Identification of not solved and/or critical topics

  Related KPIs:
  - Status of different items
  - Number of completed and not completed items
  - Completion status of projects and iterations
  - Number of loops
  - Amount of time lags and rescheduling
  - Number of critical items
  - Number of safety-relevant not completed items
- Durations:
  This method allows the investigation of all kind of processing durations.
  Related question categories:
  - Control and comparison in development
  - Evaluation of average durations
  - Allocation of resources
  - Definition of realistic durations by planning of new projects

  Related KPIs:
  - Project durations
  - Iteration durations
  - Fulfilment degree of freeze points

- Target-performance comparison:
  This analysis method enables a comparison of target and performance as well as an investigation of deviations in development processes.
  Related question categories:
  - Control of degree of fulfilment at release points
  - Support of risk analysis at release points

  Related KPIs:
  - Fulfilment degree of requirements at milestones
  - Fulfilment degree of tests at milestones
  - Fulfilment degree of planning elements at milestones
  - Degree of solved faults at milestones
- Tracking:
  Tracking supports control of data quality and progress management by checking target and performance as well as linkages between items on sub-component or module level.
  Related question categories:
  - Process control of subcomponents
  - Data management

  Related KPIs:
  - Fulfilment degree of subcomponents
  - Degree of connection between data
  - Degree of reviewing
- Trend analysis:
  This method supports the understanding of development processes and evaluation of tendencies of performance or progress.
  Related question categories:
  - Evaluation of performance trends
  - Checking the trend of gap between performance and target
  - Early warning system by increasing gaps
  - Analysis of implementation and test process characteristics
  - Evaluation of saturation distribution
  - Early warning system because of too low values in comparison to project maturity

  Related KPIs:
  - Gap between target and completed planning elements
  - Implemented requirements divided through all requirements
  - Completed planning elements divided through all planning elements
  - Completed tests divided through all tests
  - Number of detected faults

  - Number of solved faults
  - Fixed detected faults divided through all detected faults
  - Number of detected faults after release points
  - Unedited faults
- Fault estimation:
  Fault estimation enables evaluating detected and fixed faults over the project maturity or time lane.
  Related question categories:
  - Evaluation of faults and bug fixing
  - Identification of critical faults

  Related KPIs:
  - Gap between detected and fixed faults
  - Number of detected faults after release points
  - Unedited faults
- Prediction:
  Prediction enables the estimation of the residual error and checking the trend of faults during development. Furthermore, this method supports release decisions by delivering statements of possible faults remaining in the system at certain release points.
  Related question categories:
  - Modelling of the residual error rate
  - Early warning system for further bug fixing and increasing error rate trend
  - Statement of residual error rate supports risk analysis for release decision

  Related KPIs:
  - Residual error rate

Figure 4-10: Categories of KPI-related methods during V-model, c.f. [52]

Figure 4-10 shows the application and frequencies of the KPI-related methods during V-model. Generating KPIs during development is a continuous process. Calculating these metrics several times by looping enables efficient control and support of quality management. This approach starts at the steps status and progress, durations and tracking. Target-performance comparison allows checking the possibility of a release at a certain milestone. Fault estimation, trend analysis, and the prediction of the residual error rate usually take place in phases of tests and towards releasing the system to the customer. Potentials of KPIs in development are very variegated. Different KPIs delivered by these analysis methods enable various possibilities for quality management optimisation, [52].
The following chapters explain the analysis methods and KPIs during development in detail on basis of synthetic projects data in automotive engineering.

4.3.3.1 Status and progress

Starting the development process requires a continuously monitoring of KPIs considering status and progress of relevant items to improve quality management. For this purpose, analysis methods using descriptive statistics are developed to enable KPIs which check the actual status and progress of tasks of projects. Figure 4-11 shows KPIs as different status of all relevant parameters over development time, such as requirements, planning elements, tests, and faults. The possibility is to compare status of all relevant development items or to investigate which of them are in progress or done, [45], [50].

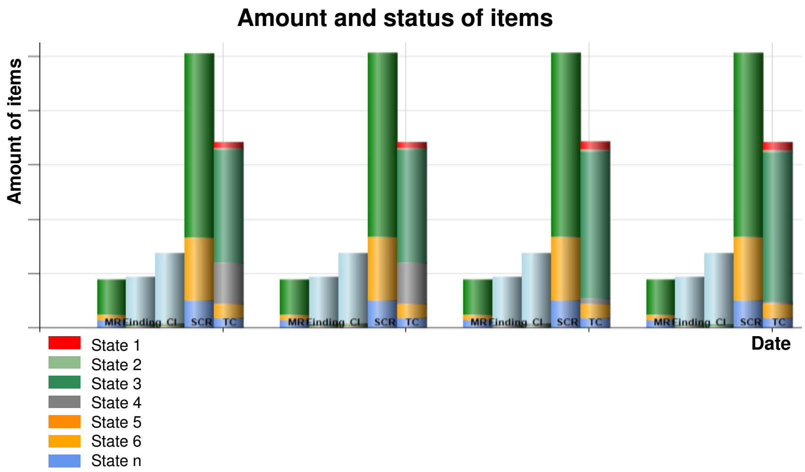

Figure 4-11: General overview of overall tasks and items, c.f. [45], [50]

The next KPIs are based on collected data considering all planning elements during development in detail, as shown in Figure 4-12 and Figure 4-13. The status and progress of these items delivers an overview and daily control of every ongoing task. Beside the number of days, which are invested in those items for different development steps, this analysis method enables KPIs such as number of planning elements that are completed in time, calculation of a possible upcoming time lag, number of rescheduling due to non-completed requirements, number of test cases, and detected faults. Statements due to these analysis methods are indicators for further risk management. Thus, special attention lies on detecting critical and not solved planning elements in comparison to the target date. These project deviations always cause further effort to meet requirements during a certain project. A set of KPIs displays all required information to check all of these impacts, [45], [52].

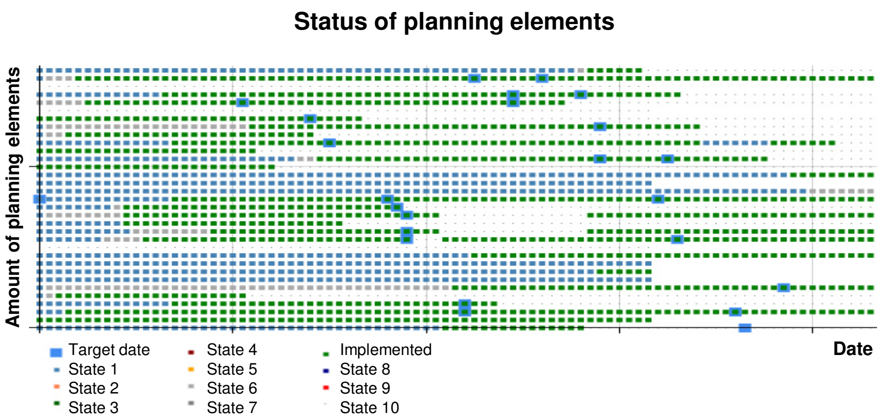

Figure 4-12. Status of planning elements starting simultaneously, [45], [52]

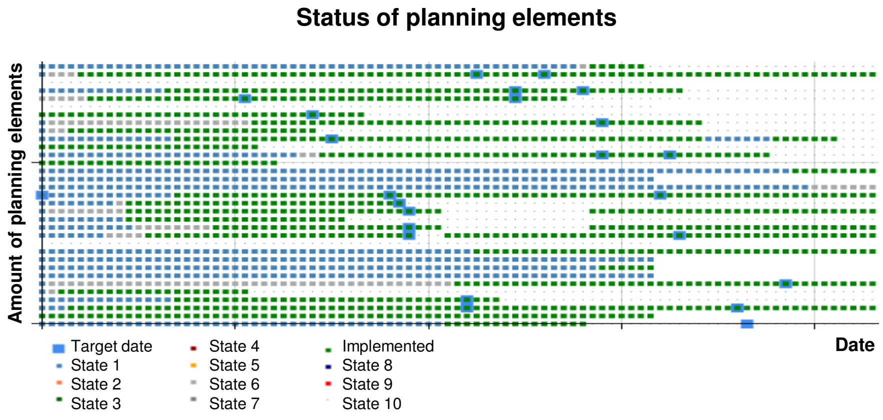

Figure 4-13. Status of planning elements over time, [45], [52]

Figure 4-12 shows the status of planning elements to check different durations and progresses by daily time steps starting from zero. Besides all status of planning elements, the most important parameters to detect deviations are target date and status called "implemented" where all functions have to be included within the system. Figure 4-13 depicts the investigation of planning elements over the time lane. This analysis method provides KPIs to control different parameters over time to detect weeks with huge amount of items in progress and checking legitimacy of peaks by mapping with milestones. Investigating the fulfilment of planning items in comparison with the planned target dates delivers a statement of possible time lags or critical elements, (blue squares in Figure 4-12 and Figure 4-13). Furthermore, a descriptive analysis includes various data, e.g., [45], [52]:

- Item ID
- Project
- Created date (triggers KPIs duration, loops and rescheduling)
- Planned releases (triggers KPIs number of loops and rescheduling)
- Completion release (triggers KPIs duration, number of loops and rescheduling)
- Classification (triggers detection of challenging areas)
- Assigned user
- Property (triggers KPI safety-related)
- Target date (triggers KPIs duration and rescheduling)
- SIL (Safety Integrity Level) (triggers KPI safety-related)
- Number of development loops (triggers KPIs number of loops)
- Time lag (triggers KPI rescheduling)
- Last record date (triggers KPI duration)

Besides some general information about project and item ID, comparison of creation/fulfilment date or planned/completion releases, this descriptive analysis method delivers a statement about project deviations. Hence, special attention lies on safety-relevant systems represented with the qualitative, significant indicators "priority" and "SIL" (safety integrity level). A further set of KPIs considering status and progress deals with a status overview to support the comparison of different project distributions, states, and progresses. For this purpose, relevant information is project milestones, completion status, and the current run-times in comparison to the whole project content or duration. To check the completion and duration status throughout the whole project, these parameters are also calculated for specific milestone iteration. Figure 4-14 shows these KPIs delivered by a new developed analysis method for the application of automotive software engineering considering all projects as well as individual software iterations related to a certain milestone. This analysis method calculates KPIs as percentage indicator of completion and duration status to deliver a statement about deviations between these parameters. In addition to evaluating the productivity of development, these KPIs support a comparison of different projects by the enabled overview visualisation. Thus, this method provides efficient planning and allocation of resources focusing on projects with a minor completion status and advanced duration, [45], [52].

Overview projects status

Project	Milestone	SW release	System release	Project completed	Project duration	Iteration completed	Iteration duration
Project X	MS 3	SW 1	SR 1	84%	77%	68%	68%
Project Y	MS 2	SW 5	SR 2	99%	98%	80%	80%
Project n	MS n	SW n	SR n	n%	n%	n%	n%

Figure 4-14: Overview of project status, c.f. [45], [52]

4.3.3.2 Durations

The analysis of durations of projects is an efficient process management method. The aim is to determine maximum and minimum run times of projects for enabling better understanding of durations to support new projects plans. The evaluation of realistic project durations realises optimised project planning, [45], [52].

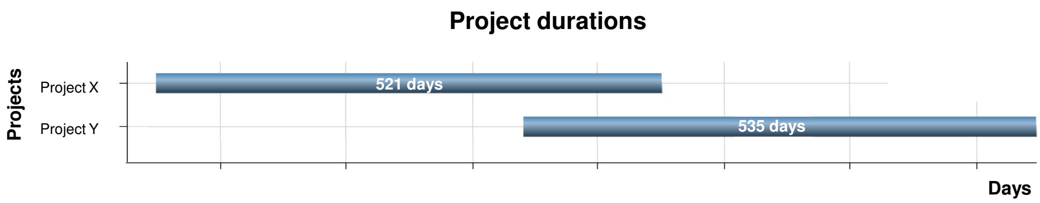

Figure 4-15: Comparison of project durations, [45], [52]

Additionally, KPIs considering the status of iterations allows a more detailed statement. Due to milestone planning and introduction of freeze points in development processes, evaluating durations and fulfilment of specific requirements in these iteration phases is an efficient method to enable control by process management. Special attention lies on durations of product maturity and different project phases as well as the observation of possible time lags due to non-compliance of freeze points during development, [45], [52].

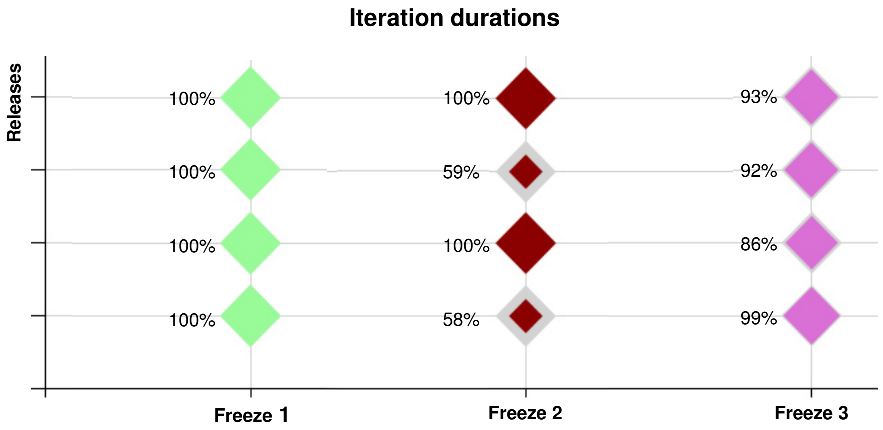

Figure 4-16: Visualisation of iteration durations, [45], [52]

4.3.3.3 Target-performance comparison

Target-performance comparison is an analysis method to deliver KPIs to control the progress of development by permanently aligning of targets and performance. Thus, this control method considers target values of desired requirements, related planning elements, tests, and detected faults related to specified releases and due dates. The highest priority for quality control is the fulfilment of parameters above a defined boundary at a certain milestone. The KPI fulfilment percentage of target-performance comparison can be a precondition to ensure high reliability and low risk in field uses. Furthermore, this method delivers a quality control gate for an appropriate degree of fulfilment of requirements and provides a trend evaluation. An early warning system can detect an increasing distance of target and performance to apply preventive risk measures as soon as possible in development process. Hence, using this analysis method in combination with boundary levels enables a statement about different measures being necessary due to criticality values. The aim is to get a high percentage of every parameter representing successfully tested systems with completely implemented functions and effective bug fixing. The statement of target-performance comparison supports decisions related to risk evaluation for releasing safe products, [45], [50], [52].

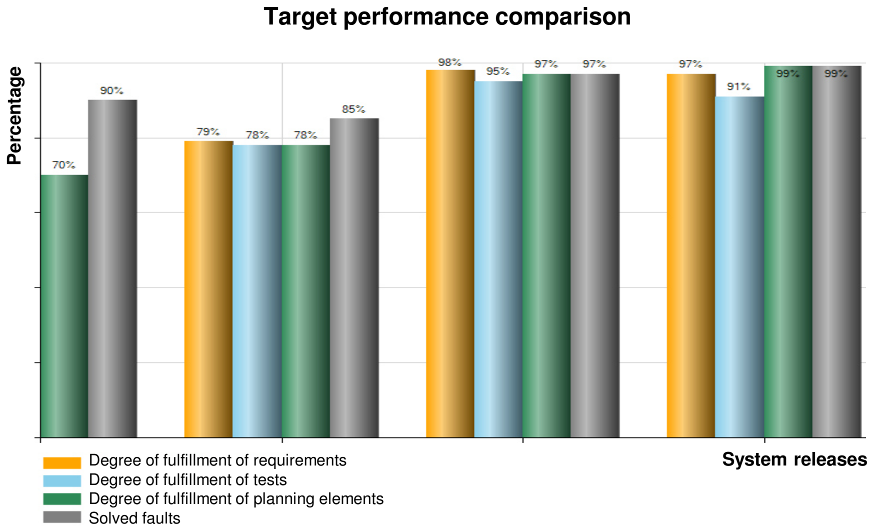

Figure 4-17: Target-performance comparison, [45], [52], c.f. [50]

4.3.3.4 Tracking

Tracking KPIs represents an evaluation similar to the target-performance comparison but with finer granularity. Besides checking the progress of development at subsystem or module level, this analysis method delivers the data for predefined KPIs considering the quality of data management, such as number of links and reviews. Hence, the architecture and specification of the system is necessary to check the product in detail. This detailed analysis provides an evaluation of connectivity of requirements, tests, and planning elements between system and subsystem levels. In addition, this investigation delivers statements about data management to answer several questions, such as “Are all requirements linked to test cases?” and “Did all requirements passed the reviewing process?”, [45], [52].

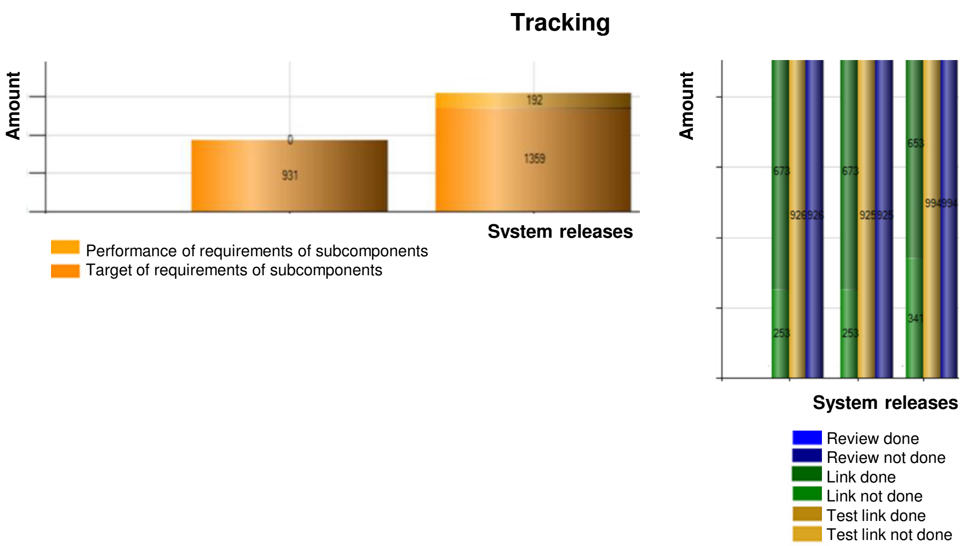

Figure 4-18: System tracking, c.f. [45], [52]

4.3.3.5 Trend of planning elements

On the basis of previous methods, an analysis of trends of planning elements enables KPIs to evaluate development performance tendencies. This analysis method supports understanding and checking projects' fulfilment status over product maturity. As shown in Figure 4-19, this analysis method visualises the course of target (orange line) and performance (green line). Thus, efficient checking if the gap between these lines increases, enables a preventive risk evaluation. For a more detailed analysis of this gap, see Figure 4-20. The mean of moving average curve supports the investigation of deviations between planned and fulfilled tasks. This trend line delivers a statement about possibly closing the gap and whether further implementation or resources have to be spent on the investigated project. The criticality of the gap increases with maturity of the project duration. With advanced project time, this gap should be as small as possible. This analysis method enables an early warning system to introduce measures as soon as possible to meet requirements and save costs induced by belated changes, [45], [52].

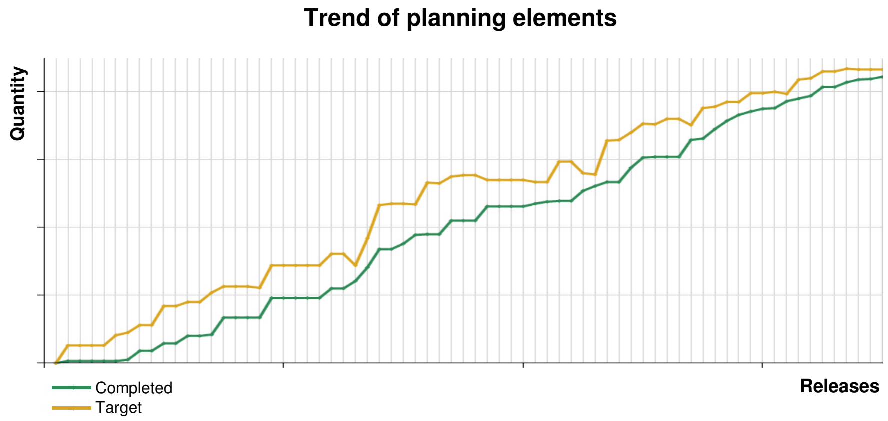

Figure 4-19: Trend of planning elements, [45], [52]

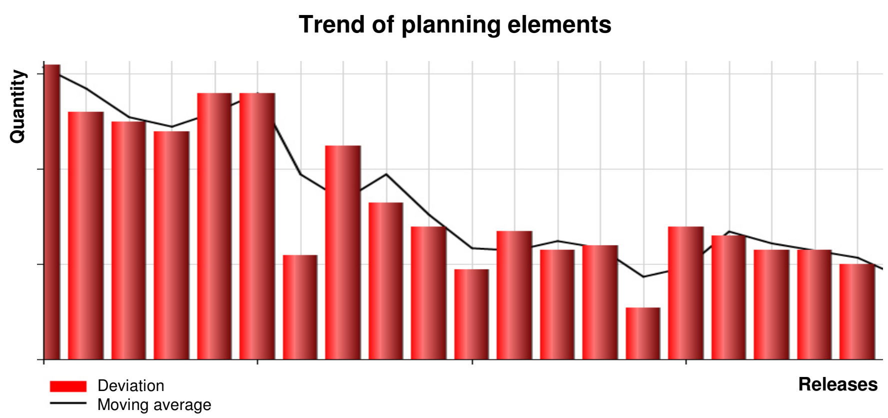

Figure 4-20: Trend of gaps of planning elements, [45], [52]

4.3.3.6 Trend of ratios

Another analysis method, the so-called trend of ratios also investigates KPIs considering trends in development. Thus, ratios as percentage values visualise the degree of saturation over development process time. The chosen KPIs depend on the relevant interests and success factors. Relevant KPIs represented as ratios are, [45], [50], [52]:

- Currently implemented requirements divided through all requirements
- Actually implemented planning elements divided through all planning elements
- Passed tests divided through all tests
- Fixed faults in comparison to detected faults

The distribution of these ratios and their comparison delivers a statement of the characteristics of the implementation process. The enabled investigation possibilities are varied. One example is considering the ratio “fixed faults” in comparison to “detected faults”. On the one hand it shows the amount of detected faults due to testing strategy, on the other hand it represents faults that happened during development in comparison to the amount of fixed faults. The correct statement and interpretation depends also on the investigation of other ratios including tests, planning elements, and requirements. Hence, an increasing number of tests and implemented functions is an indicator for higher number of detected faults, because the probability to detect faults increases with the quantity of testing and included functions. In comparison, products or systems including fewer implemented functions and tests delivering high fault rates are indicators for abnormality in the development process and induce need of further quality management. These high fault rates in testing combined with low complexity and functionality require an intensified focus on quality management of development to avoid increasing fault rates and high costs over project maturity, [45], [50], [52].
In addition, this analysis method supports investigation of saturation of different ratios. This important evaluation of specific KPIs supports an effective analysis of performance, process, and progress saturation distribution. The curve ideally reaches saturation as soon as possible representing efficient implementation process, effective bug fixing procedures, and successful management to enable high quality with low costs. Due to these parameters, this analysis method can also be used as an early warning system through permanent monitoring. Thus, defined boundaries related to different release levels and milestone mapping enable a check of degree of fulfilment at certain release points. Due to this function, resources and capacity can be allocated as soon as possible to meet the project targets. As shown in Figure 4-21, the course of the ratio of tests (blue line) can

show zigzag profiles or jumps due to looped tests, while ratios regarding requirements and planning elements should present continuous approximation to 100% saturation as soon as possible, [45], [50], [52].

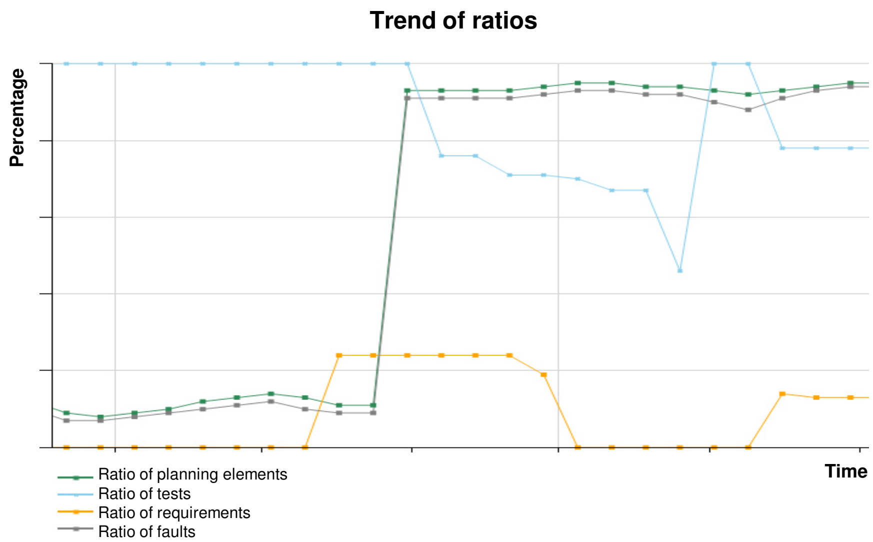

Figure 4-21: Trend of ratios, [45], [52]

4.3.3.7 Trend of solved and detected faults

The investigation of trends of newly detected and solved faults enables statements about bug fixing processes as well as the quality of the product, as shown in Figure 4-22. For this purpose, a moving average (e.g. over three months) delivers the evaluation of these trends. Thus, preventive checking of these curves supports the decision whether the testing procedure is successful and the bug fixing process is able to solve all faults. In the example, an increasing blue line represents more detected faults, whereas an increasing green line shows efficient fault removing processes. A decreasing green line and increasing blue line indicate complications during development and the fixing process, [45].

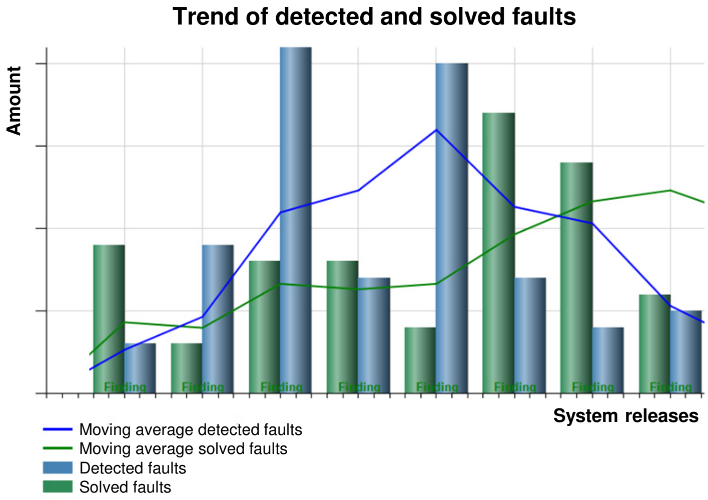

Figure 4-22: Trend of solved and detected faults during development, c.f. [45]

4.3.3.8 Faults after system release

System releases represent an important part in the development process, because the company delivers products or parts of the system to the customer. Hence, analysing if any faults appears after a release point supports the control of process, and quality of products, as shown in Figure 4-23. If faults appear after a certain release, project and quality management have to induce a new release or reschedule planning elements in further development phases which can be combined with additional costs and effort, [45].

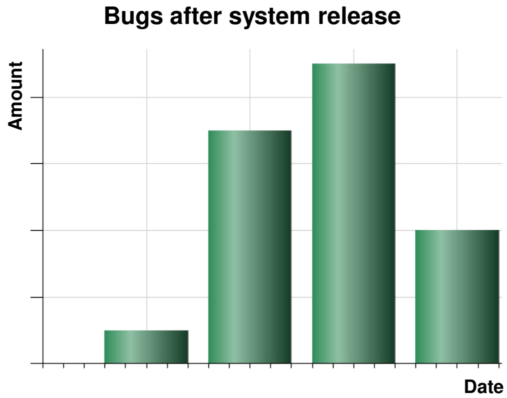

Figure 4-23: Trend of solved and detected faults during development, c.f. [45]

4.3.3.9 Fault estimation and modelling of residual error rate

Fault estimation

Special attention in development has to be paid to fault distribution over time. This evaluation allows a comparison of detected faults as target value (orange line) with fixed faults presented as completed (green line), as shown in Figure

4-24. To increase the reliability of the product or system the gap between the target KPIs of detected faults and the actual value of fixed faults has to be minimised. Closing the gap over the project maturity enables a reliable product and release with minimal risk, [45], [52].

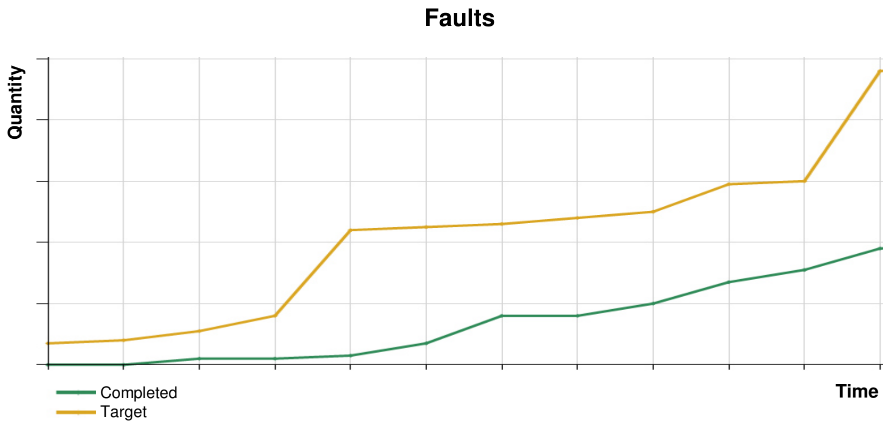

Figure 4-24: Fault estimation e.g. over software releases, [45], [52]

Another investigation method delivers the data for predefined KPIs to analyse, whether faults have been unedited for a longer period of time, see Figure 4-25. This method is used to improve process and quality management, because there can be two reasons which affect unedited items. The first reason can be too little data or project management resulting in forgetting to edit or delete faults which are maybe irrelevant. The second more critical reason belongs to project and quality management and includes forgotten important faults which are still in the system and not fixed, [45].

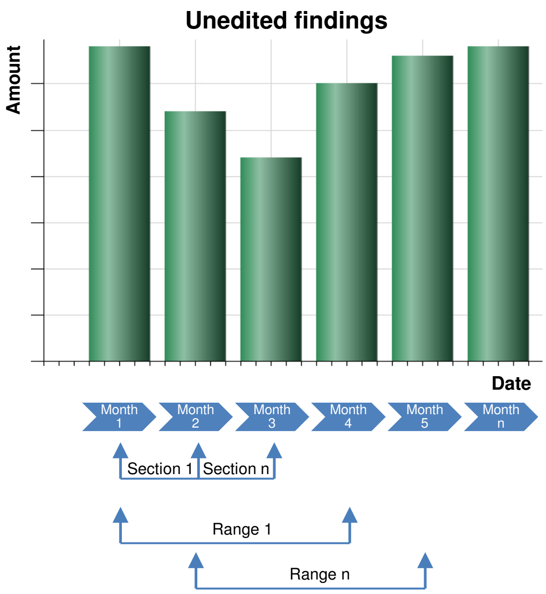

Figure 4-25: Remaining unedited faults, c.f. [45]

Modelling of residual error rate

Changing from descriptive to inductive statistics allows modelling of faults during development. Complex mechatronic systems need special attention to model fault distributions. Due to interactions of mechanics, electronics, and software, analysis methods face the challenge to estimate faults with various characteristics and circumstances. For instance, mechanics and electronics show an aging effect over lifetime. Reliability analysis using distributions, such as Weibull distribution, have to consider this effect (see Chapter 3.3.2.3), [10], [52], [49].
Considering software components as part of mechatronic systems, the aging of software parts does not appear in this way. Other circumstances can have a higher significance for modelling fault rates of software components, such as repair rate, MTBF (mean time between failures), MTTF (mean time to failure), or MTTR (mean time to repair). During software development, faults can occur due to further implementation processes due to bug fixing or as a result of previous events. These characteristics have to be considered during the selection of the reliability analysis method using distribution models. Primarily in software development, fault distribution characteristics can look like a saw tooth curve due to various impacts. Figure 4-26 shows an idealised curve (green line) representing one bug fixing period and a fault rate with tendency to zero. In comparison, the actual curve (blue line) shows a more realistic distribution including several development steps, bug fixing periods or new faults in every revision period due to side effects and further implemented functions, [45], [49], [52], [155].

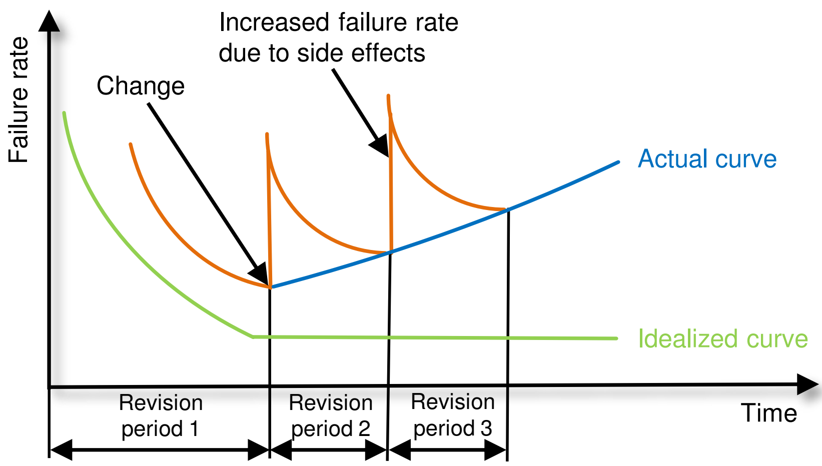

Figure 4-26: Fault estimation of repairable systems in development, c.f. [52], [155]

An analysis method considering fault estimation and prediction enables a significant KPI called residual error rate. This indicator is a very efficient way to check the trend of faults during development. There are various possibilities to apply

this method during development processes. For instance, modelling faults detected by testing enables – besides the definition of the residual error rates in different phases – a trend analysis of further development. Thus, this trend evaluation supports decision-making about further bug fixing or the definition of the end of testing. Furthermore, it delivers a basis for statements in view of the faults remaining in the system supporting risk analysis and releasing software at a certain point of time, [45], [49], [52].
The choice of distribution model for this analysis method also depends on the test environment in the development process, e.g. the automotive software in V-model contains MiL, SiL, PiL, and HiL tests. Various impacts occur by testing systems. The evaluation of faults has to consider circumstances like the variable mean time between failures, changing fault rates, or a permanent repairing process by bug fixing. Furthermore, during development processes the amount of functionalities and the maturity of the software changes permanently resulting in different releases. Modelling faults in software development has to include these complex approaches and characteristics. As mentioned in chapter 3.3.2.3, the selection of distribution models is differentiated into macro models, including external functions as the black box and micro models also considering internal structures as the white box. In detail, the selection of a suitable model depends on different characteristics, such as process status, previous history, or failure intensity behaviour. For instance, Markov Process models provide characteristics such as perfect repair by fixing faults immediately. Thus, each fault appears only one time and cannot influence the further process. In addition, the subcategory of binominal models assumes faults have the same criticality and a fixed number before testing. In comparison, Semi-Markov process models consider – besides the current point of time – the last time of failure. It is also possible to include the condition of further faults due to bug fixing. For software development and the mentioned characteristics, Non Homogenous Poisson Process models (NHPP) are an efficient method to enable fault distributions due to the assumptions and attributes (such as function of time, non-relatedness to the past, consideration of repairing further faults caused by bug fixing, and development), [45], [49], [52], [116].

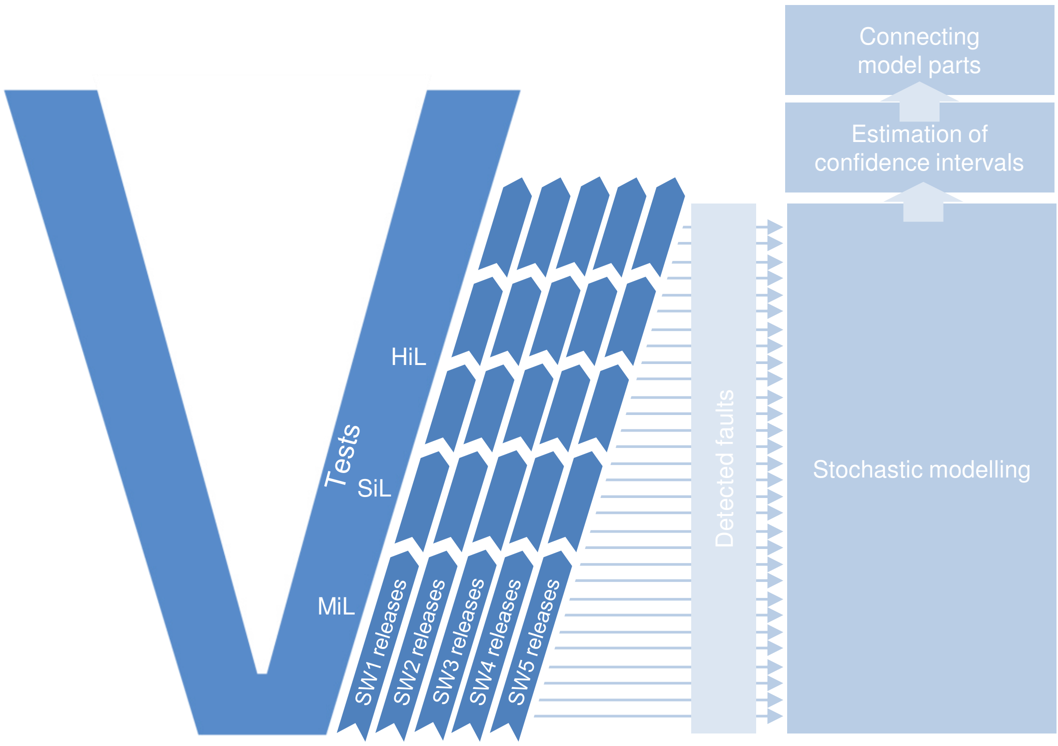

Figure 4-27: Approach of the analysis method to model the residual error rate

Figure 4-27 shows the approach of the analysis method to model the KPI residual error rate. The testing process of different software releases during the integration path of the V-model delivers detected faults over time. These parameters are the input for the stochastic modelling process. In detail, the new developed analysis method uses the power law intensity function of NHPP to enable a realistic modelling of the software engineering process, such as automotive applications, [36], [45], [49]:

$$\lambda_{(t)} = \left(\frac{t}{\theta}\right)^{\beta} \ldots failurerate \tag{4-1}$$

$$\lambda_{i(t)} = \frac{d\lambda_{(t)}}{dt} = \frac{\beta}{\theta} * \left(\frac{t}{\theta}\right)^{\beta-1} \ldots instantaneous\ failurerate \tag{4-2}$$

For parameter estimation, using the maximum-likelihood method is advisable with the assumption that the censored time t_i of one observation point n is shorter than the survival time t_R of the system, [5], [49], [52]:

$$\hat{\beta} = \frac{n-1}{\sum_{i=1}^{n} \ln \frac{t_i}{t_r}} \ldots non\ homogenity\ parameter \tag{4-3}$$

$$\hat{\theta} = \frac{t_r}{n^{\frac{1}{\hat{\beta}}}} \ldots scale\ parameter \tag{4-4}$$

Using the NHPP model to enable an efficient fault distribution analysis requires information and data about detected faults during the development process, related time in weeks, and milestone mapping. Software development is a changing process with varying fault rates, variable mean time between failures, continuous bug fixing, and increasing amounts of functionalities and/or requirements over software releases. Thus, multiple process modelling at every important milestone represents the development process including increasing functionalities and maturity rather than one simulation over the whole development timeline, [49], [52]. The analysis method enables the KPI residual error rate, which represents a metric to support risk analysis and release decisions at a certain point of time. Modelling failure distribution also includes a scattering scope regarding the error rate. For this purpose, the method includes the definition of the 95% confidence interval of the model distribution by using a Fisher information matrix, [114]. This interval represents the calculated and expected maximum, minimum, and average of residual error rate at a system release point, [5], [49], [114]. The combination of multiple recharging of NHPP modelling over various release phases and use of confidence interval estimation enables a realistic failure modelling as well as prognosis of the residual error rate. This analysis method and KPI support risk analysis by delivering a statement of the amount of remaining faults in the system. Furthermore, modelling enables a trend analysis to define whether further development, bug fixing, or tests are necessary or whether these tasks can be closed because of a residual error rate under a defined boundary, [49], [52].

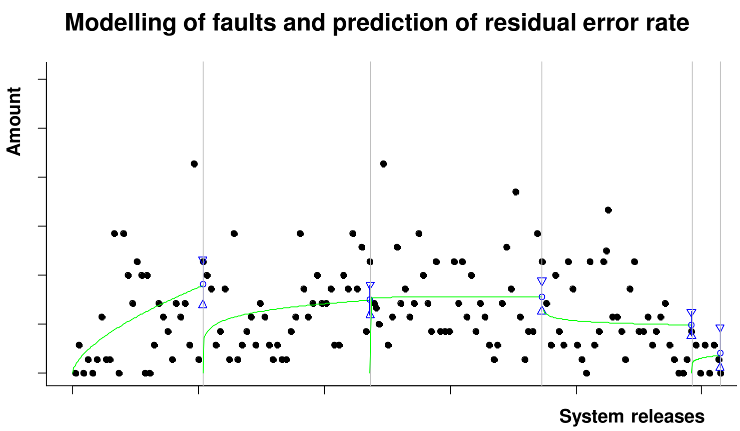

Figure 4-28: Modelling by using of NHPP to get residual error rate (main releases), c.f. [45], [49], [52]

Exemplary results of the analysis method including those of the NHPP model is shown in Figure 4-28 as a green line repeated for each major release. Figure 4-29 depicts the NHPP model considering all releases including intermediate stages. The typical saw tooth profile is more distinctive in Figure 4-29 due to the shorter revision periods. The blue line in each figure indicates the 95% confidence interval estimation by the Fisher matrix representing the area of residual error rate at a certain release point, [5], [49], [52].

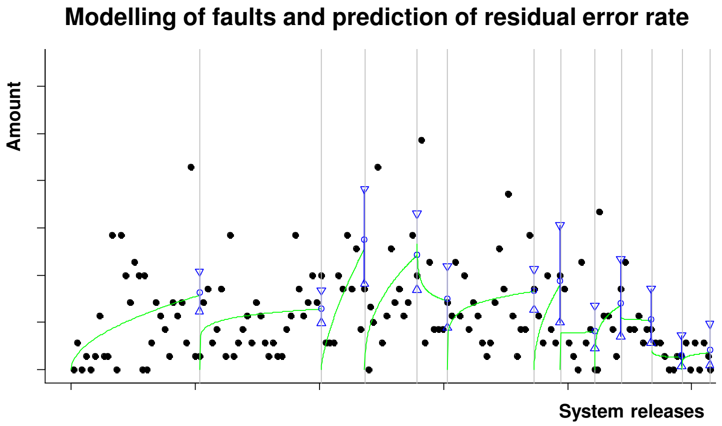

Figure 4-29: Modelling by use of NHPP to receive the residual error rate (all releases), c.f. [45], [49], [52]

This method enables a calculated residual error rate for risk estimation. Additionally, this method including tests in development should be compared to faults of field use in customers' cars. Due to cost and time pressure, tests during development are often carried out in protected test environments. Thus, the comparability to field use conditions has to be considered. While the residual error rate defines the faults remaining in the system, the faults in field use represent the actual failure rate. To get the residual error rate even in a development process, which is representative for the failure rates of cars in the field (in customer use), as well as boundaries for risk analysis in development, the prognosis has to be combined with field use data. Therefore, a field and complaint data investigation is crucial, [45], [52].

4.4 Analysis methods using KPIs on the basis of field and complaint data

The following content incorporates and extends the research of [130], [136], [37], [113], and [191].

4.4.1 Theoretical introduction

The next step of the developed methodology deals with KPIs on basis of field and complaint data. Due to increasing quality demands of customers, standards and increasing cost and time pressure during whole PLC, companies need to store and analyse various information of product behaviour, possible faults, as well as their failure causes in field use. For this purpose, the feedback of field and complaint data delivers important information about the PLC, such as fault memories, behaviour of products, diagnostic information, and wear out data. This feedback into development processes enables an enhancement of the quality of the systems and further product generations. Field data includes additional information, which often cannot be provided by development tests, e.g. misuse of customers or unforeseen circumstances. It is available in form of fault memories, service inspections, ECUs and on board or off board diagnostics. These data include various information about the product behaviour during field uses, for instance failures, mileage, temperature, fault codes, and environmental issues. These information sources deliver analysis potential by KPIs to increase reliability and quality of products as well as processes. To enable the possibility of quality improvement, fault evaluation and failure prognosis needs extraction of information from field use. This allows an investigation of the product behaviour to understand systems and processes as well as to improve further development and product generations, [28], [39], [43], [117].

4.4.2 State of the art and limits

The state of the art shows different approaches that try to analyse and feedback field data (see 3.1.5.2). Big data or concepts like OpenMDM are a further step in the right direction, but they do not include a satisfying approach to provide comprehensive feedback. An effective analysis and feedback of different available field data is not solved efficiently yet. Analyses of field data often are limited to overall failure distributions and do not distinguish different fault types and their causes. Nevertheless, this research area needs enhancement and new concepts to increase quality of development processes and products.

4.4.3 Method

4.4.3.1 Data management and storage

Field and complaint data can have several data sources. The use of data sources is related to the possibilities of a company to provide this information. Due to the rising application of mechatronic systems in industry and connection of several systems, also the number of data sources is increasing. Figure 4-30 shows the example of possible data sources of field and complaint investigation in automotive industry. Figure 4-31 shows another example of possible data sources for the application industry 4.0.

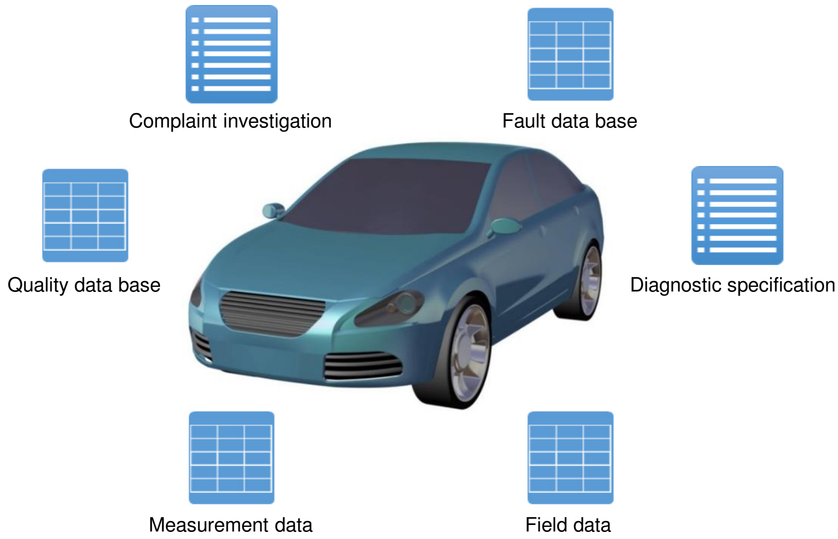

Figure 4-30: Different data sources of field and complaint investigations in automotive industry

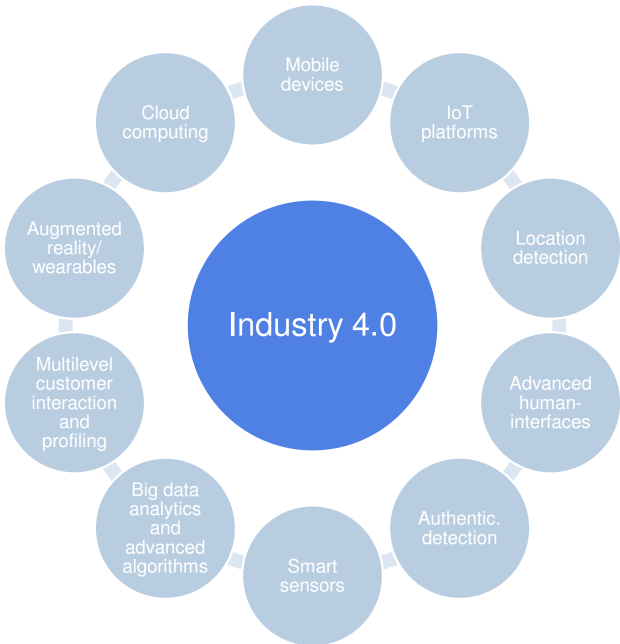

Figure 4-31: Different data sources of industry 4.0, c.f. [141]

Possible data sources deriving from three divisions:

- Engineering:
  - Diagnostic specification:
    Diagnostic specification is a very important data source from the development phase. Usually, it includes a categorisation and further information regarding system architecture and failures, such as failure ID, failure type, storage type, description, and repair measurement.

    - Quality database:
      Depending on companies and quality management, databases for quality investigation are often available. This source usually delivers a communication platform between customer and supplier including various information, e.g. complaint investigations.
- Complaint or diagnostic investigation:
    - Complaint investigation:
      Complaint investigations represent one of the most powerful possibilities to get data of a real product's use. Experts conduct an exact investigation of complaints to enable precise information about the system and failure types or causes.
    - Fault database:
      Efficient complaint and diagnostic investigation divisions store various failures and complaint data in fault databases. This source represents the maintenance of knowledge over different projects resulting in an improvement of quality and know-how.
- Customer:
    - Field data:
      Field data usually comes from customer or customer-related service departments. This type of data is one of the most important sources, usually including failures, environmental data and other measurements.
    - Measurement data:
      Measurement represents continuous tracking of different measures rather than failures. These data include continuous information about behaviour and relevant parameters. Due to connection of mechatronic systems (e.g. for autonomous driving), this sector and its importance are increasing.

The incompleteness or incorrectness of the available data is a main challenge. A high data quality and completeness degree are preconditions for significant statements of analysis. Various data sources enable the possibility of checking this data in terms of fulfilment and correctness. Check routines can adjust incorrect fields by comparing them to enable high reliability of correctness and details. Furthermore, incomplete fields can be filled up and redundancies can be avoided. Additionally, considering various projects or customers (e.g. OEMs) represent the challenge of different data structures, scope, and content. To ensure a generic method and a comparative table in the form of equivalence list, a generic definition of parameters is necessary. Thus, a generic approach to generate a relational database is shown in Figure 4-32. Data resources show all kinds of information with different structures. For this purpose, parsers enable to read out information and include them as generic information in the database. Thus, this method enables a structured database by using different parsers to ensure generic definition of all items. This approach ensures one database for different projects and all kind of data structures or content. Even if the process automatically runs through, values in the database are still changeable. The generic data model combines and connects the data to deliver a platform for various, efficient analysis possibilities. This method is based on the equivalence list, see appendix A-1.

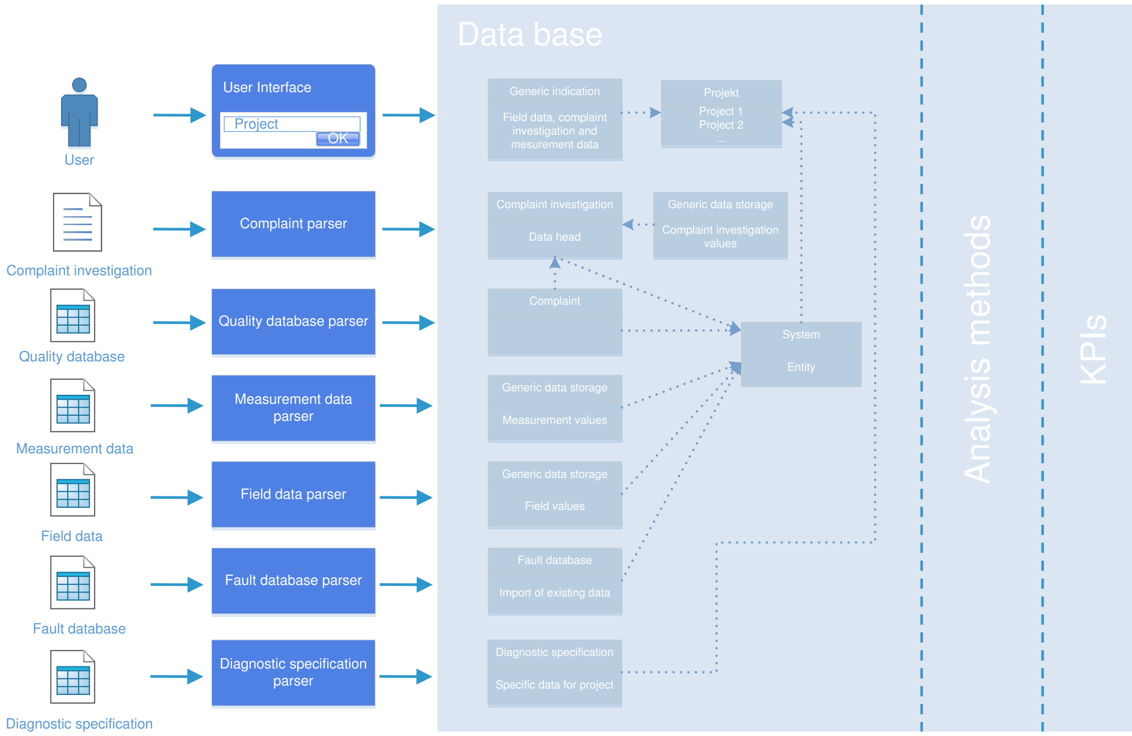

Figure 4-32: Approach of building a generic and relational database

On basis of this generic and relational database, various KPIs (see chapter 4.4.3.2 and chapter 4.4.3.3) delivered by new developed analysis methods can be introduced to enable feedback and knowledge of real product behaviour. Thus, two analysis categories can be differed:

- Retrospective analysis
- Predictive analysis

4.4.3.2 Retrospective analysis methods

Retrospective analysis methods evaluate data of the past and generate information or knowledge about behaviour of products in field use by the customer.

4.4.3.2.1 Analysis method considering KPIs related to failure frequencies

This descriptive analysis method delivers an investigation of the KPIs related to failures over field uses based on the connection of field and complaint investigations. Due to the generic database, the connection of the data sources combined with the diagnostic specifications enables different possibilities to visualise failure frequencies. The first general option is to show failures and their frequencies, as depicted in Figure 4-33. The algorithm calculates KPIs as sum of failures related to applied filters.

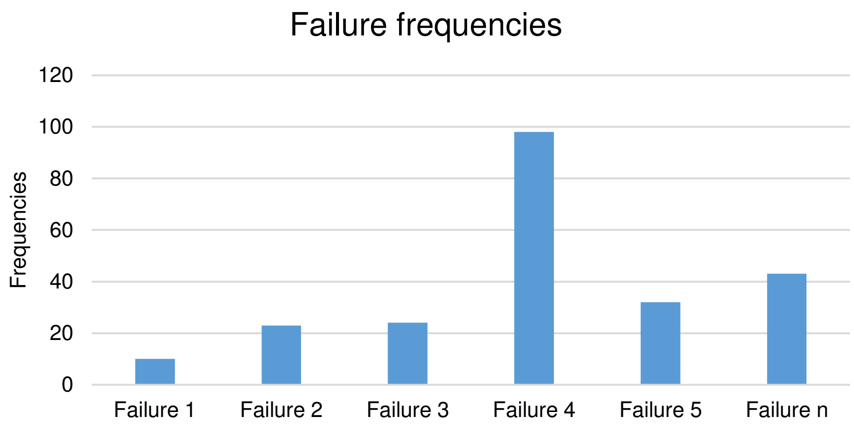

Figure 4-33: Failure and their frequencies, c.f. [131]

An application of filters due to various items in generic database enables a huge potential of various investigation possibilities, such as specific failures or all failures related to product models or systems (see Figure 4-34 and Figure 4-35).

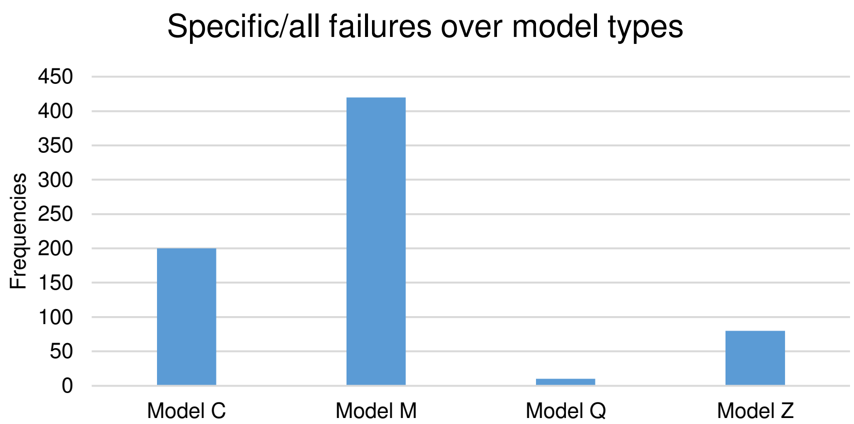

Figure 4-34: Specific/all failures over different product models, c.f. [131]

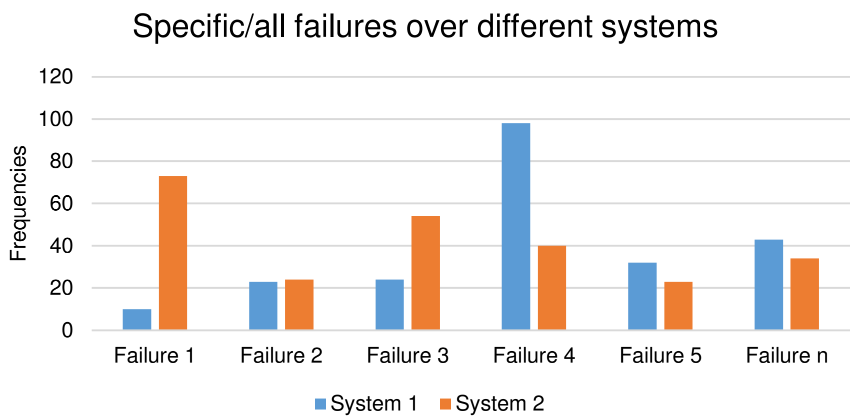

Figure 4-35: Specific/all failures over different systems, c.f. [131]

Another possibility to visualise KPIs related to failure frequencies is to refer them to different failure categories of the mechatronic system, e.g. classifying them as hardware (mechanics and E/E), software, and unspecified failures. The data basis and algorithm is similar to previous analysis methods, but the filter function can include more items to investigate each failure category:

- Hardware-related failures
- Software-related failures
- Unspecified failures

These analysis methods affect engineering and quality assurance by delivering a statement of fault category occurrence.

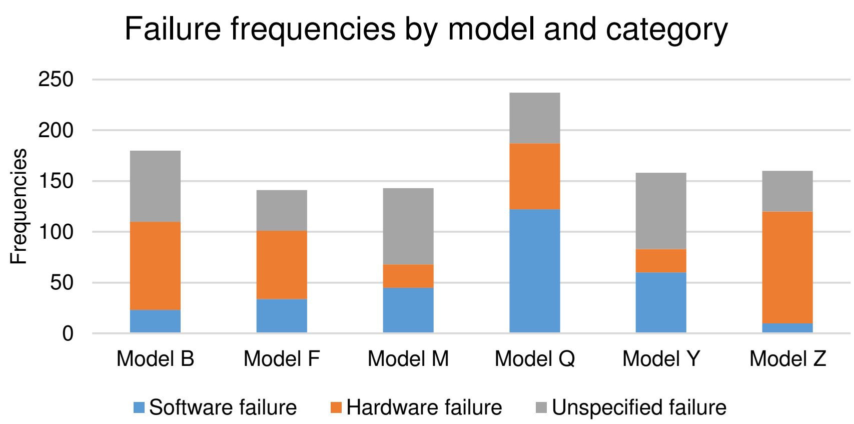

Figure 4-36: Failures related to different models and categories, c.f. [131]

The next diagram shows the KPIs related to failure frequencies over use metrics, e.g. time or mileage. This method delivers a statement of how many failures appear over the time lane. Besides data of previous analysis methods, the parameter information of use and production data is required to enable the related algorithm that calculates failure frequencies as function of use metrics.

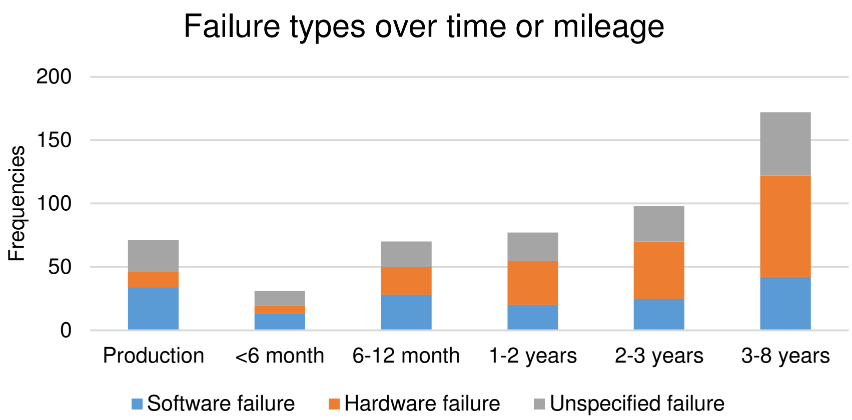

Figure 4-37: Failures over time or mileage categorised by type, c.f. [131]

The next analysis method shows the KPI failure frequencies in the field related to production date of the systems. The necessary data, applied filter, and interest groups are similar to the previous visualisation. The algorithm detects failures in a specified time range and relates it to production phases in form of months.

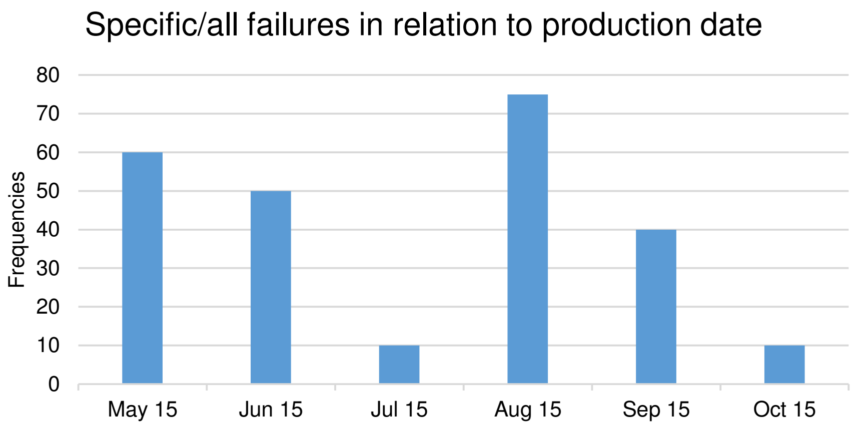

Figure 4-38: Specific/all failures over a defined field use related to the production date in months, c.f. [131]

Analysis of product use under different environmental conditions can be very important, such as the investigation of failures related to countries. Thus, the additional information (e.g. the country where the product has been sold) is required to allow an algorithm summing up failures in relation to countries.

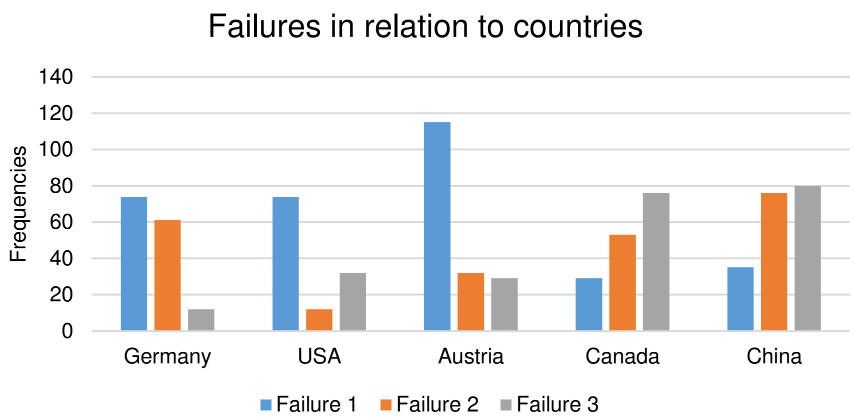

Figure 4-39: Failures related to user countries, c.f. [131]

Another very interesting analysis possibility in the special case of mechatronic systems is to show the relationship between software releases and failures. For this purpose, further information of the generic database is necessary, such as software releases. The query sums up failures in relation to SW releases. This investigation is very important for engineering and quality assurance.

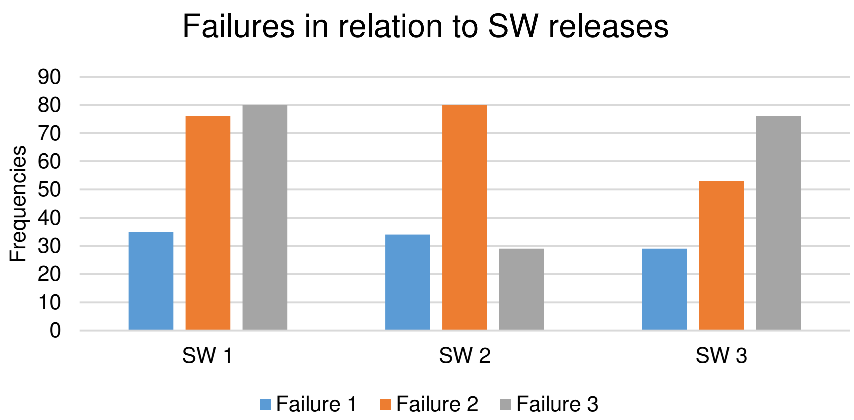

Figure 4-40: Failures related to SW releases, c.f. [131]

4.4.3.2.2 Analysis methods considering KPIs related to the process of changes in field uses

The next investigation group deals with different analysis of change processes of software during field uses of mechatronic systems. For this purpose, further information of the generic database is necessary, e.g. SW release of rollout point and current SW release. The query of this method calculates the KPIs as sum of products with specific SW releases and enables analysis for engineering, quality assurance, and complaint investigation. The first investigation in this group is relevant for engineering and shows how much products include the SW level of roll out in comparison to products with new flashed SW version, as shown in Figure 4-41. The flashing process of a new SW release is shown in Figure 4-42. This analysis is relevant for engineering, quality assurance, and complaint investigation to evaluate how long the process of replacing a SW level with bugs can take. With a similar method, the chronological approach of SW implementation on different project generations can be considered.

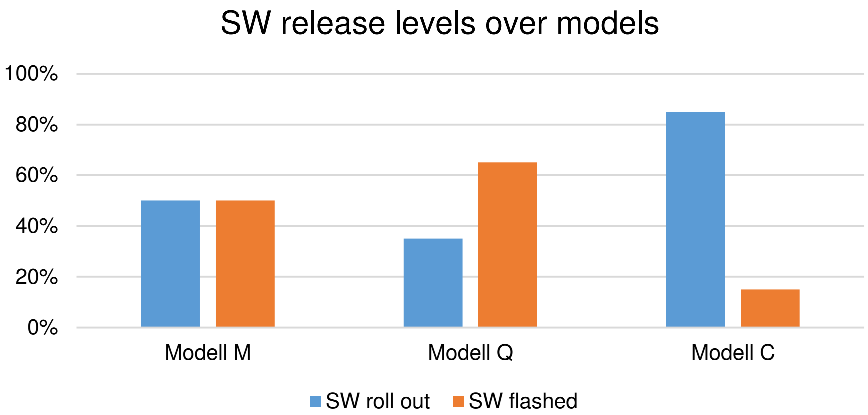

Figure 4-41: Comparison of percentages of roll out and new SW release levels in relation to different vehicle types, c.f. [131]

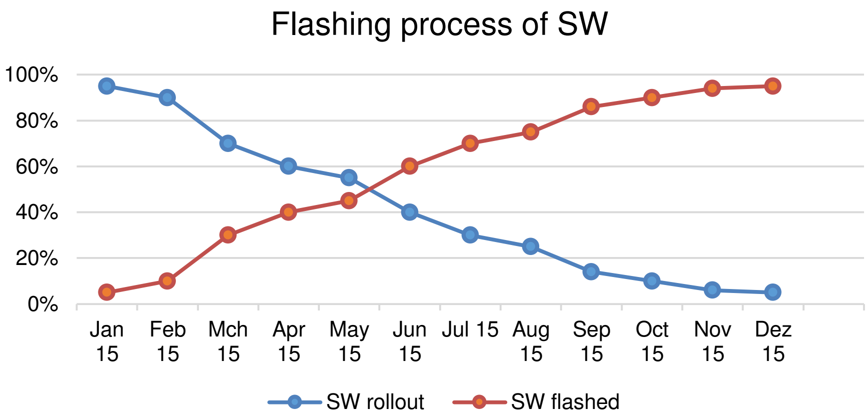

Figure 4-42: Flashing process of SW releases, c.f. [131]

4.4.3.2.3 Analysis methods considering KPIs related to processing of tasks during field uses

The next investigation group deals with processes of tasks during field uses. For this purpose, all target and performance dates are necessary. For this analysis method, an algorithm calculates the days of complaint processing of different products in comparison to the complaint entrance date and delivers a KPI about the whole complaint process for reporting and quality assurance.

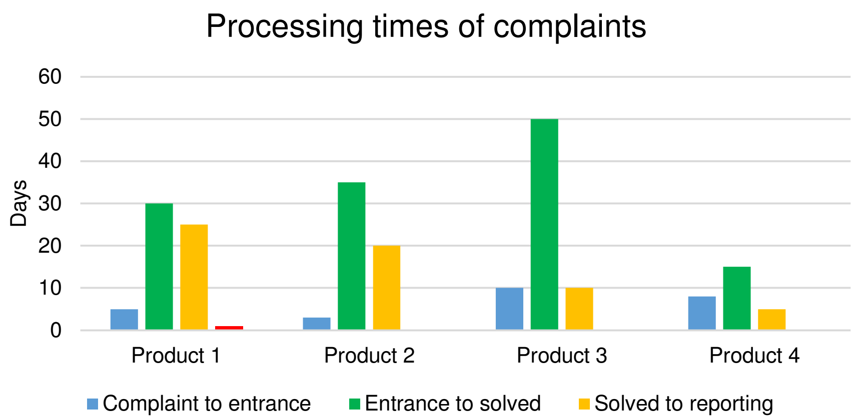

Figure 4-43: Complaint investigation processing times, c.f. [131]

The next investigation deals with delays in terms of registration. This analysis is interesting for lifetime distributions considered by quality assurance. Due to the generic database and further information about registration date, delay can be calculated as difference for several models and types.

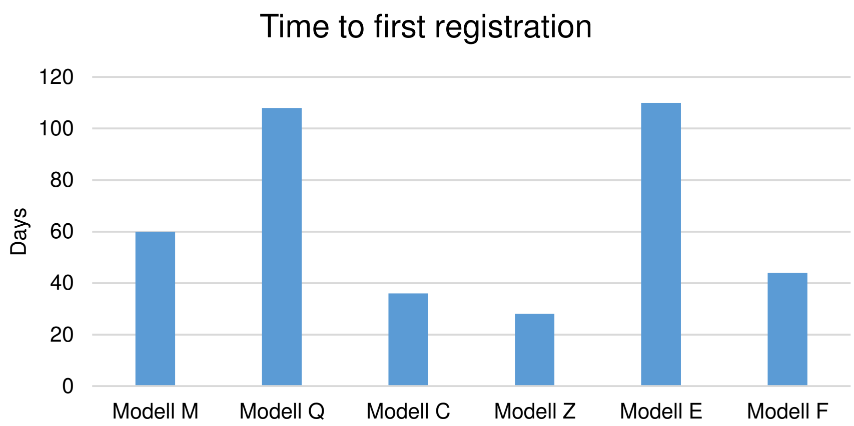

Figure 4-44: Delay of registration, c.f. [131]

4.4.3.3 Predictive analysis methods

Predictive analysis methods evaluate data of the past and generate information about future behaviour of products in the field use by the customer.

4.4.3.3.1 Control basis

KPIs generated by predictive and preventive analysis methods are usually based on the measurement data of field use, but can also use information of complaint investigations and field data. Furthermore, these proactive methods need boundaries to check if a critical value is reached or not. Thus, checks are controlled in predictive analysis methods and are represented by the following relations, [136]:

- Comparative value (which is directly controlled)
- Comparative basis (comparison to value basis)
- Comparative symbol (represent relations as ranges)
- Severity (enabling warning levels)

These relations enable different control types, such as, [136]:

- Static control:
  Static control allows the comparison of two values (measurement and comparative value)
- Reference-based control:
  Reference-based control compares measurement values with different attributes
- Complex control:
  Complex control enables a comparison of measurement values to different combinations of boundaries and checks using equations

4.4.3.3.2 Analysis methods considering KPIs related to wearing or other use values

For mechanical and E/E parts of the mechatronic system, the analysis of the wearing process predicts if a breakdown can appear in the next time. Hence, this preventive analysis method requires information about measurement and complaint data, such as customer, project, and several wear values. Furthermore, boundaries or other checks are necessary to get comparable values or rules. An applied filter allows the investigation of specific customer, projects, products, systems, field uses due to mileage or time, and individual checks. The algorithm calculates KPIs of relevant wear data related to parameters, such as mileage or time, and compares values to boundaries enabling an early warning system to investigate if a problem in field use is coming up. This analysis method delivers the data for predefined KPIs which influence all areas of engineering, quality assurance, and complaint investigation. A very important part represents the prediction of current component behaviour. Figure 4-45 shows predictive areas by dashed lines as linear symbols. These linear lines can denote modelled curves using a stochastic distribution to simulate real behaviour of components (see Chapter 3.3.2.3).

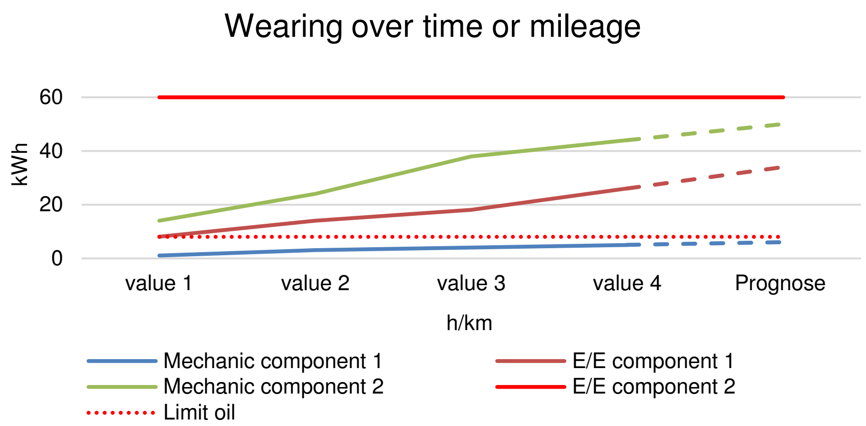

Figure 4-45: Preventive analysis, exemplary of wearing in field use by the customer, c.f. [131]

The analysis of KPIs considering use values of system delivers predictive statements of critical values over time. For this purpose, the next preventive analysis method requires information of measurement and complaint data, such as customer, project, product, mileage or time, subcomponent number, values of use (e.g. temperature level), and readout time. Furthermore, boundaries or other checks are necessary to get comparable values or rules. The applied filter enables the investigation of specific customers, projects, product, subcomponents, or time intervals. The algorithm calculates the distribution of use values (e.g. temperature) related to processing time. The comparison boundaries allow an early warning system to get a statement whether a problem is coming up. This analysis method influences all areas of engineering, quality assurance, and complaint investigation.

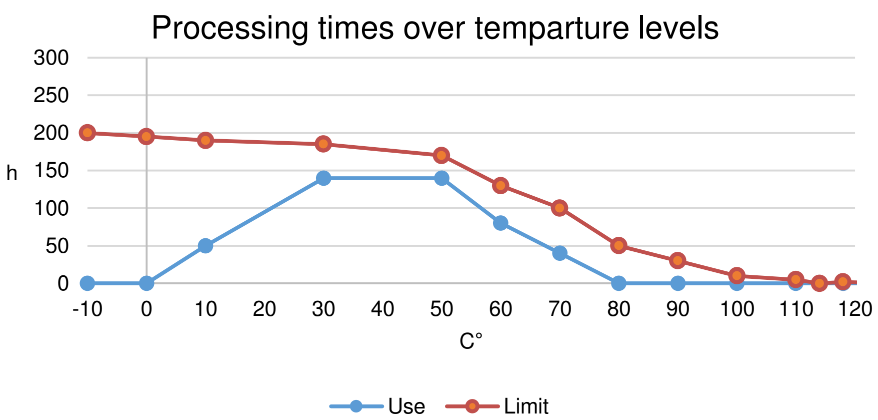

Figure 4-46: Distribution of processing times, exemplary over temperature, c.f. [131]

4.4.3.4 Absolute and relative values

Analysis of failures have to consider absolute and relative KPI calculations. While absolute values represent the amount of all failures during field uses, relative values are related to the number of systems or products in field use. In case of breakdowns, relative values represent the significant KPI of real failure rates. Therefore, the analysis methods have to be able to calculate the number of used systems by combining measurement and field data and ensuring the avoidance of duplicates. Figure 4-47 shows the absolute amount of failures and Figure 4-48 represents the amount of systems in use. The result is the quotient of these parameters as relative failure frequencies, as shown in Figure 4-49.

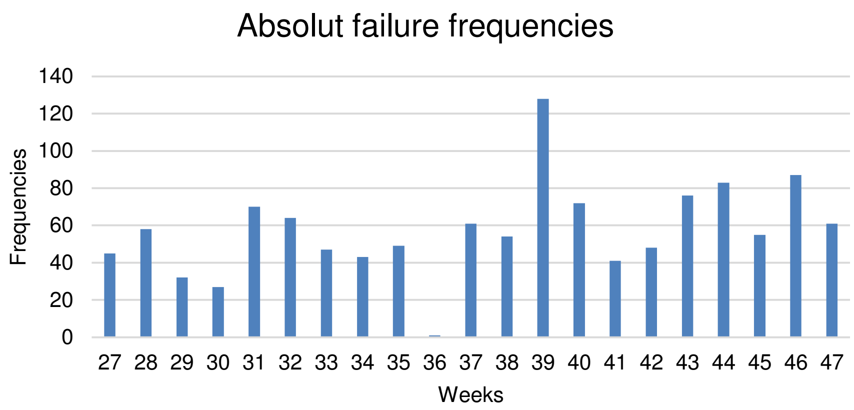

Figure 4-47: Absolute failure frequencies, c.f. [131]

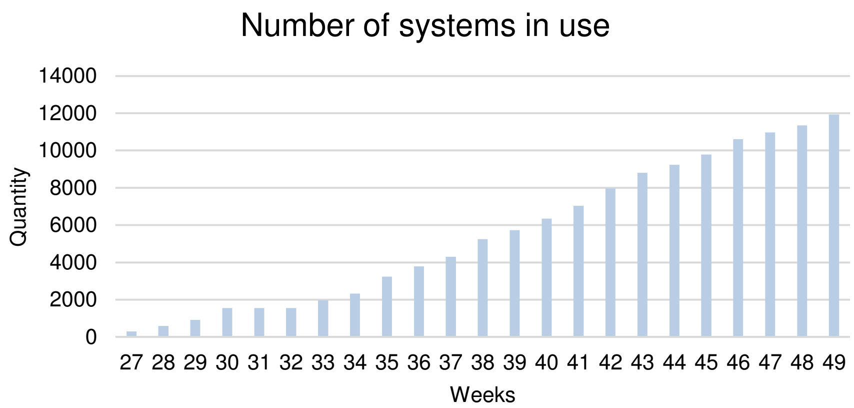

Figure 4-48: Number of systems in use, c.f. [131]

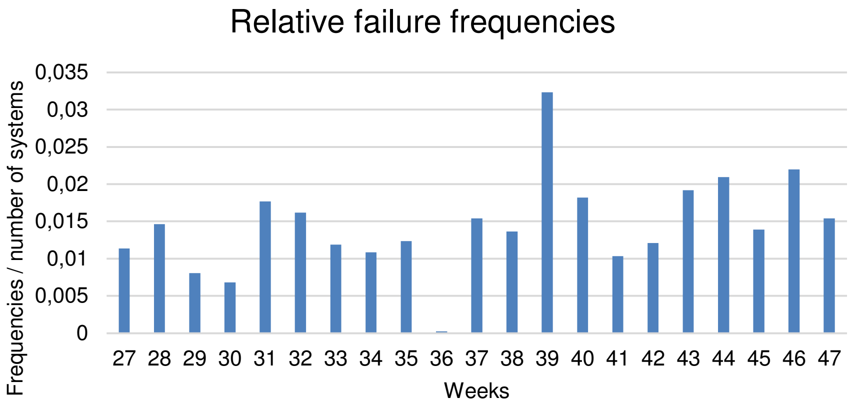

Figure 4-49: Relative failure frequencies, c.f. [131]

4.5 Stochastic reliability investigations in development and field use

4.5.1 Theoretical introduction

Due to increasing complexity of systems, quality requirements from customers and safety standards, industrial quality management has to enable efficient and effective reliability analysis during development and field uses. Besides the above-mentioned new analysis methods, special attention has to be paid to stochastic reliability investigations of faults during whole PLC. The combination of different disciplines of mechatronic systems is an additional challenge for this evaluation. Modelling of the expected residual error rate during development in comparison to real failure rate during field uses delivers basic information for decision-making and risk analysis, [49], [52].

4.5.2 State of the art and limits

The stochastic investigation of both development and field or complaint data represents a difficult and not yet satisfyingly solved topic in industrial applications. Different methods try to face subproblems of this research area, but do not deliver effective solutions. For further information see chapter 4.3.2 and 4.4.2. The limits of the common methods are e.g. focus on only separated phases of PLC, no differentiation of failure categories, and missing comparison or feedback of field data and development processes. Thus, new approaches and stochastic analysis methods are required to enhance quality management.

4.5.3 Method

This new developed reliability investigation follows an approach in stages, as shown in Figure 4-50.

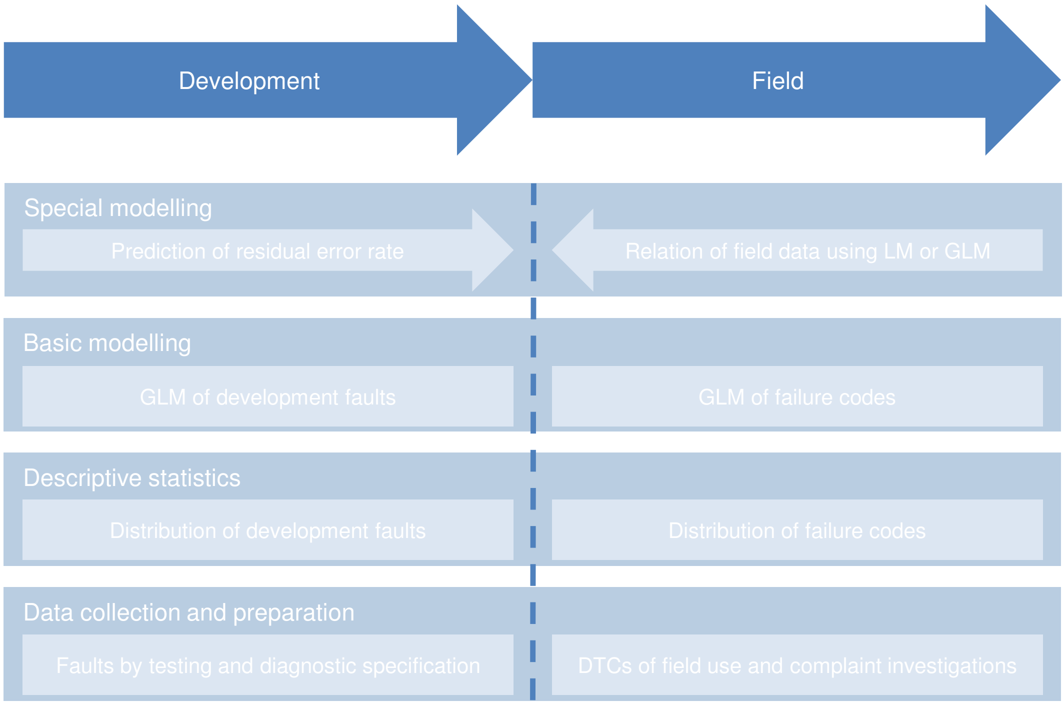

Figure 4-50: Approach of reliability investigation

The first step represents collection and preparation of data. All information about faults during development and field use has to be collected. In development, this data represents findings due to testing and their relation to milestone planning. In field use, the necessary data are failures and complaint investigations as well as diagnostic specifications from engineering. The preparation of this data is very important to provide a representative database. Hence, impacts on data collection have to be correct, such as redundancies, failures during data input, delays, reasonable negligence not representative or unimportant data, and normalising. The next stage of the approach represents analysis using descriptive statistics. This investigation shows faults in development and failures in field use related to periods or milestones. This analysis enables an overview and evaluation of distribution of the collected data (for development see Figure 4-22 and Figure 4-26, for field use see Figure 4-47 and Figure 4-48). The next stage contains basic modelling. This stage enables modelling and investigation of faults in development or failures in field use (compare chapter 3.3.2). To describe distribution of faults, an efficient method of regression analysis (compare chapter 3.3.2.4) is enabled by use of so-called generalised linear models (GLMs) as an enhancement of linear models (LMs).

LMs describe both systematic and error/random components assumed as normal distributed by followed equation, [122], [124]:

$$Y(x) = \sum_{i=1}^{n} \beta_i * x_i + \varepsilon$$
$$x_i \ldots regressor\ variables\ or\ covariances$$
$$\beta_i \ldots parameters\ or\ regression\ coefficients$$
$$\varepsilon \ldots perror\ rate$$
(4-5)

In comparison, GLMs enable distributions of error components of the exponential family, such as normal, binomial, Poisson, geometric, negative binomial, exponential, gamma, and inverse normal distributions, [122], [124]:

$$g_{(\mu_i)} = g\left[E_{(y_i)}\right] = \boldsymbol{x}_i{}'\boldsymbol{\beta}$$
$$\boldsymbol{x}_i{}' \ldots vector\ of\ regressor\ variables\ or\ covariances$$
$$\boldsymbol{\beta} \ldots vector\ of\ parameters\ or\ regression\ coefficients$$
$$\mu_i \ldots mean\ value$$
(4-6)

Thus, every GLM includes a response variable distribution or error structure y_i, a linear predictor n_i that involves the regressor variables or covariates $\boldsymbol{x}_i{}'\boldsymbol{\beta}$, and a link function $g\left[E_{(y_i)}\right]$ connecting linear predictor with natural mean μ_i of the response variable as so-called canonical link. This definition of GLMs enables the following distributions of error components, [122], [124]:

$$n_i = \mu_i\ identity\ link\ for\ normal\ distr. \tag{4-7}$$

$$n_i = \ln\left(\frac{\mu_i}{1-\mu_i}\right) logistic\ link\ for\ normal\ distr. \tag{4-8}$$

$$n_i = \ln(\mu_i)\ log\ link\ for\ normal\ distr. \tag{4-9}$$

$$n_i = ln\left(\frac{1}{\mu_i}\right)\ reciprocal\ link\ for\ poisson\ distr. \tag{4-10}$$

$$n_i = \frac{1}{\mu_i}\ reciprocal\ link\ for\ gamma\ distr. \tag{4-11}$$

The last stage in the approach represents prediction of development data and combination with field data. The first step enables a prediction of fault rates based on GLM of development data to allow a prognosis of field use periods. The next step is comparison of this data to real failure rates in the field. This can be done by using a LM or GLM enabling an equation to describe the correlation between development and field data. Introducing this method as self-learning approach represents an early warning system for development to support release-related decisions. The more data or projects are analysed by this method the more precise is the prediction and interpretation of results.

The presented methodology in chapter 4 is a new, generic approach to introduce KPIs by analysis methods and stochastic investigations in different phases of the PLC. This concept can be applied in several industrial areas. An exemplary application in automotive industry is presented in chapter 5.

5 Automotive application

The following content incorporates and extends the research of [25], [41], [45], [47], [48], [49], [50], [52], [57], [130], [136], [177], [191], and [193].

5.1 Mechatronic system

5.1.1 All-wheel drive system

The increasing number of mechatronic systems in automotive industry delivers several possible applications of the developed analysis methods. All-wheel drive systems represent one good example for an automotive mechatronic product including mechanics (e.g. clutch), E/E (e.g. senor, actuators), software (e.g. functional software), and their connections. In this case, special focus lies on the automotive software component which represents the whole controlling and functionality part of the system. This system distributes the torque depending on the powertrain system and driving strategy to front or rear wheel to enable four-wheel power, as shown in Figure 5-1.

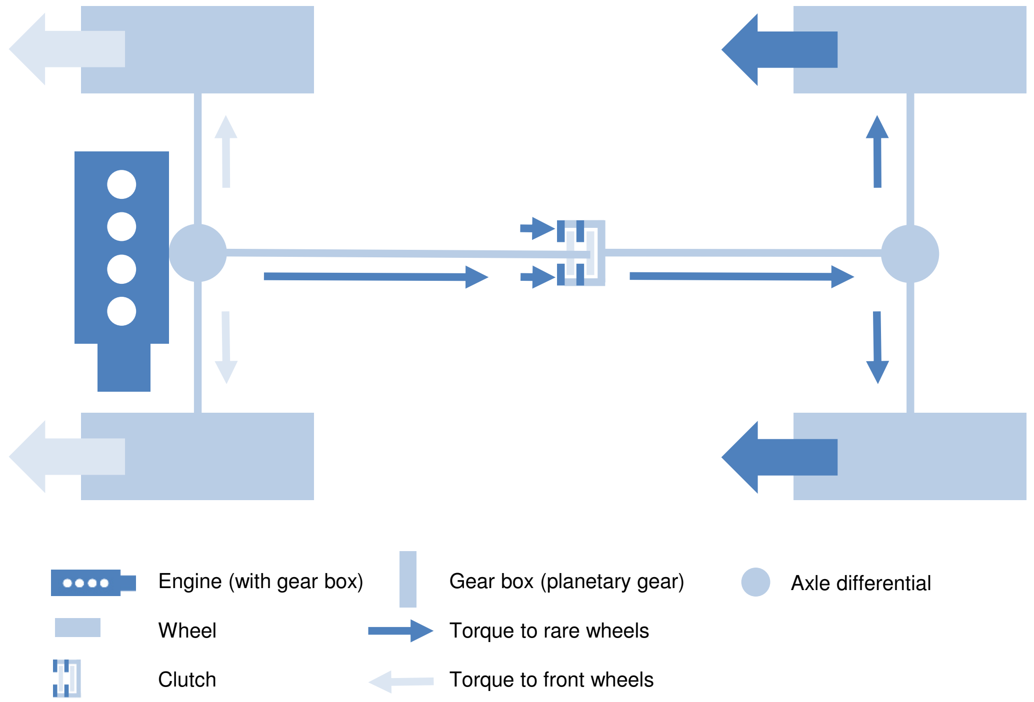

Figure 5-1: Exemplary all-wheel drive system concept with variable torque distribution using a clutch, c.f. [58], [123], [142]

This drive system provides various advantages such as improved traction, positive influence to drivability, and safety. Disadvantages are for instance more weight, more losses (e.g. from oil or friction), as well as more fuel consumption and CO_2 emission, [109], [133]. There are different kinds of systems for various use cases and strategies available. In general permanent or variable/switchable systems are differed that use clutch or limited slip differential, [64], [123].
Detailed investigation of different all-wheel drive systems can be found in [191].
Current all-wheel drive systems use E/E parts and software to enable a precise control of safety and drivability. Thus, these systems include mechanical parts, ECUs, sensors, actors, and automotive software, [95]. ECU represents the central control and communication system for the all-wheel drive systems. This system processes all different signals and control mechanisms as well as connection to other systems of whole vehicles, [146].

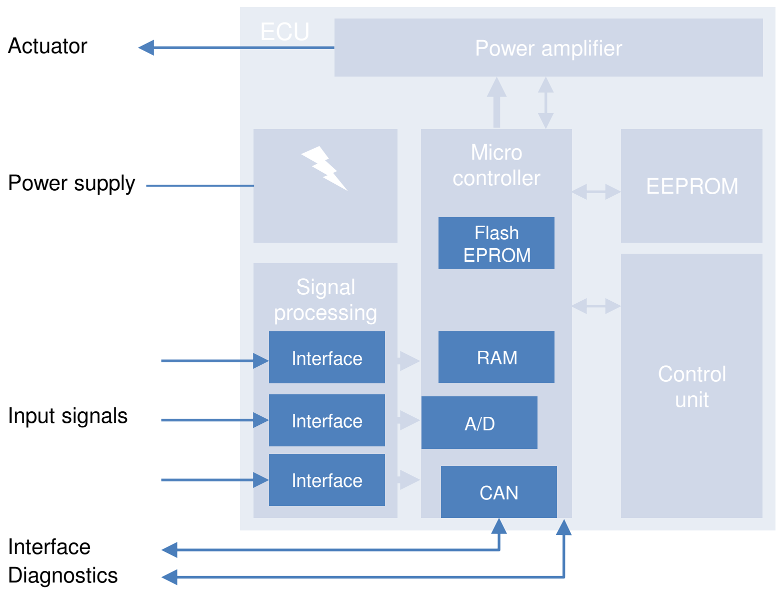

Figure 5-2: ECU of all-wheel drive system, c.f. [146]

Microcontroller represent systems including automotive software with input of signals and output of component controller. EEPROM includes DTCs which are available by on- and off-board diagnosis. The interface for integration and communication in the overall vehicle system is enabled via bus (CAN or Flexray) and a control unit checks the correct processing, [123], [146]. The actuation of the system can be done actively (electromagnetic or electrohydraulic), semi-actively (electromagnetic or hydromechanic), and passively (viscosity or hydromechanic). Within the automotive application, the actuator is electromagnetic using an electrical servomotor and other mechanical parts such as lever, and shift collar to control the clutch mechanism, [142].

5.1.2 Automotive software

The automotive software to control all-wheel drive systems represents an important part of the application of the developed methods within this thesis. The architecture of the system software follows the standardisation of automotive software architecture by AUTOSAR [3]. The architecture is split into basic software, functional software, and run time environment. The basic software provides standardised services such as system (e.g. diagnostic protocols), communication framework (e.g. CAN or Flexray), operating system (static task definition etc.), and microcontroller, ECU, or CDD abstraction (access to microcontroller, ECU, and complex devices). The application or functional software layer includes the whole functionality of the system. In the application of all-wheel drive system, this automotive software component contains all functions for controlling the actuation of the clutch, such as vehicle controller (VC) or component controller (CC). The runtime environment (RTE) enables communication of software components and provides an interface for inter-ECU and intra-ECU communication, as show in Figure 5-3. The aim is to deliver a standardised interface, a connection between functional and basic software through runtime environment and a flexible implementation of different software applications, [3] , [52], [95], [149], [156].

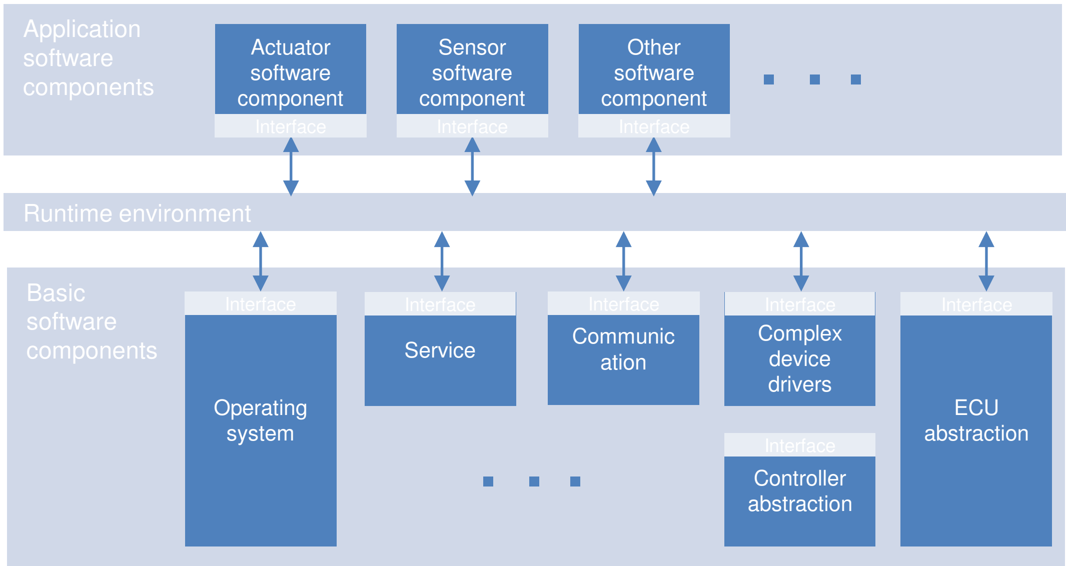

Figure 5-3: Exemplary automotive software architecture, c.f. [3]

5.1.3 Interaction of all-wheel drive components

The control process of this automotive application acts like a mechatronic system. Usually the control unit calculates the manipulated value for the actor. After the process (e.g. mechanical) a sensor controls the output value by measuring (considering disturbances) and delivers a feedback and comparison to the set value including changing (considering error), [90]. In case of a clutch systems are available that only steer the actor without feedback of sensor due to a given torque distribution based on mathematical models, as shown in Figure 5-4. The clutch and vehicle signals represent the input for the calculation of the torque request by the vehicle controller. The output of the component controller is the actor steering based on different models and routines, [58], [142]. The communication in the system is limited to component controller and actuator, [95]. The interaction of the system with other vehicle components is bus-related, [149]. Various signals influence all-wheel drive systems in the overall vehicle.

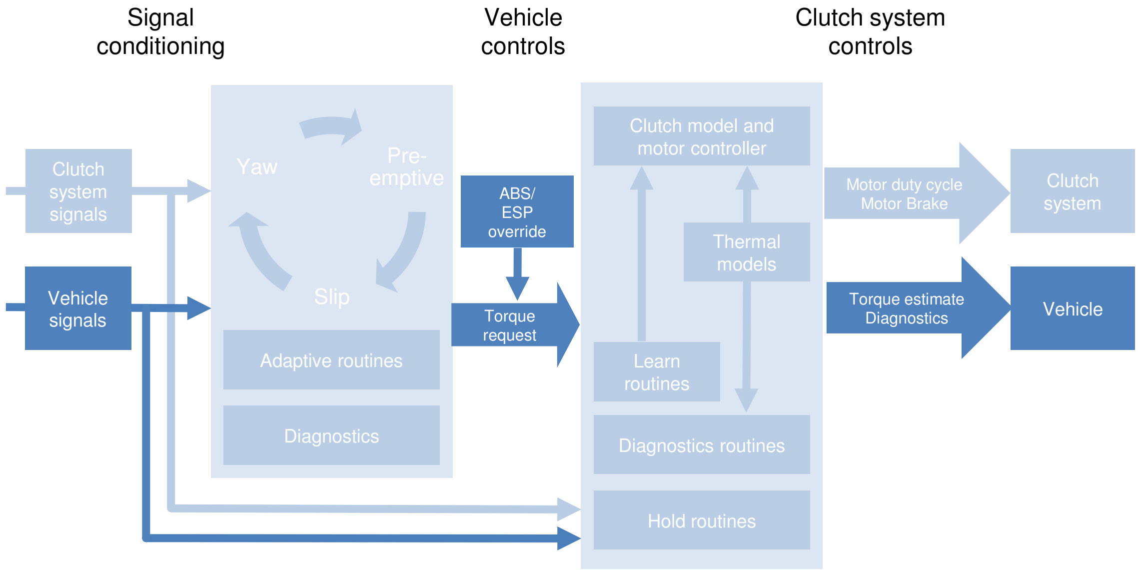

Figure 5-4: Active torque control, c.f. [142]

5.2 Application of the comprehensive analysis method

5.2.1 Comprehensive analysis method for all-wheel drive systems

The application of the proposed comprehensive analysis method on all-wheel drive systems is shown in Figure 5-5. The use of this method enables a detailed investigation of all subcomponents and their interactions. In Figure 5-5 the upper level of the whole system with the environmental in- and outputs, such as CAN, chassis, and battery, is depicted. In the next level, the system is divided into an ECU, mechanics, and cabling. Level 3 includes further details of subsystems, such as dividing ECU into HW and SW, or actuator and sensor, mechanics, and motor. The complex interaction, signal flows, and connections between all components can be visualised and investigated through a higher detail depth. Even more details can be generated by additional levels, [47], [48], [51], [161].

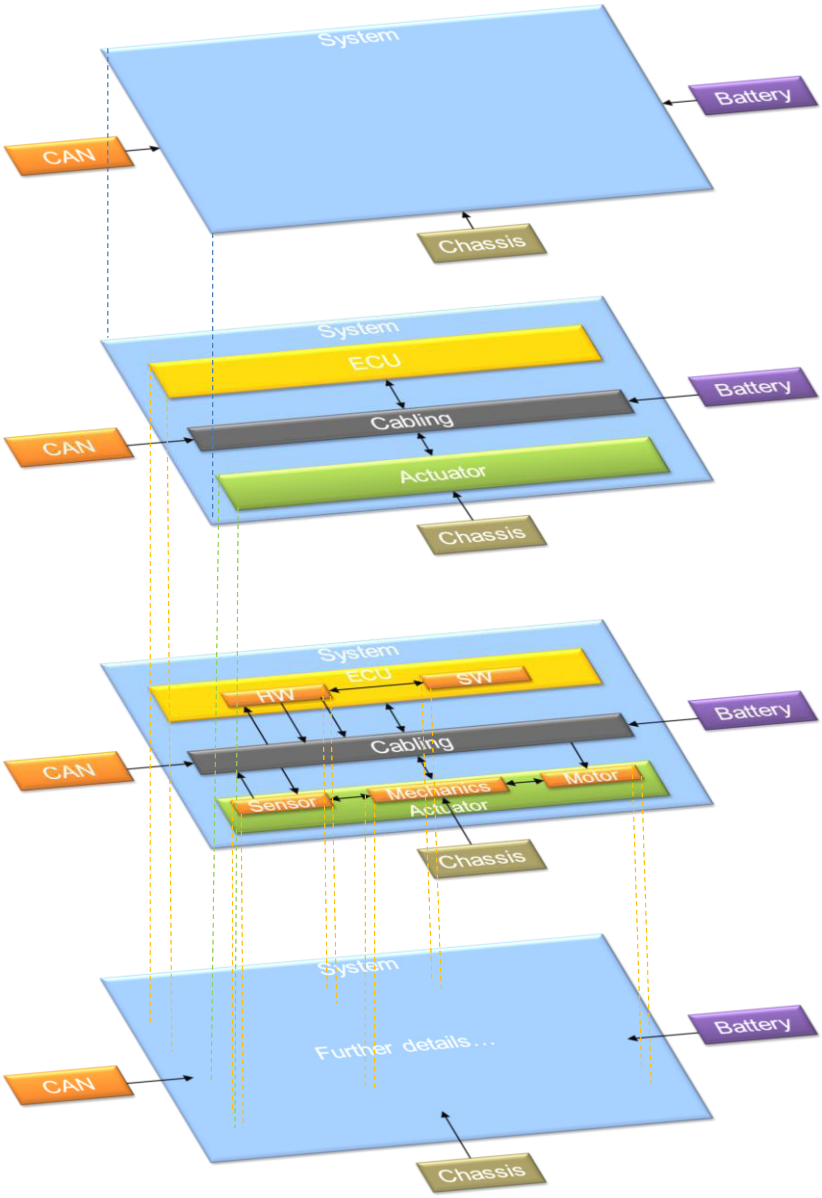

Figure 5-5: Innovative analysis method applied on an all-wheel drive system c.f. [47], [48], [51], [161]

5.2.2 Extended investigation of faults, signals, and flows

Using this method for all-wheel drive systems enables an investigation of different connections and relationships, such as signal- or data flows and fault propagation paths. Above all the optimised visualisation of fault propagation paths allows a detailed investigation of possible faults. Due to interactions of components, various failures and failure types can occur in this mechatronic system such as expected, not expected, differing in time of occurrence, related to various causes, or additive or multiplicative failures, [90].

Potential failure sources of all-wheel drive systems can be:

- Mechanics:
  Nowadays, the main part of the mechanical system is developed over many years or product generations including testing, calculation, and experience. Therefore, the failure rate related to these subcomponents is usually low and related to tiredness, wearing, corrosion, overloading, failures due to montage, production, service, or development, [123]. Mechanical failures are often combined with huge impacts on functionality or breakdowns of the systems. For this purpose, every produced mechanical part of the all-wheel drive system has to pass an end-of-line test (EOL) to preclude failures due to assembly and production, [123].
- E/E:
  Usually E/E components of all-wheel drive systems show a similar behaviour of faults frequency as mechanics. Fault causes for this category can be defect sensors, loss of power supply, failure of individual E/E components (e.g. cabling), electromagnetic disturbance, vibrations, corrosion, humidity, and failure current. Some of them represent temporary failures, while others lead to impacts on functionality or breakdowns, [147].
- Software:
  Automotive software components of these systems are often related to faults during development due to time and cost pressure. These failures are often difficult to investigate and can be caused by the connection of different software modules of various development companies or divisions. Special challenges occur because of the connection with hardware parts and pressure in terms of correct test specification. In addition, the system cannot be tested against every situation due to time and costs. Failures in these subsystems are related to development, specification, coding, design, verification, integration, and interfaces, [156].
- Interaction within the integrated vehicle:
  The all-wheel drive system as part of various mechatronic systems within an integrated vehicle interacts with several units in form of signals, flows and electricity. Due to these various impacts failure potentials increase, such as false architecture, incorrect integration, wrong signals, sources of disturbance, or transmission errors, [148], [156]. The most important issues are the following, [150]:
  - Defective configuration data:
    Defective configuration data are errors in data including configuration, such as engine characteristic curve or gear ratio.

- Defective value range:
  This effect appears if values of signal sources are not in range, e.g. negative values or wrong physical values.
- Defective hardware:
  This effect can appear because of increasing hardware functionality, e.g. defect sensors, breakdown of ECUs, or overloaded EEPROM.
- Defective information order:
  Wrong or incomplete order of information can lead to defects in signal communication.
- Defective task order:
  Wrong or incomplete order of tasks can influence the calculation or control of components.
- Defective tests:
  Wrong calculation and specification of tests can produce defects.
- Defective bus use:
  Wrong communication via bus, such as errors in bit occupancy influence data processing of ECUs.
- Defective synchronisation:
  Simultaneous exchange of data on bus systems has to be controlled by priority systems. Defects lead to impacts on the whole communication process.
- Defective relations:
  Wrong relations of components affect data and communication flows, such as errors in priority relation.
- Defect data flow:
  These effects are impacts on data flow, e.g. delay or electromagnetic disturbance can lead to wrong signal flows and errors in actuation or processing.

For further research on investigation of faults and signals of all-wheel drive systems see [191].

5.3 Application of analysis methods using KPIs on basis of development and test data

5.3.1 Initial situation and development process following V-model

The development process of an exemplary automotive application follows a modified version of the V-model on basis of standard VDI 2206 [183] for mechatronic systems. Thus, the beginning of the development process contains requirements and specifications to define the functional range of the system. The step of the development process is split into interdisciplinary subareas of the mechatronic system, such as mechanical, electrical, and software development areas. Hence, the left side of the V-model shows the design of the system, over subsystems to component level and the right side includes different types of testing procedures, e.g. MiL, SiL, HiL, and integration tests within the complete mechatronic system. This process and the stored information delivers the basis for new analysis methods during the development process, such as information and data of requirements, planning elements, testing, and failure handling, [49], [50], [52]. Focusing on the application of developed methods on the automotive software part of the all-wheel drive system, the initial software architecture follows the AUTOSAR standard (see Figure 5-3). Basis for the development is a modified V-model including a mapping of so-called system component requirements (SCRs) to test cases (TCs) on the integrated software level and so-called module requirements (MRs) to test cases (TCs) on module level. The so-called change issues represent planning elements over the whole development process including software faults by definition of the category "findings", [45], [99].

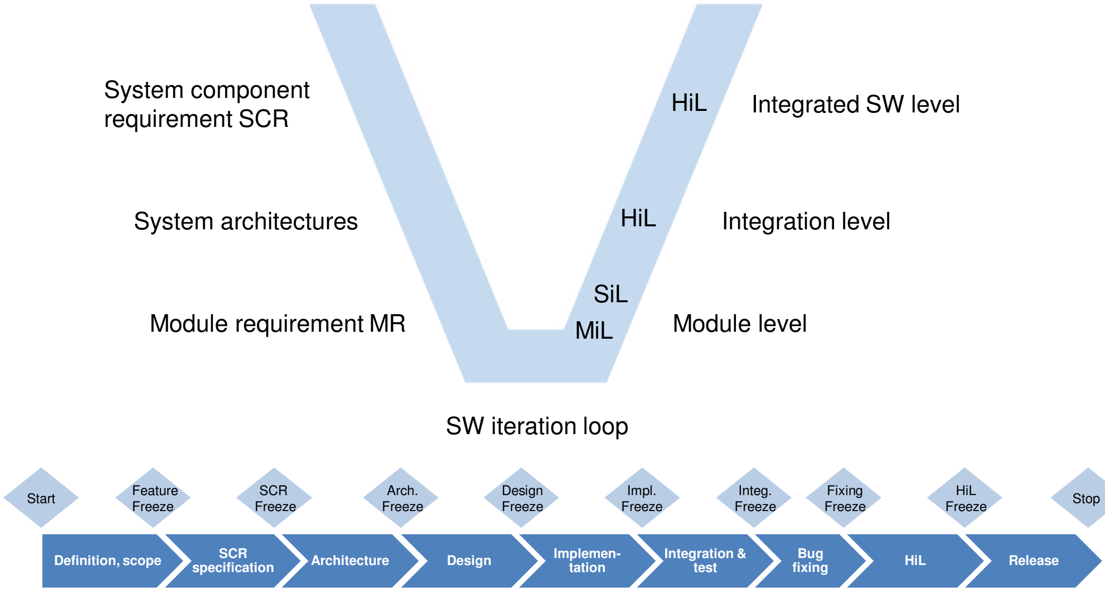

Figure 5-6: Modified V-model of initial situation, c.f. [99]

SCRs, MRs, CIs, and TCs are connected to each other, see Figure 5-7.

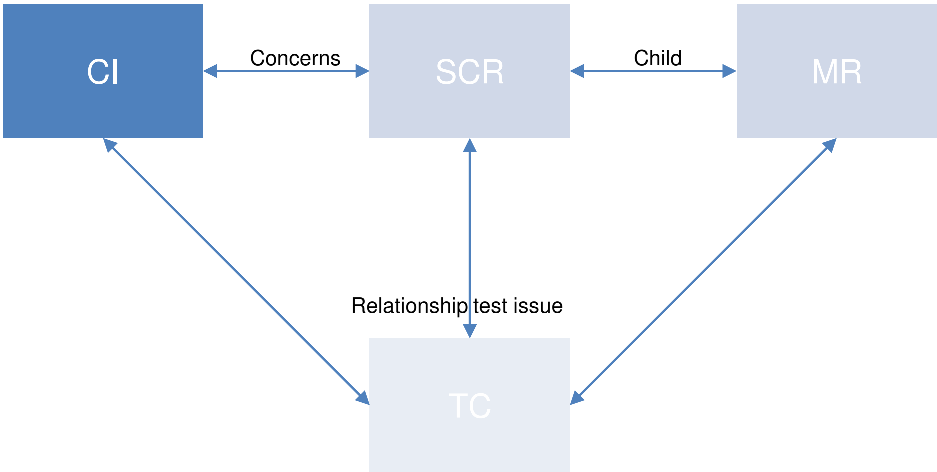

Figure 5-7: Connection of items, [45] c.f. [99]

SCRs and MRs are used to specify requirements and functions at system and module level. Due to the defined workflow, they can have a status such as new, specified, implemented, and closed. TCs are used to test the system for specifications and requirements. Due to the defined workflow, they can have status such as new, specified, in work, completed, retest, failed, completed with restriction, and closed. CIs are used to define and plan all tasks, such as implementation of functions or fixing of bugs. Due to the defined workflow, they can have status such as new, accepted, CCB OK, CCB NOK, rejected, planned, failed, implemented, tested, completed, and closed (for detailed information see [45] and [41]). Mapping this initial situation to the developed methods is shown in Table 5-1.

Table 5-1: Mapping generic analysis methods and the application situation

Generic analysis methods	Application situation
Requirements	System component requirements SCRs module requirements MRs
Planning elements	Change issues CIs
Tests	Test cases TCs
Faults	Change issues (category "findings")

A milestone list represents the planning and controlling tool in development and includes the following attributes, [45]:

- Software release:
  Software releases represent a seven-digit code including one major, three minor, and three patch numbers separated by a dot. Several items are related to this number to plan development processes.

- Target date:
  Target dates are always related to a specific software release and represent the points of time where all planned items related to this specific software release have to be fulfilled.
- Milestone:
  Milestones are internal and external declarations of important development and decision points. For internal iterations, milestones can also represent freeze points during the development process.
- System release:
  System releases represent the most important part in the development process, because the whole system or parts of them are delivered to the customer depending on the release level at this point of time.
- Project:
  The project represents the development project.
- Priority:
  Priority is used to define if a milestone or release represents a major release, minor release, or patch milestone.

All different items in combination with the milestone list deliver the available data basis during the development and test process. During the development of automotive software different parties are involved, such as general management, quality assurance, project management, and process management. All of these divisions have different views and questions during the development process which have to be satisfied and answered by introducing analysis methods. A general overview of the most important questions follows, c.f. [49], [50], [52], [100]:

General management

- Status and progress:
  - What is the development status of all projects?
  - What is the development status of a specific project or iteration?
  - What is the current milestone status (content, time, quality)?
  - What deviations of projects or iterations occur?
  - For how long are tasks in process or not in process?
  - Are defined process rules and workflows met?
- Trends:
  - How does the tendency of project control look like?
  - How does the trend of freeze points look like?
  - Are freeze points met?
  - Are trends of completed and open planning elements contrary?

- Risk:
  - Do software findings appear?
  - Can software findings be solved?
  - What is the risk of specific projects or software findings?
  - Are development teams able to react and handle deviations?
  - What are effects and risks of unknown faults?
  - What is the risk of unfulfilled planning elements?
- Release decision:
  - What is the residual error rate at a certain point of time?
  - Is a product release possible or is further bug fixing needed?
  - Are there findings after releases?

Project management

- Status and progress:
  - How many tasks are going on and what is their status at a certain point of time?
  - Can contents and milestones during development loops be met?
  - Are items in an adequate status at certain milestones?
  - What is the status of planning elements or working packages?
  - How long are planning elements in process or not in process?
  - How many planning elements are in process compared to new ones?
- Critical elements:
  - What are critical, difficult, and best topics or phases?
  - What are effects on further releases of open points?
  - How do time approaches of critical topics look like?
- Deviations:
  - Is it possible to solve tasks in time?
  - How often can milestones be met?
  - Do delays or rescheduling appear?
  - How often are tasks rescheduled?
  - What are deviations of a specific project?
- Trends:
  - What is the trend of open and fulfilled planning elements?
  - How is the project team performance?

Process management

- Processing:
  - How long are current processing times?
  - Which processing times are relevant?
  - What are processing time boundaries?

    - What are realistic processing time boundaries?
    - What is the behaviour of processing times in relation to project maturity?
    - What are challenges during project phases and processes?
    - How do trends in processing look like?
    - How does the testing procedure and coverage over product maturity look like?
    - How many planning elements are performed compared to the whole project content?
    - How does the implementation process look like?
  - Process flow:
    - How does actual process flow look like?
    - What are optimisation potentials of process flow?
    - Is there a need of additions?

Quality assurance

  - Performance:
    - How does target-performance comparison look like?
    - How does target-performance comparison over the project maturity look like?
    - Are freeze points fulfilled? If not, why not?
  - Product quality:
    - How is the product quality?
  - Process quality:
    - How long are planning elements not being processed?
    - What is the trend of new and completed planning elements?
    - What is the connection grade of items?
    - How many reviews are performed?
    - How many faults are detected compared to implemented functions?

5.3.2 Analysis methods using KPIs on basis of development and test data of automotive software for all-wheel drive systems

The application of new developed analysis methods on example of automotive software development for all-wheel drive systems enables an optimisation of processes and quality. To satisfy all needs of several development divisions, Figure 5-8 shows a categorisation of questions by statements and their allocation to perspectives and KPIs delivered by the new analysis methods. These methods are introduced by a programmed tool which can directly be used in development processes. The concept of the applied tool, which enables various analysis methods to deliver KPIs for different perspectives, is shown in Figure 5-9. On the one side, the applied tool is based on the concept of enabling an overview of all projects and tasks as well as to increase details for specific project investigations. On the other side, the complexity of analysis methods increases from general statistical overviews over comparison and calculation of various values to stochastic analysis and modelling. The aim is, besides optimising development processes and quality of products, to deliver defined indicators for a risk analysis for product release decisions.

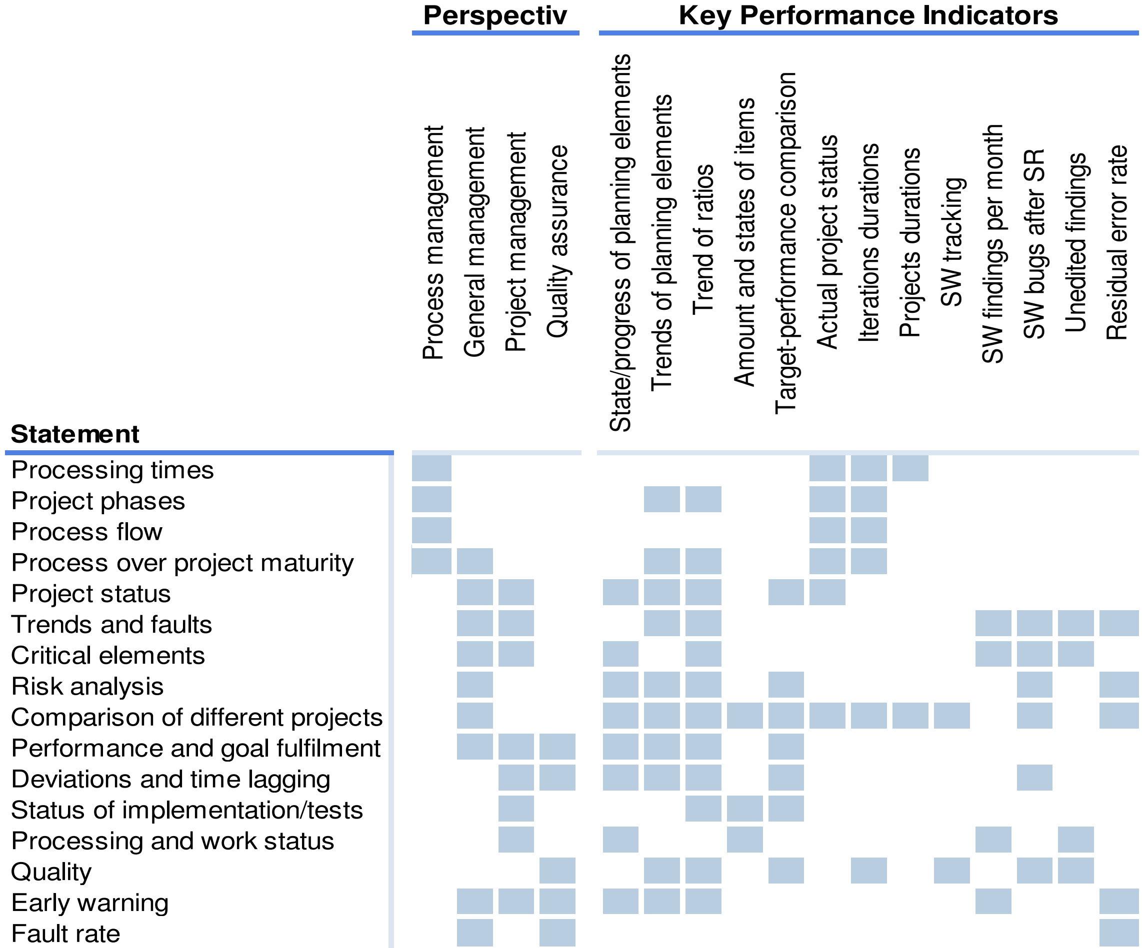

Figure 5-8: Allocation matrix of development questions, perspectives, and KPI-related analysis methods, c.f. [45]

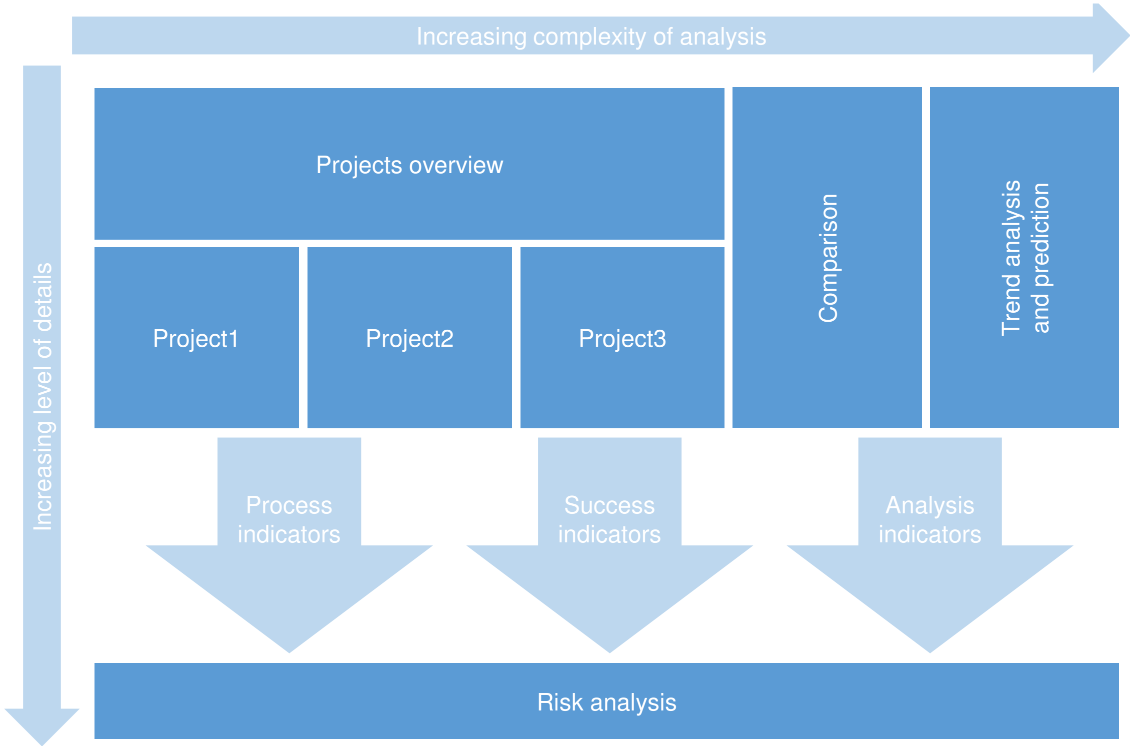

Figure 5-9: Basic concept of the applied KPI management tool c.f. [45]

5.3.2.1 Analysis process for general management

Status and progress

Analysis methods for status and progress in combination with tool concept architecture enable checking KPIs on the development status throughout different projects. For this purpose, the process of getting all necessary information for the general management to control status and progress starts with an overview of projects (see Figure 4-14). Furthermore, the KPI management tool enables controlling of the status of all items related to specific projects (see Figure 4-11). Due to implemented filters, status of items and comparison of items in work can be investigated for all projects or for a specific project. If general management needs more details of a specific progress of a project, the analysis method of "iteration durations" allows controlling each individual software iteration, see Figure 4-16. Finally, the status and progress can be analysed by using the "target-performance comparison". KPIs of this method enable control as completion degrees over project maturity and deliver the statement about ongoing progress and quality of products as well as information about the status at important milestones (see Figure 4-17). They also show the possibility to investigate deviations of a project. The method "trend of planning elements" in combination with the KPIs related to gap analysis between target of tasks and the performance allows a more detailed analysis (see Figure 4-19 and Figure 4-20). If necessary, a more detailed control of iterations of a specific project is possible by using specific KPIs of the method “iteration durations” (see Figure 4-16). If the general management wants to know how long different tasks are in progress or not, the “status of planning elements” (see Figure 4-12 and Figure 4-13) and “unedited findings” (see Figure 4-25) help to investigate the work process of tasks. The investigation of the possibility of project teams to fulfil defined working processes starts with checking deviations in combination with a detailed investigation of the “target-performance comparison” and “iteration duration”. This shows if milestones and freeze points are met in time and content.

Trends

Very important information for general management is generated by checking trends of development processes. Hence, the KPI management tool delivers several KPIs by analysis methods to enable an investigation of project tendencies. This can be done by checking the KPIs considering the actual project status and time schedule over all projects (see Figure 4-14). After this, different trends can be analysed starting with “trend of planning elements” (see Figure 4-19 and Figure 4-20). The next step is to analyse trends of important KPIs, such as ratios (see Figure 4-21), which deliver an estimation of development, test, and bug fixing tendencies. Finally, the trend of detected faults and residual error rate enables a statement about the fault situation in a specific project. The investigation of trends at milestones or freeze points can be done by using KPIs generated by the methods “target-performance comparison” and “iteration duration”.

Risk

The general management always has to be informed about possible risks of development processes. Risks in development are related to KPIs considering faults, bug fixing process, and critical elements. Thus, the analysis method “trend of new and solved faults” shows the comparison of the KPI detected faults and the process of fixing them (see Figure 4-22). A further investigation of faults and fixing progress can be done by checking the distribution of faults over development time or software releases (see Figure 4-24, Figure 4-28, and Figure 4-29). To get a statement about the risk of specific planning elements and not solved tasks, a detailed investigation using “status of planning elements” is necessary. This method delivers the data for predefined KPIs related to the criticality of different tasks as well as SIL classification to show if a safety-critical situation is coming up (see chapter 4.3.3.1). To check if the development team is able to handle critical situations is a basis for decision-making of general management. Hence, the combination of statements mentioned above and the methods “software bugs after system release” or “unedited software findings” represent an approximation if the situation can be controlled by the development team (see Figure 4-23 and Figure 4-25).

Release decision
The most important information for general management is whether a product release is possible at a certain milestone. Besides the mentioned tasks above, the KPI management tool allows checking the status at various release points by using the KPIs of the “target-performance comparison”. Additionally, the analysis method “trend of planning elements” enables the investigation of KPIs considering the gap between planned and performed tasks at a specific release. “Software bugs after system release” allows checking the KPI whether there are any unforeseen risks due to bugs in the system. Finally, for the investigation of the residual error rate using NHPP and confidence interval estimation delivers a statement about how many faults will remain in the system at a certain release point, as shown in Figure 4-28 and Figure 4-29.

5.3.2.2 Analysis process for project management

Status and progress
Project management requires information about status and progress of specific projects in their area of responsibility. Thus, the KPI management tool enables an overview of the status of all items related to a relevant project (see Figure 4-11). The main KPIs for project management come from the method “status of planning elements” (see Figure 4-12 and Figure 4-13). The algorithm investigates the amount, processing, and status of tasks or working packages of a specific project. The question whether contents of a project are met at milestones or freeze points and whether items have an adequate status can be answered by using the methods “target-performance comparison” and “iteration durations”. Project management also needs knowledge about the working process of planning elements as well as a comparison of trends of new and solved faults. This information can be provided by checking the status and target date using the method “status of planning elements” and by investigating KPIs related to faults using “trend of new and solved faults”.

Critical elements
Project management has to control the whole process and progress of a specific project. Thus, special attention has to be paid to critical elements. For this purpose, the analysis method “status of planning elements” (see Figure 4-12 and Figure 4-13) enables the control and classification of working tasks. Due to criti-

cality, SIL classification, rescheduling, looping, and not solved items, critical elements can be filtered and different circumstances can be analysed (see chapter 4.3.3.1).

Deviations
Controlling the progress of projects is very important to meet the schedule and save unforeseen costs. For this purpose, the KPI management tool supports project management to investigate the progress and enables an early warning system about possible deviations. The main KPIs for this use case are not solved planning elements, planned and fulfilled software release, and amount of loops enabled by the analysis method "status of planning elements". With these parameters, a delay or rescheduling can be detected as soon as possible. This information can be included in the further development process. Furthermore, the analysis methods "iteration duration" and "target-performance comparison" can be used to check if there are some deviations in the progress of projects due to not met milestones or freeze points.

Trends
Trends provide a good preventive analysis for project management. Hence, the KPI management tool enables the investigation of project tendencies. First steps for this investigation are to control the project team performance by checking the actual project status (see Figure 4-14) as well as analysing the trend in development by using the analysis methods "iteration durations" and "target-performance comparison" (see Figure 4-16 and Figure 4-17). Next step is to investigate the project saturation curves. This can be done by analysing ratio KPIs using the analysis method "trend of ratios". This delivers a statement about how early requirements, planning elements, tests and faults are completed over project time line and if the team performance is high enough to meet project goals. The last step represents the investigation of faults and bug fixing by using analysis method "modelling and residual error rate" for checking, if these trends are satisfying.

5.3.2.3 Analysis process for process management

Processing
The analysis of processing, times, and boundaries is very important to see if current processes work well to plan realistic schedules of further projects. The analysis method "project duration" delivers durations of all relevant projects as KPIs. This can be used to get a feeling of average project durations to allocate tasks and resources. Analysis methods "status of planning elements" and "iteration durations" enable a more detailed investigation by delivering different KPIs, such

as duration of freeze points, duration of work packages, and number of loops. These objective parameters provide a basis for defining realistic project or task durations and boundaries.

Process flow

In addition to the above-mentioned methods, the behaviour of processing times and flows in relation to project maturity, investigation of implementation process, as well as detection of challenges during project phases and processes can be analysed by using the “target-performance comparison” and “trend of ratios”. Furthermore, KPIs of in work and done items over different projects detected by “overview status of all items” allow various investigations of how process flows can look like. To get the statement about test and fault process flows, the analysis method “modelling and residual error rate” provides suitable information.

5.3.2.4 Analysis process for quality assurance

Performance quality

The quality assurance uses the analysis method “target-performance comparison” to control the performance and quality of projects. A more detailed investigation and check of freeze points is enabled by the analysis method “iteration duration”. The main KPIs of these methods are degrees of fulfilment, such as requirements, tests, fixed faults, tasks, or specific fulfilments of requirements at freeze points.

Product quality

A very important point for quality assurance is to evaluate the product quality. For this complex topic, various analysis methods can be used to deliver a statement. For instance, “target-performance comparison” delivers the data for predefined KPIs of maturity and implemented functions, whereas “trend of ratios” provides information about implementation process and saturation function. The method “faults per month/software release” delivers detected or fixed faults and “modelling and residual error rate” defines how many faults are remaining in the system. A combination of these methods and several KPIs deliver a basis for product quality evaluation.

Process quality

Quality assurance has to control processes and compliance during development. The control of processes in project can be done by using “status of planning elements”, “trend of new and completed planning elements”, and “unedited software findings”. These methods enable different KPIs to check if processes are passed through correctly by delivering information, such as counter for loops, meeting of target points, rescheduling in form of releases, comparison of bug fixing trends, or number of unedited findings. The analysis method "tracking" represents another important way to check the compliance of development teams to processes of a company. This method delivers the data for predefined KPIs which represent a mix of data management, process fulfilment, and connection quality (e.g. the KPIs number of reviews and percentage of connected elements). Furthermore, this method allows checking the fulfilment degree of requirements on subcomponent level (see Figure 4-18).

5.3.2.5 Excursion software metrics

In case of automotive software development, a further efficient opportunity to evaluate systems and processes is to use so-called software metrics. Hence, several static and dynamic metrics can be calculated to support quantification and comparability of software development. To provide effective investigation software and process metrics, metrics have to be considered for the analysis. For further research on this topic see [41] , [44] and [49] delivering an aggregation of individual metrics into superior categories by using weighting factors such as quality, quantity, and complexity. Hence, a suitable visualisation is advisable to allow a comparison of different software releases and development progresses. The decision, which of the huge number of metrics has to be used, is difficult and related to significance of the statement to satisfy needs. A suggested combination of software and process metrics allocates different metrics into three superior categories, [44], [49]:

- Quality:
    - Percent test coverage
    - Percent density of comments
    - Percent planning elements done
    - Percent faults solved
- Quantity:
    - Lines of code
    - Number of functions
    - Number of planning elements done
    - Number of tests done
- Complexity:
    - Cyclomatic complexity
    - Number of function parameters
    - Number of calling levels

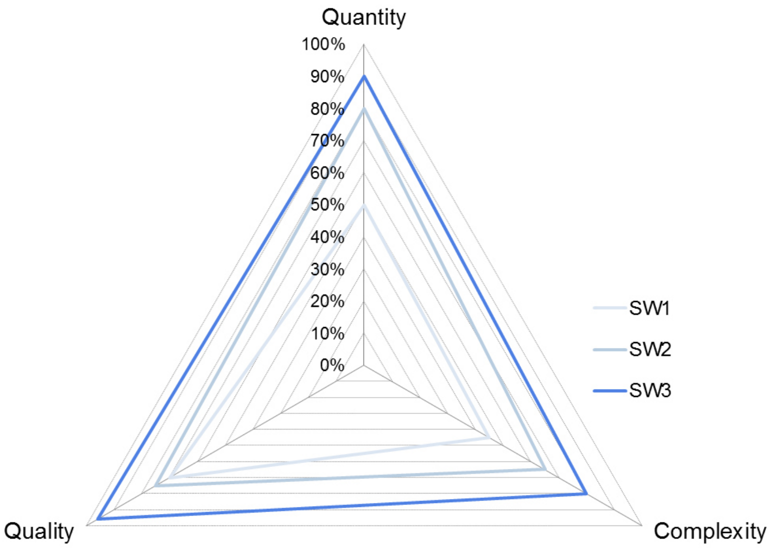

Figure 5-10: Exemplary application of software metrics in combination with process indicators, c.f. [44], [49]

The visualisation of Figure 5-10 gives an overview of relative modifications using different characteristics. A spider chart enables a relative comparison of software releases showing software and process metrics in the categories quantity, quality, and complexity over project or release maturity, [44], [49].

5.4 Application of analysis methods using KPIs on the basis of field and complaint data

5.4.1 Initial situation and database

The initial situation of the exemplary project is based on the field and complaint data of all-wheel drive system from several sources. The use of data sources is related to the possibilities of a company to enable this information. The following figure shows possible data sources at the application of an all-wheel drive system in automotive industry. Furthermore, there are three main interest groups for analysis methods of field and complaint data with different views or needs, such as engineering, complaint investigation, and quality assurance.

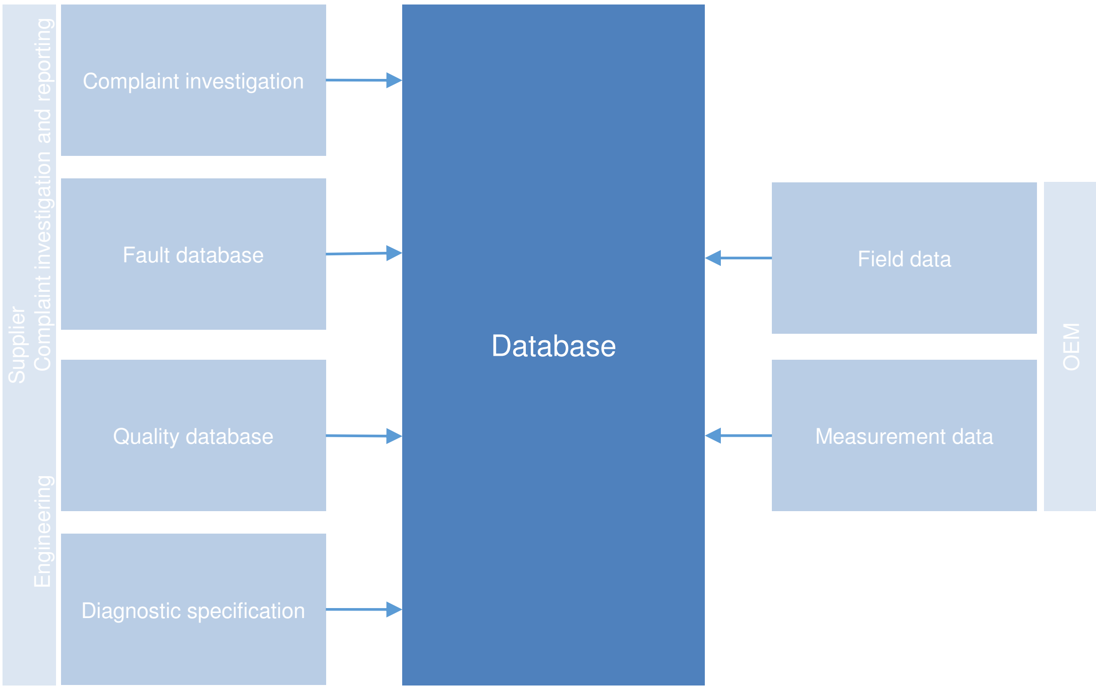

Figure 5-11: Data sources related to field use, c.f. [130]

The data sources derive from three divisions:

- Engineering supplier:
  - Diagnostic specification:
    The diagnostic specification for relevant board net delivers a categorisation and further information for DTCs (diagnostic trouble codes), such as failure ID, failure type, storage type, and description, repair measurement.
  - Quality database (QDB):
    The QDB delivers a communication platform between OEM and supplier. The reports allow documentation, storage, and exchange of information such as project, series number, VIN, complaint number, complaint reason, production date, and mileage.
- Complaint or diagnostic investigation supplier:
  - Complaint investigation:
    The investigation of complaints is based on tools and expert knowledge to enable precise information of the system. In the automotive application, the results are txt-files with detailed information of the aggregate or system and their failure codes such as complaint number, complaint reason, aggregate number, aggregate type, VIN, optical defects, defects of devices, failure codes, software releases, and wear out data.

  - Fault database:
    The complaint and diagnostic investigation division stores various failures and complaints in a fault database to extend knowledge. This type of data includes information about complaint time, readout time, aggregate, failure cause, repair measures, further expert information etc.
- OEM:
  - Field data:
    Field data usually come from service departments, but can also be collected by direct connection to cars. These data represent DTCs with further information due to diagnostic devices (on- or off-board), such as DTC, vehicle type, VIN, motor type, readout time, mileage, and description.
  - Measurement data:
    Measurement data are similar to field data, but usually represents continuous tracking of different measures rather than DTCs. Thus, further information of measure parameters is available, such as software release, temperature, voltage, processing time, wear out, and torque value.

The generic approach to generate a relational database by parser for the automotive application is shown in Figure 4-32. This method is based on an equivalence list and on generic information of a database, see Appendix A-1.

5.4.2 Analysis methods using field and complaint data

5.4.2.1 Analysis process for engineering

Retrospective analysis methods

Engineering needs to have various information about product and system behaviour in the field use. For this purpose, a continuous investigation using retrospective analysis methods enables knowledge of past events and important inputs for further development and projects. The first step is to investigate KPIs considering DTCs of different projects and system types. Hence, analysis methods considering KPIs related to failure code frequencies over field uses of automotive systems based on the connection of field and complaint investigations deliver the right information. The generic database connects all relevant data sources to enable different visualisations of failure code frequencies. An implemented filter option allows the selection of different criteria such as customer, project, data source, vehicle type, aggregate number, production date, period of field use, and failure codes. These filters enable several analysis possibilities. Besides normal investigation of failure code KPIs (see Figure 4-33), engineering has the possibility to analyse faults of specific vehicle types and all-wheel drive systems (see Figure 4-34 and Figure 4-35) by using new developed analysis methods. These perspectives enable various information about the product behaviour for different customers and circumstances. The required information for this analysis method are failure codes, customer, project, vehicle type, aggregate number, production date, and readout date.

The analysis method checking failure frequencies over the use of time of vehicles is very important to investigate the behaviour of systems. This method delivers a statement of how many failures over a time lane or mileage of vehicles appear, as shown in Figure 4-37. Besides the data of previous analysis methods, mileage information and production data are required to enable the calculation of the algorithm. The applied filters support research about lifetime distributions of different systems and customers uses.

Due to analysis possibilities of environmental conditions, the investigation of KPIs considering failures of automotive systems related to countries is interesting. Thus, the additional information of the country where vehicles are sold is required to enable the algorithm summing up failure codes in relation to countries. All of the other information and filters are similar to previous analysis methods (see Figure 4-39).

Another possibility to visualise failure frequencies of all-wheel drive systems is to relate them to vehicle types and different fault categories, e.g. hardware (mechanics and E/E), software, and unspecified failures. This analysis method affects engineering by delivering a statement of fault category occurrence for cause research (see Figure 4-36).

A detailed analysis possibility enables an investigation of SW division, which shows the relationship between software releases and failure codes. For this purpose, further information about the generic database is necessary, such as bootloader ID, software ECU ID, current SW release, and SW release of rollout point. The query sums up failure codes in relation to different SW releases, as shown in Figure 4-40.

In this context, analysis methods deliver KPIs considering changes of SW during field uses are very important to enable knowledge of the customer's product behaviour. Thus, further information of the generic database is necessary, such as bootloader ID, software ECU ID, current SW release, and SW release of rollout point. The query sums up vehicles with specific SW releases and enables investigations for engineering. The first investigation shows how many cars include the SW level of roll out in comparison to cars with a newly flashed SW version, as depicted in Figure 4-41. Additionally, Figure 4-42 shows the progress and saturation of SW flashing in field. This analysis is very important for engineering to investigate the duration of the flashing processes, for instance how long the replacement of a SW with bugs in the field takes. A similar method enables checking the chronological approach of SW implementation on different project generations in the field. It enhances an investigation of SW flashing processes by comparing different projects to include the knowhow in further development.

Predictive analysis methods

Predictive and preventive analysis methods are very important tools for engineering to estimate whether problems are coming up in the field. The necessary data mainly comes from measurement data but also from field data and complaint investigation. The boundaries to check whether critical KPIs are reached or not, are included in the tool and determined by engineering.

The first analysis method enables KPIs related to the wearing process by delivering a predictive statement about a possible breakdown in future. Hence, this preventive analysis method requires information about measurement and complaint data, such as customer, project, VIN, mileage, aggregate number, wear values of oil, clutch discs, and chain. Furthermore, boundaries or other checks are necessary to get comparable parameters or rules. The applied filter allows the investigation of specific customer, projects, vehicles, aggregates, field uses due to mileage, and specific analysis of individual checks. The algorithm calculates wear data in relation to mileage and compares values to boundaries constructing an early warning system.

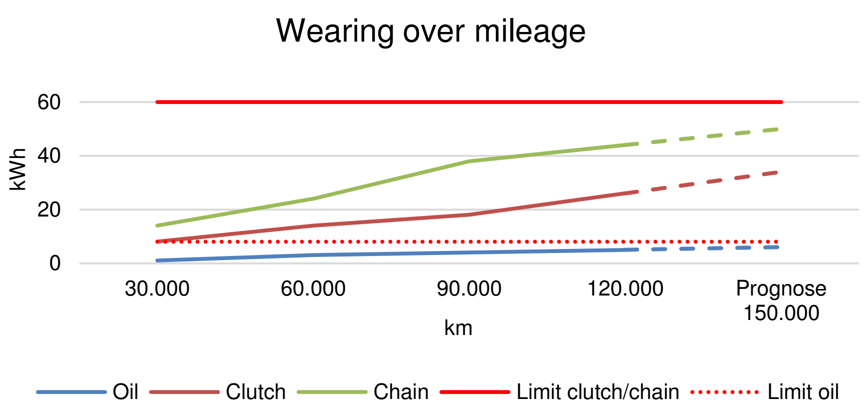

Figure 5-12: Preventive analysis of wearing in field use of an all-wheel drive system

A similar investigation is to visualise the resettable KPIs of wearing. This enables the control of wearing behaviour, for instance between services. To enable a realistic prediction system the estimation of further component behaviour can follow stochastic models (see chapter 3.3).

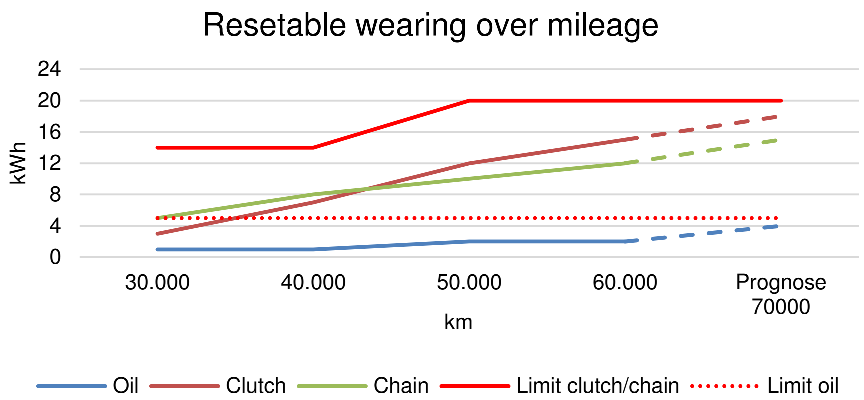

Figure 5-13: Preventive analysis of resettable wearing in field use of an all-wheel drive system

Additionally, further analysis methods deliver KPIs considering use values (such as temperature or torque) by enabling predictive statements of critical values of the system over time. For this purpose, the next preventive analysis method requires additional information about measurement and complaint data, such as values of temperature level and readout time. The applied filters are similar to the above-mentioned. The algorithm calculates the distribution of temperature related to processing time. The comparison of boundaries or checks allows an early warning system (see Figure 4-46).

The next preventive analysis method in this case deals with torque and rotation numbers which – combined with time lane – deliver a statement about the clutch work. Hence, this investigation requires information about measurement and complaint data. The additional information about the generic database includes values of torque/rotation and processing time. In addition, boundaries or other checks are used to compare values or rules. The algorithm calculates the distribution of torque related to rotation per minute and processing time. The result is a three-dimensional diagram which also enables an early warning system to get a statement whether a problem is coming up by checking boundaries.

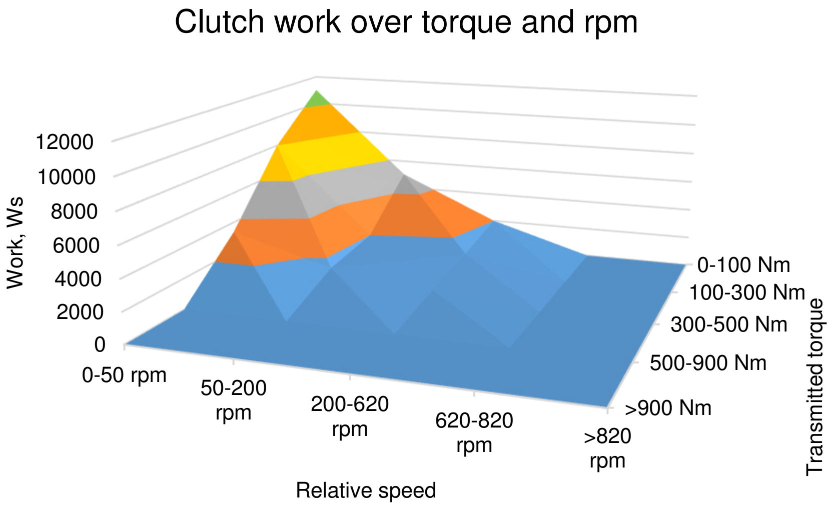

Figure 5-14: Three-dimensional analysis of clutch work over torque and rpm of an all-wheel drive system

5.4.2.2 Analysis process for quality assurance

Quality assurance uses the same KPIs by retrospective and predictive analysis methods as engineering to investigate the product behaviour as mentioned above. In addition, analysis methods considering processing of tasks during field uses are very important to investigate approaches and time management during complaint investigations. This method considers all target and performance dates by calculating the days of complaint processing of different vehicles in comparison to the complaint income date. Another method allows the investigation of KPIs related to delays in terms of registration by calculating registration times of different models and types. This enables a statement about the whole complaint process for quality assurance and influences the investigation of lifetime distribution (see Figure 4-43 and Figure 4-44).
Another important method shows how many failure codes appear in the field in relation to their system production date. For this purpose, failure codes of a defined time range are related to production phases in form of months, as shown in Figure 4-38. Quality assurance can use this method to detect critical production phases and improve the production line or development processes. The analysis method for quality assurance is the comparison of fault distributions of development and field data (see chapter 5.5).

5.4.2.3 Analysis process for complaint investigation

Besides the above-mentioned KPIs and analysis methods, this applied tool delivers various benefits and possibilities for the division of complaint investigation, e.g.:

- Efficient storage for data management of various sources:
  The concept of the generic database and implementation of various data sources by parsers allows complaint investigation to store all information. The connection of data enables a combined database for further investigations.
- Improvement of maintenance of knowledge:
  Due to the database, the investigation of various faults can be stored to improve the knowledge of component behaviour. If new components have to be carried out by a complaint investigation, the database provides support for analysing faults and their causes efficiently. Due to this tool, the complaint investigation division is also able to deliver precise expert knowledge to engineer new product generations.
  The connection of various data sources, such as field, measurement, complaint data, and fault databases, enables an analysis of the whole product behaviour and investigation history. Thus, complaint investigation is able to check and control the completed process after SOP.
- Retrospective and preventive analysis methods:
  Complaint investigation uses similar retrospective and preventive analysis methods as engineering and quality assurance to evaluate a system's behaviour as mentioned above.

5.5 Application of stochastic reliability investigations in development and field use

The application of the new analysis method for reliability investigation follows the presented approach, see chapter 4.5.

5.5.1 Data collection and preparation

The application of the stochastic reliability approach starts with the collection of data. Thus, software faults in development, milestone lists including system releases, failure codes in field use, and the diagnostic specification are necessary. The preparation of the data comprises allocating faults of development to system releases, elimination of redundancies, correction of failures during data input, consideration of delays, and reasonable negligence of not representative or unimportant data. Ideally, a relation and normalisation of faults in development and failure codes in field use is possible using similar reference values. In the application of automotive software, protected test environments (like MiL, SiL, and HiL) detect faults rather than real vehicle tests. Thus, the comparison and relation to failure codes of field use is difficult. Division through number of aggregates or vehicles can normalise failure codes from relevant periods of field use.

5.5.2 Descriptive statistics

The result of descriptive analysis enables overviews of collected and prepared data. Figure 5-15 shows faults in development related to system releases (main releases: bolt lines, intermediate releases: thin lines). Figure 5-16 shows the distribution of failure codes divided through number of systems in the field use.

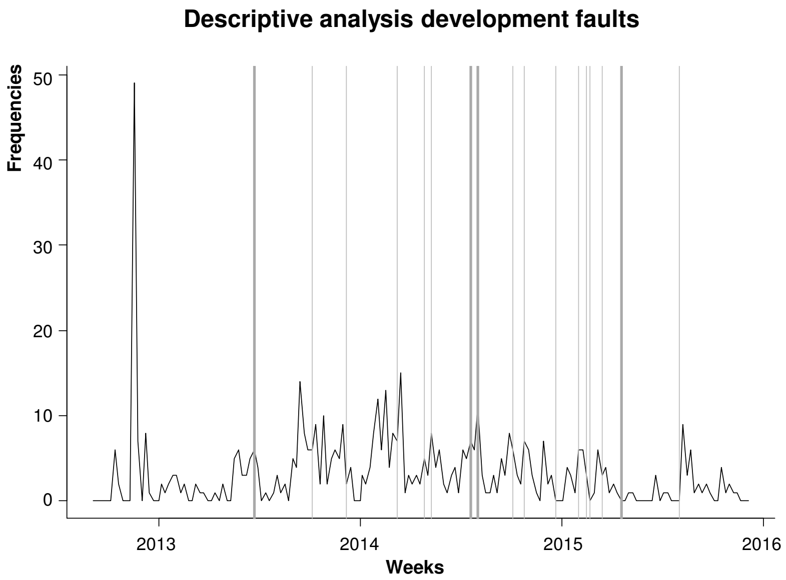

Figure 5-15: Descriptive analysis of detected faults in automotive software development (exemplary application)

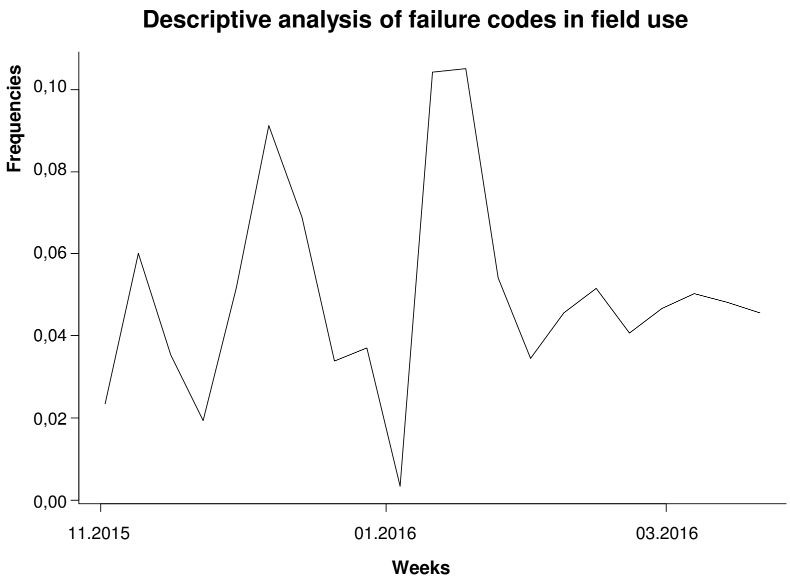

Figure 5-16: Descriptive analysis of software failure codes in field use (exemplary application)

5.5.3 Basic modelling

The next stage of the stochastic analysis approach represents basic modelling. Due to the mentioned characteristics of automotive application, a GLM with Poisson distribution is used to describe faults in development and failure codes in field use. Considering different release points with changing maturity and functionality of the automotive software over a development process, modelling fault rates by GLM has to be done for every main system release, as shown in Figure 5-17. In field use, one GLM model describes failure rate distribution, as shown in Figure 5-18. In case of software flashing, an enhancement of this method is repeating modelling after flash points during field use to consider different maturity of the system. Besides GLM both diagrams show also 95% confidence intervals of several models, c.f. [45], [49], [52], [122].

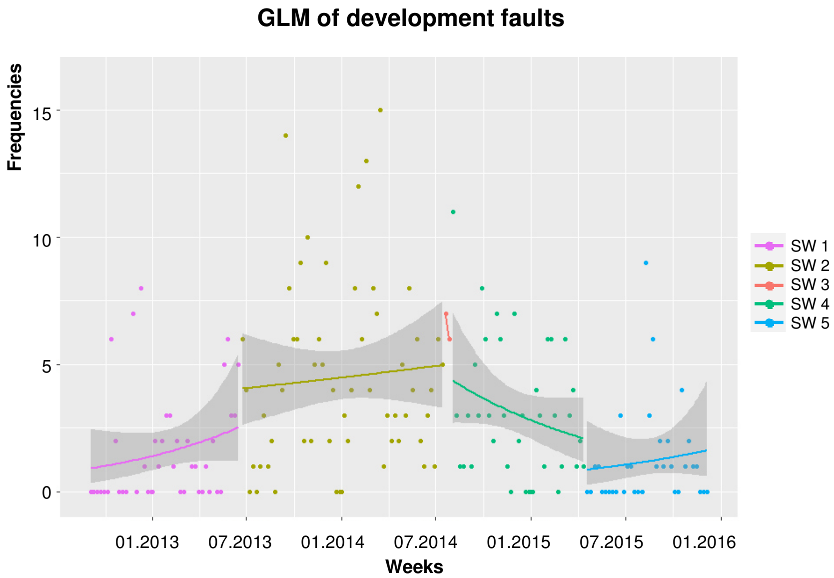

Figure 5-17: GLM of faults in automotive software development of the exemplary application

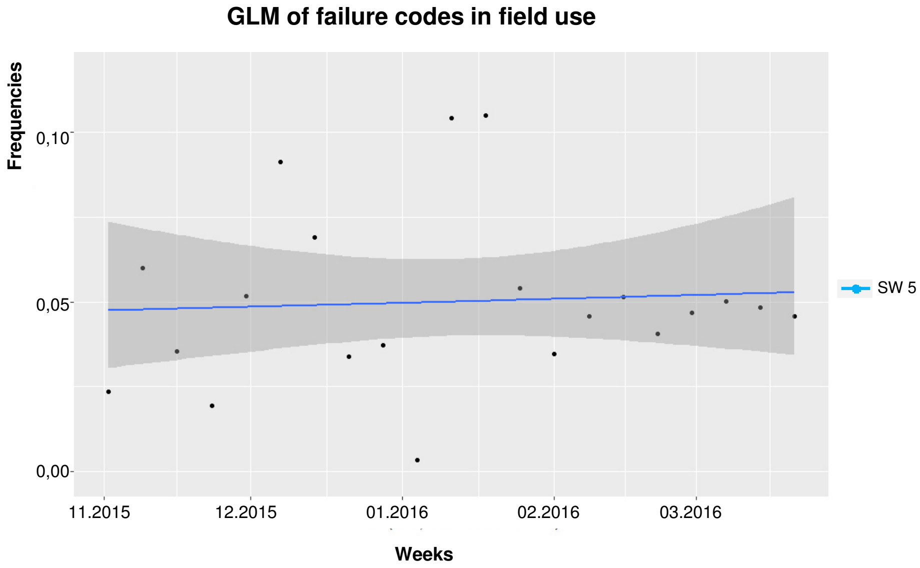

Figure 5-18: GLM of failure codes in field use of the exemplary application

Figure 5-19 and Figure 5-20 show the analysis of fitting of GLM as well as the parameter estimation by MLE in development and field use. The investigation passes through four steps. First step is the analysis of residuals data in comparison to modelled values of GLM (upper left) to check the fitting of the model. The residuals should be a randomly distributed, showing low distance and no significant trend. The next step identifies the scale location as variance of measured values over modelled parameters to analyse the deviation of a mean value, the so-called homogeneity of variances (lower left). Thus, fitting is constituted by no significant trend and a deviation beyond the standardised distance. The influence of separate values on regression can be demonstrated by Cook's distance [23] of observation points (upper right). The last step of fitting analysis of GLM is checking the Leverage effect of these separate observation points to identify possible critical elements. Due to the observation by Figure 5-19 and Figure 5-20, the GLM and parameter estimation fits for development and field data. Considering the Leverage analysis, the few data points with higher squared distance have no critical impact to distort the result. [122].

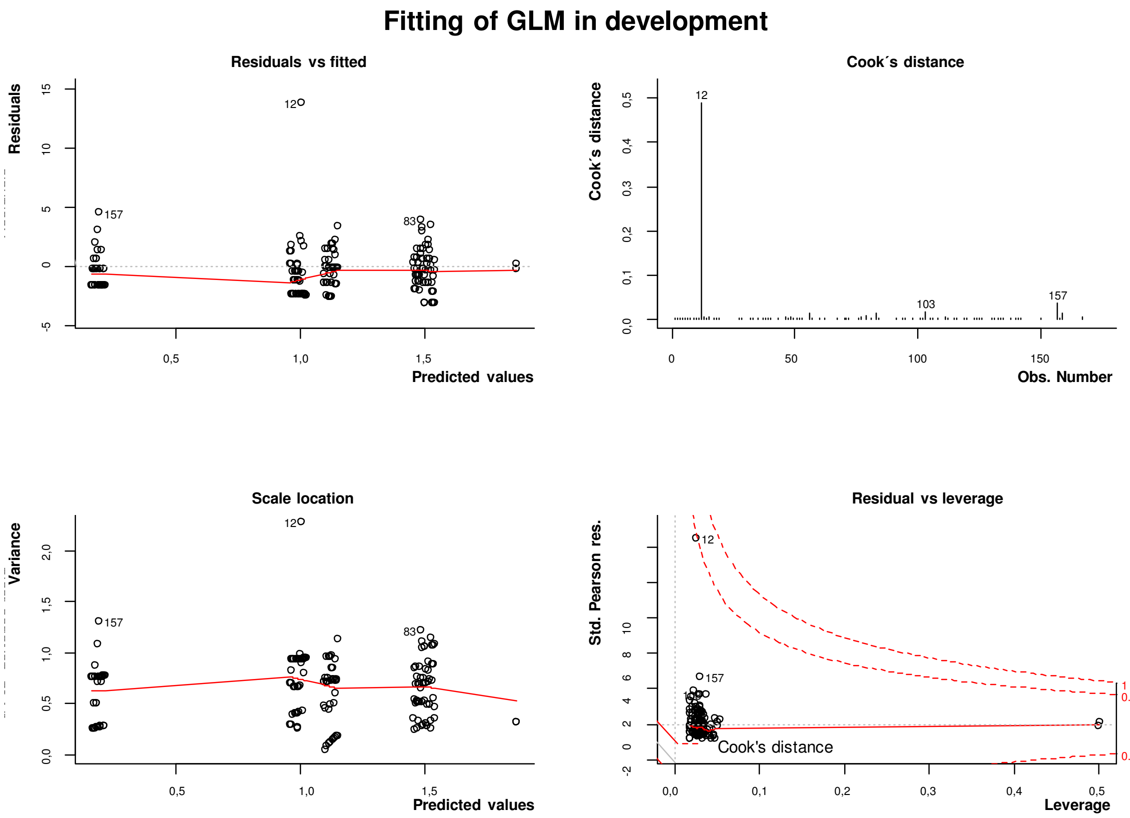

Figure 5-19: Analysis of GLM of faults in automotive software development

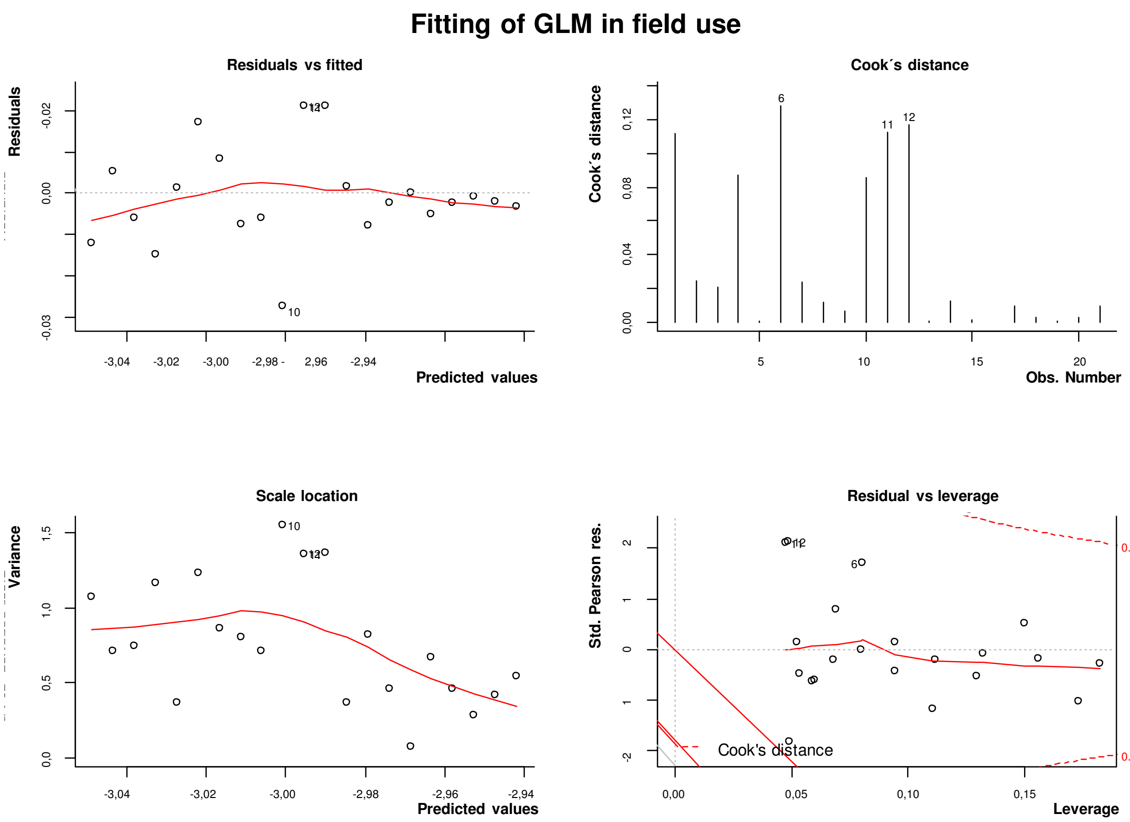

Figure 5-20: Analysis of GLM of failure codes in field use

5.5.4 Extra modelling by prediction and relation

The final stage of the stochastic approach deals with prediction of the residual error rate of development data in comparison to field rates. For this purpose, the GLM in development can be used to predict the residual error rate at a certain release point. To enable comparable models the increasing maturity and functionality of automotive software over development time has to be considered. Hence, the GLM for prediction is based on detected faults of the last testing loop with full maturity and functionality of the automotive software system, as shown in Figure 5-21. This enables a comparison of predicted values as residual error rates with the real faults during field use, as shown in Figure 5-22.

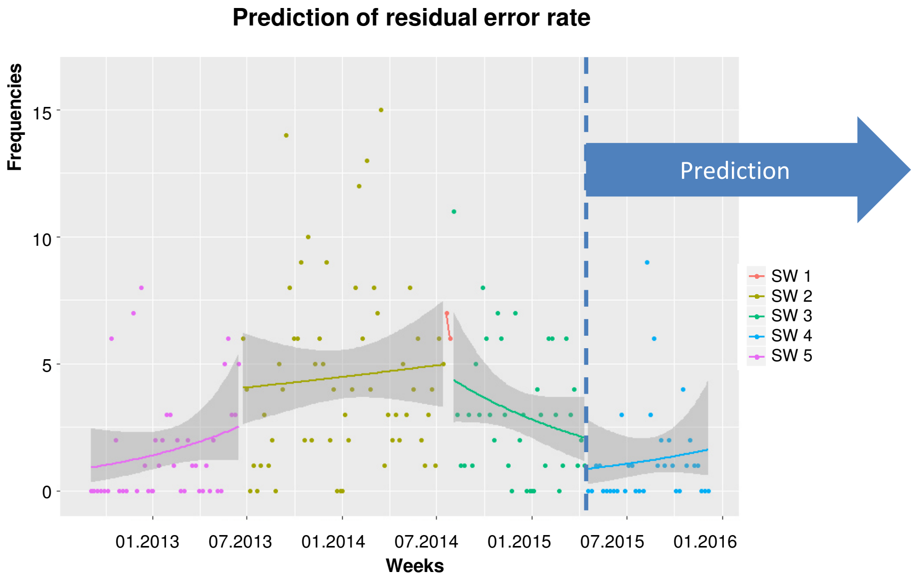

Figure 5-21: Considering maturity and functionality of automotive software for prediction

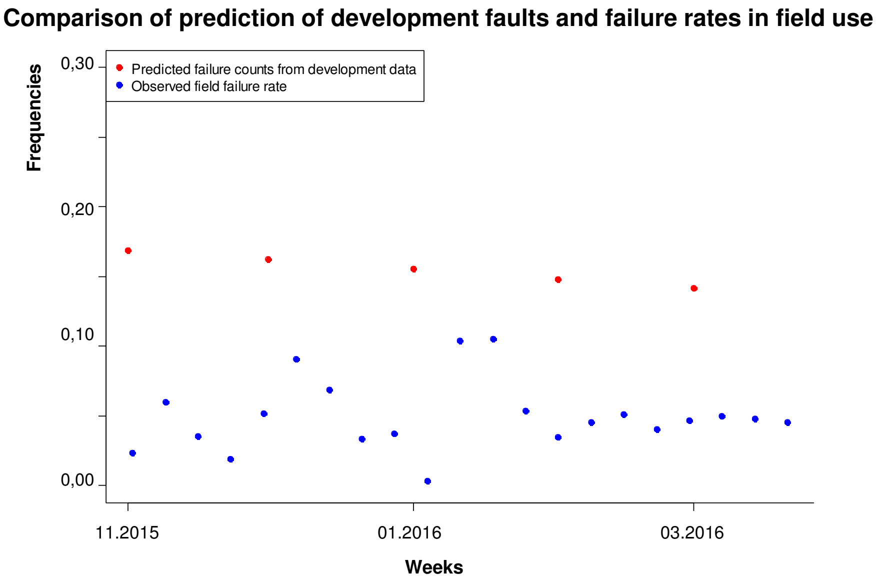

Figure 5-22: Predicted values of GLM in development in comparison to field use

The deviations of prediction and values of field use depend on related reference values or fitting of the model. For the application of automotive software, the comparability will always be a challenge due to differences between MiL, SiL, or HiL testing and field use. Nevertheless, one possibility to describe the difference can be enabled by using LM or GLM to calculate a factor or comparability function of predicted and field values:

$$\log(\text{predicted values}) - \log(\text{field use values}) = \text{estimated factor} \tag{5-1}$$

In this application the estimated factor for LM and GLM is about -1.5. Fitting LM shows, beside the analysis of the residuals, the standard Q-Q plot enables a suggestion if the residual errors are normally distributed. Fitting the GLM is similar to the above-mentioned method.

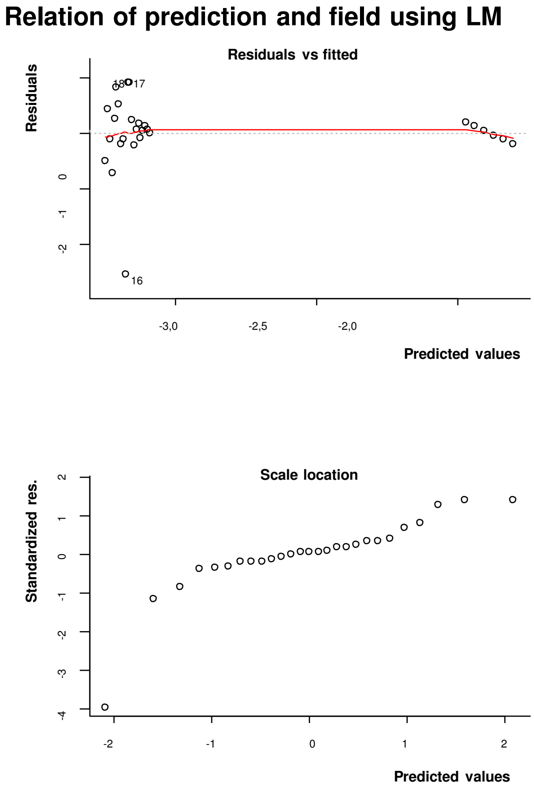

Figure 5-23: LM as relation of development and field

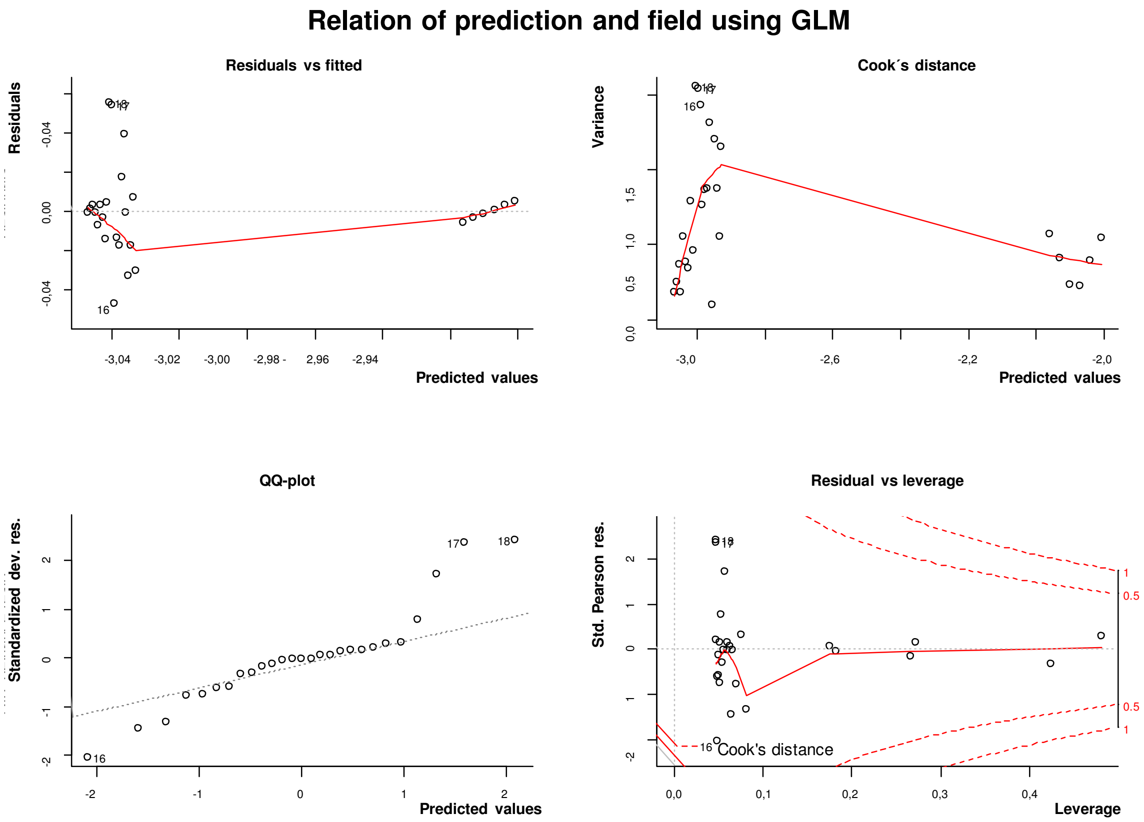

Figure 5-24: LM as relation of development and field

The investigation of fitting of LM or GLM to compare the predicted values of development data and failure rates in the field shows no significant trend or abnormality. Hence, the presented method can be introduced as self-learning early warning system to support release-related decisions.
The more projects can be analysed during development and field use, the more exact is the prediction method. Thus, with increasing data the generic approach enables a self-learning system that provides better prediction of the residual error rate. This method can be used as early warning system during development, to check if bug fixing loops work and if further testing periods are necessary. Furthermore, the prediction of the residual error rate supports the decision, if testing and bug fixing can be finished to save development time and costs. Additionally, it enables a statement of possible warranty issues in the field and the introduction of preventive measures to avoid these impacts.

6 Results and potentials of to support process optimisation

The potentials of the presented analysis methods enable an integrated investigation and control of the whole PLC. The generic approaches of the developed methods support an application in every industrial area.

6.1 Potentials of comprehensive analysis method

The main potential of the comprehensive analysis method is the enhancement of common analysis methods like FMEA and FTA for several industrial applications. This method enables complete new analysis possibilities by considering complex functions and failure behaviour of systems. Furthermore, all advantages from common analysis methods are combined and enhanced by this new method. Hence, the danger of incomplete investigations of systems combined with unconsidered failures and prevention measures can be efficiently decreased. Using this method enables an efficient job of the risk and FMEA specialists of a company. Introducing the innovative method in PLC shows various potentials. The delivered KPIs are e.g. number of possible failures, risk priority number, and quantitative risk parameters due to Boolean algebra. The first use of this method takes place during requirement phase and at the beginning of development process. Additionally, this method can be used during the development process to analyse changes of the system with their effects on possible failure behaviour. An iterative development can be enabled to include all further changes in the process. Another possibility is to support fault cause investigations during complaint data analysis. This can be done by the enhanced visualisation enabling the analysis of signal, data, or failure flows. Thus, the comprehensive analysis method can take place in every step during PLC with focus on requirement phase.

6.2 Potentials of analysis methods by using Key Performance/Process Indicators (KPIs) on basis of development and test data

The analysis methods using KPIs on basis of development and test data answer different questions of various stakeholders (e.g. general management GM, project management PM, process management PrM, or quality assurance QA). The generic definition and approach of these developed methods enables an introduction

in several industrial applications. The main points for different industrial applications are the definition of KPIs to give satisfying statements and present a structured, connected database, [52].
The potentials of the KPIs delivered by the developed analysis methods are varied and can be used in different ways, such as in development, during use–phases, further projects, continuous process improvement, and self-learning systems. The method to control status and progress in development enables KPIs to provide continuous investigation during the whole process. Project management and quality assurance can use these methods to control issues, evaluate different phases of the development process, and compare different projects. Furthermore, monitoring of these methods delivers relevant KPIs to avoid ignoring of not solved or critical elements which is important for general management. Another potential is an efficient KPI calculation of possible time lags and their effects to ensure compliance of time schedules. The next analysis category "durations" also can take place in all phases of the development process. The potentials of these methods are control of durations to get KPIs considering averaged key-values of different projects, development phases, or tasks. This information enables the definition of realistic durations for further projects and tasks as well as the evaluation of critical phases during development processes. A combination of methods "status and progress" with "durations" comprises the potential to compare KPIs of project progress and maturity. The methods and KPIs related to "target-performance comparison" support release-related decisions of general management and control by quality assurance. Thus, a risk matrix including boundary values can be introduced to check if the conditions for a certain release or milestone are fulfilled. Using this method for different projects enables enhanced and exact boundary values for risk matrix by a self-learning system. The basis for all analysis methods is an effective data management, because compliance of data documentation is the precondition for efficient introduction of analysis methods. This can be controlled by quality assurance using the analysis method "tracking". Hence, this method enables the control of data connection and grade of database entries. The potentials of the analysis method "trend of planning elements" are the delivered KPIs which provide an early warning system for several interest groups during development phases. This method enables the analysis of deviations or gaps between target and performance related parameters. Hence, critical phases and tasks during development can be defined by investigation of the trend of gaps as a function of project maturity. An introduced early warning system enables the potentials to react in a current project, to support the planning of resources, and to deliver knowhow for further projects. Another trend analysis also takes place in all phases of development process, especially during testing. This method

"trend of ratios" delivers different parameters as relative values. The significance of these parameters increases in later phases. The goal is to investigate the saturation of parameters in percentage. The aim is to get high KPI values as soon as possible. An early warning system can be introduced by the use of boundaries related to time steps. Hence, a warning appears when pre-defined boundaries are not reached at a certain time step to react before it comes to unrecoverable impacts. For this purpose, a reference project demonstrating the ideal progress serves for comparison and evaluation of certain projects. In this context, "fault estimation" is a very important method to detected and analyse abnormalities during development processes, e.g. giving a statement about the necessity and efficiency of bug fixing loops. Another potential is detecting the KPI of critical recurring faults over different projects to optimise development processes. The goal increases the reliability of development over various projects resulting in better products by detecting and fixing similar faults or systemic deviations. In addition, modelling of faults in development enables the KPI residual error rate and a trend analysis of fault distribution during all project phases. This method comprises the possibility of an early warning system delivering a statement for further bug fixing loops or risk analysis using prediction at certain release points. These potentials can be improved by the combination and comparability of modelling, testing, and real-field uses. Hence, the feedback of information of test and field uses is a precondition for exact prognosis models of early warning systems. The introduction of a self-learning system using prognosis models and KPI based evaluation will lead to a significantly higher reliability of products, [52]. The generic concept of analysis on the basis of development and test data as well as the delivered KPIs has various potentials for several industrial applications.

6.3 Potentials of analysis methods using field and complaint investigation data

The method to combine several data sources in field use into a structured database delivers the precondition for an effective data handling as well as the introduction of effective analysis possibilities. The potentials of analysis methods considering field and complaint data are varied. Starting with retrospective analysis methods, the potentials are the receipt of various information about product and system behaviour in the field use. The knowledge of past events delivers important inputs for further development and projects. Due to the generic database concept, a comparison of different projects, the customer, and their effects on failure behaviour can be investigated. Furthermore, the target-oriented feedback of KPIs considering product behaviour, circumstances, environmental impacts, and failure causes in phases of development enable an improvement of quality as well as specific performance analysis of development divisions. Another potential of retrospective investigations is the analysis of KPIs considering changes (e.g. flashing new software) in comparison to previous product behaviour. The potentials of predictive analysis methods are very important for engineering, quality assurance, and complaint investigation to estimate if problems are coming up in the near future. The predictive analysis by use of boundaries or checks enables an early warning system. The analysis of the wearing process by delivering predictive KPIs enables the prognosis of the further wearing or possible times of breakdowns. Besides, data related to tracking can be used to deliver continuous inputs of the behaviour and relevant KPIs for different development phases. Due to the increasing connection and interaction of systems (e.g. autonomous driving) measurement data by tracking delivers more and more information on field behaviour.

6.4 Potentials of the reliability investigation in development and field use

The reliability investigation of development and field data enables several possibilities to analyse or predict phases of the PLC. Besides these potentials, the final stage of the developed stochastic approach lies in the comparison of the KPIs of the predicted residual error rate of development data and field rates. The potentials of combining GLMs of development and field data are a prediction of the residual error rate during development to enable an early warning system. The aim is to support release-related decisions during development. The prediction of the residual error rate enables a significant statement if a system can be released or if further bug fixing loops are necessary. Hence, the developed concept is a self-learning, early warning system to support release-related decisions. This method can be used to save costs and time in development phases by ensuring the optimised point for a product release.

6.5 Potentials to combine the developed methods

The possible combinations of the developed methods are versatile (example shown in Figure 6-1). For starting the PLC with the requirement phase, the comprehensive analysis method is needed to enable an overview of all possible product faults. This method influences the development phase and contains the potential to avoid failure behaviour of products by introducing preventive measures as soon as possible. Thus, this method affects development processes by the two KPIs number of possible failures and risk priority number. The comprehensive analysis method

can also be applied in field phases to support fault cause investigations. Furthermore, KPIs considering quantitative risk parameters due to Boolean algebra enable significant inputs for reliability investigations.
One example to combine the methods in development and field phases is the prediction of the KPI residual error rate by using stochastic analysis. Modelling detected faults during development delivers the residual error rate, investigating field data delivers the real fault rate. Stochastic modelling and combination of these fault rates enable a self-learning prediction process of the residual error rate during development phases. Here, the main potential is saving time and costs in development.
The developed field analysis methods can directly affect requirement and development phases. One example for this combination potential is the feedback of fault behaviour and causes. This enables improvement of processes during requirement and development phases to avoid these fault causes in new product generations. Furthermore, the investigation of field use supports knowledge building about customer behaviour to enable customer-oriented development and production processes. Additionally, the introduction of consistent KPIs, including reliability parameters in all phases of the PLC, enables the potential of self-learning processes to get a better understanding and investigation of product and component reliability.

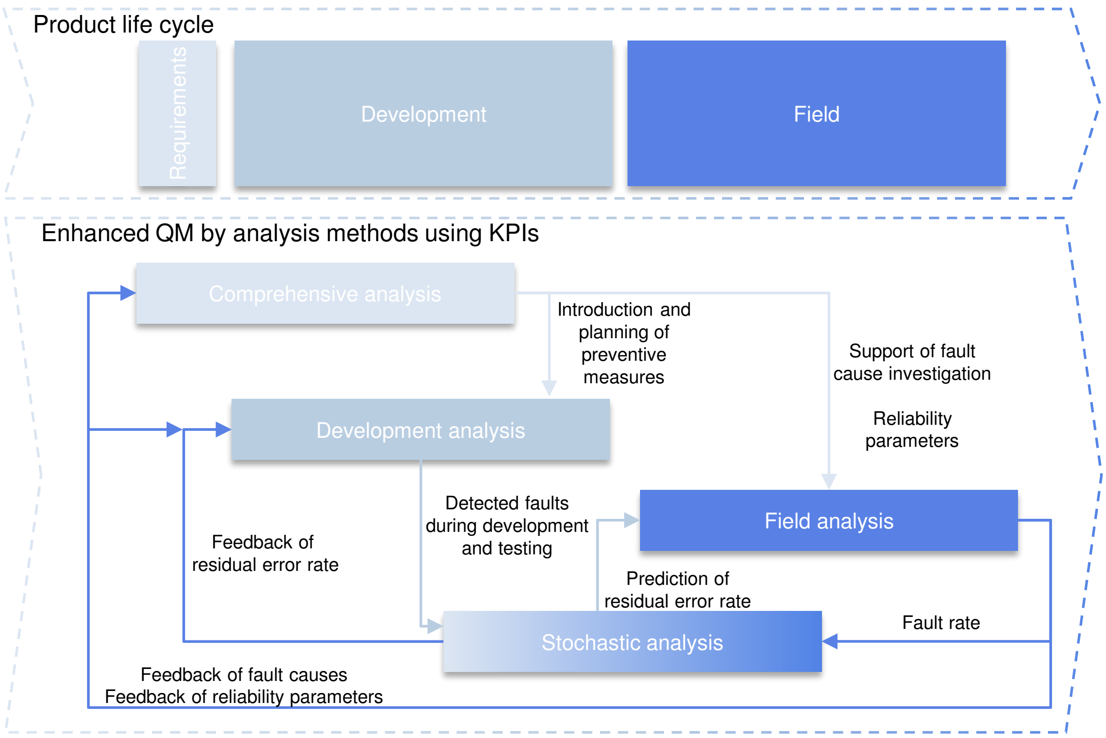

Figure 6-1: Exemplary combination potential of the developed methods during PLC

7 Conclusion and final statement

The trend of increasing complexity of products and customers' demands for quality represents enormous challenges for development processes. Current industrial applications need mechatronic systems to meet the functional requirements, e.g. in automotive industry the alternative propulsion systems, connectivity, and enhanced safety measures. Hence, the development process of mechatronic systems and investigation of the product life cycle (PLC) are a huge challenge in practice regarding the different included components such as electronics/electrics (E/E), software, and mechanics, as well as their interaction. However, malfunctions and teardowns of these systems occur during operation which could lead to unacceptable consequences and thus represents a huge risk due to functional safety in industrial engineering. Thereby, safety-relevant components are very important because their breakdown could lead to damage of humans and machines. In addition, the avoidance of warranty claims due to malfunctions in field use is very important for companies. The increasing quality and safety standards are a big challenge for OEMs and their supplier, because default risks have to be reduced or reliability of components has to be increased. The multitude of technical properties and complex interactions of mechatronic systems of different industrial applications (e.g. modern vehicle concepts) require innovative analysis and evaluation methods both in development and in field uses. A reliable development of such complex systems can only be enabled by an optimisation of an integrated development process of mechatronic systems and the investigation of field use behaviour.

The aim of the PhD thesis is to introduce KPIs by new analysis methods based on development, test, and field data to enable optimised, reliable, and predictive development processes and control of the whole PLC. Hence, analysis methods for delivering significant KPIs are developed and introduced in different phases of the PLC to reduce the occurrence of malfunctions in field use as well as warranty claims. Through this, requirements of quality standards are ensured.

The generic approach of the methodology starts with the introduction of a comprehensive analysis method in the requirement phase to provide new possibilities for depicting energy and information flows as well as fault connections and error propagation paths. Thus, faults and their consequences can be determined more precisely and safety measures can be better introduced to increase safety and reliability of systems. The next step of the approach provides an optimisation of development processes by introduction of KPIs delivered by analysis methods on basis of development and test data. The developed methods enable analysis and evaluation of occurred faults and their reasons. These fault distributions can be transferred to reliability models by using stochastic methods resulting in a prognosis model to estimate a residual error rate. Furthermore, the development of new analysis methods can be used to define KPIs for different interest groups to evaluate and analyse development processes by objective, quantitative metrics. A combination of the evaluation of development and test data, prognosis modelling, and risk analysis supports release decisions of mechatronic systems. The next step deals with the evaluation of field data as an important part to analyse the integrated development process and PLC. This analysis and a well-aimed allocation of malfunctions and their respective reasons enable a significant potential to generate new knowledge of product behaviour. Thereby, many different influence factors such as environmental conditions and vehicle types have to be considered. The aim is to generate knowledge and findings from field use and feed this information back to cause-related development phases. A combination of the mentioned analysis and evaluation approaches based on development and field data enables an enhanced and integrated method to support the development process of automotive mechatronic systems. By use of reliability models, a residual error rate can be predicted to decrease faults during PLC and optimise the development of new products or product generations.

The PhD thesis enables a generic approach which is used in series development projects yet. Due to this direct relation of developed analysis methods to series development, a direct application in automotive industry as well as in other industrial areas is possible to improve product quality.

Appendix

A-1 Content of generic database for application

The generic information of the database includes the following information categorised by data source which represents the necessary data for further analysis:

- Fault database:
  - Series number
  - Aggregate type
  - Project
  - Part number
  - Part number customer
  - Production date
  - Version of software
  - CC-number
  - 8D-number
  - Customer complaint number
  - Customer-8D-number
  - Customer complaint date
  - Incoming goods date
  - Customer report target date
  - RK type
  - Failure code
  - Failure text
  - Other failure message
  - Failure type
  - Findings
- Field data:
  - FS-type
  - SGBD
  - Storage position
  - VIN
  - Readout time
  - Failure code
  - Failure text
  - FA/UB
  - FA/UB number
  - UB km

  - UB value
  - UB unit
  - FA type
  - UB set
  - ECU series number
  - Vehicle type
  - Motor type
  - km
  - Failure frequency
  - Logistic counter
  - FA/UB text
- Measurements data:
  - VIN
  - Vehicle type
  - Motor type
  - Readout date
  - km
  - Production date
  - SW level
  - Job name
  - Job result
  - Parameter number
  - Parameter name
  - Parameter value
  - Hex-Dump
  - Unit
  - SGBD
  - Expire
  - Land

Measurement data represents a two dimensional information, because the category job result includes further measurement values. In the presented example, each job result represents one readout process and includes 181 values, such as:

  - Job result:
    - Project
    - Complaint number (CC, Customer Complaint)
    - Reference number (representing bus system, e.g. FR=flex ray, CAN)
    - External complaint number (from customer)
    - External aggregate number (from customer)

        - Aggregate number
        - Part number
        - Aggregate type
        - ITA/NIA type („ITA“ = Integrated Transfercase Actuator, „NIA“ = New Integrated Actuator)
        - ITA/NIA data code (production date)
        - ITA/NIA series number
        - ITA/NIA part number (from customer)
        - ITA/NIA part number (from industrial partner)
        - VIN
        - Vehicle type
        - Motor type
        - Mileage
        - Customer complaint date
        - Incoming goods date
        - Production date
        - Complaint reason
        - Attached failure memory
        - Findings
- Complaint investigation data: (which is devided into a standard part and customer specific diagonstic jobs)
    - Standard part:
        - Project
        - Complaint number (CC, Customer Complaint)
        - Reference number (representing bus system, e.g. FR=flex ray, CAN)
        - External complaint number (from customer)
        - External aggregate number (from customer)
        - Aggregate number
        - Aggregate part number
        - Aggregate type
        - ITA/NIA type („ITA“ = Integrated Transfercase Actuator, „NIA“ = New Integrated Actuator)
        - ITA/NIA data code (production date)
        - ITA/NIA series number
        - ITA/NIA part number (from customer)
        - ITA/NIA part number (from industrial partner)

        - VIN
        - Vehicle type
        - Motor type
        - Mileage
        - Customer complaint date
        - Incoming goods date
        - Production date
        - Complaint reason
        - Attached failure memory
        - Findings
    - Diagnostic part:
        - Including various diagnostic jobs and customer-related information

The equivalence list of this generic system includes all information with generic labels, such as:

- SW levels
- Bootloader
- ECU system
- Vehicle
- Net calibration number
- ITA/ECU series number
- Measurement unit (for different projects)
- Lifetime work oil (performed work of oil representing wear)
- Lifetime work 1 to 3 (performed work of discs representing wear)
- Lifetime work 4 (reserved field for further projects)
- Lifetime work chain (performed work of drivetrain chain)
- Total distance (overall distance)
- Run in comp 1
- STAT distance (resetable distance, e.g. service)
- Km since last oil change
- Work oil 4 diag (wear of oil in kWh)
- Over roll counter
- Gain class (slope of clutch field)
- Offset (distance of clutch field)
- Delta phi cal first (first calibration angle)
- Stat delta phi calibration (difference of first and current calibration angle)
- Count calibration (number of performed post calibration)

- Count force (performed strict calibration)
- Number of planned calibrations
- Count temperature range 01 to 06 (different temperature areas)

For further detailed information of data storage systems see [136].

List of figures

List of tables

References

[1] Ahrens, V.; Hieronymus, M.: Arbeitspapier der Nordakademie, Das Kreismodell als Grundlage der Entwicklungsmethodik für cyber-physikalische Systeme, ISSN 1860-0360, Hochschule der Wirtschaft Nordakademie, [online] available: https://www.nordakademie.de/fileadmin/dateien/downloads/Dokumente/Forschung/Arbeitspapiere/AP_2014_07.pdf, accessed on: May 09 2016, Elmshorn, Germany

[2] Atlassian JIRA Software: Jira, [online] available: https://de.atlassian.com/software/jira, accessed on: Oct. 22 2015, Sydney, Australia

[3] AUTOSAR: [online] available: http://www.autosar.org/, accessed on: Sep. 15 2015, Munich, Germany

[4] Axivion GmbH: [online] available: https://www.axivion.com/de/produkte/axivion-bauhaus-suite/index.html, accessed on: Jun. 10 2015, Stuttgart, Germany

[5] Bauke, H.: Parameter estimation for power-law distributions by maximum likelihood methods, vol. 58, no. 2, pp. 167–173, 2007

[6] Baumann, G.: Neue Testmethoden für vernetzte Kraftfahrzeugelektronik im Labor und Fahrversuch, Forschungsinstitut für Kraftfahrwesen und Fahrzeugmotoren Stuttgart (FKFS), Stuttgart, Germany, 2007

[7] Beck, K.: Embracing Change with Extreme Programming, pp. 70–77, IEEE Computer, U.S., 1999

[8] Belli, F.; Grochtmann, M.; Jack, O.: Erprobte Modelle zur Quantifizierung der Software-Zuverlässigkeit, Informatik Spektrum 21, pp. 131–140, Springer, Germany

[9] Bender, K.: Embedded Systems, Qualitätsorientierte Entwicklung, Springer, available: http://site.ebrary.com/lib/alltitles/docDetail.action?docID=10182914, Berlin, Germany, 2005

[10] Bertsche, B.: Reliability in Automotive and Mechanical Engineering, Determination of Component and System Reliability, VDI, Springer, available: http://site.ebrary.com/lib/alltitles/docDetail.action?docID=10281321, Berlin, Heidelberg, Germany, 2008

[11] Bertsche, B.; Göhner, P.; Jensen, U.; Schinköthe, W.; Wunderlich, H.-J.: Zuverlässigkeit mechatronischer Systeme, Grundlagen und Bewertung in frühen Entwicklungsphasen, VDI, Springer, available: http://dx.doi.org/10.1007/978-3-540-85091-5, Berlin, Heidelberg, Germany, 2009

[12] Bishop, R. H.: Mechatronic systems, sensors and actuators, Fundamentals and modeling, The electrical engineering handbook series, Taylor & Francis distributor, available: http://site.ebrary.com/lib/alltitles/docDetail.action?docID=10196284, CRC Press, Baton Rouge, U.S., 2008

[13] Boehm, B., W.: Center for System and Software Engineering, Guidelines for verifying and validating software, [online] available: http://csse.usc.edu/TECHRPTS/1979/usccse79-501/usccse79-501.pdf, accessed on: Nov. 05 2016, Los Angeles, U.S., 1979

[14] Boehm, B., W.: A Spiral Model of Software Development and Enhancement, pp. 61–72, IEEE Computer 21, 5, TRW Defense Systems Group, U.S., 1988

[15] Borgeest, K.: Elektronik in der Fahrzeugtechnik, Hardware, Software, Systeme und Projektmanagement ; mit 28 Tabellen, ATZ/MTZ, 3rd ed., Springer Vieweg, Wiesbaden, Germany, 2014

[16] Brodtrager, C.: Entwicklungsprozesse und Eingliederung von Analysetools, bachelor thesis, Institute of Automotive Engineering, Graz University of Technology, Graz, Austria, 2016

[17] Brüggemann, H.; Bremer, Peik: Grundlagen Qualitätsmanagement, Von den Werkzeugen über Methoden zum TQM, Lehrbuch, 2nd ed., Springer Vieweg, available: http://dx.doi.org/10.1007/978-3-658-09221-4, Wiesbaden, Germany, 2015

[18] Bruhn, M.: Qualitätsmanagement für Dienstleistungen, Handbuch für ein erfolgreiches Qualitätsmanagement. Grundlagen - Konzepte - Methoden, 9th ed., Springer Gabler, available: http://dx.doi.org/10.1007/978-3-642-33992-9, Berlin, Germany, 2013

[19] CA Technologies: [online] available: http://www.ca.com/us/products/ca-agile-central.html?intcmp=headernav, accessed on: Nov. 12 2015, New York, U.S.

[20] Center of Automotive Management: AutomotivePERFORMANCE 2013, 2014, 2015, [online] available: http://auto-institut.de/pm_studien.htm, accessed on: Nov. 05 2015, Bergisch Gladbach, Germany

[21] Chemuturi, M.: Requirements engineering and management for software development projects, Springer, New York, U.S., 2013

[22] Chen, Y.; Singpurwalla, N., D.: Unification of Software Reliability Models by Self-Exciting Point Processes, vol. 29, London, U.K., 1997

[23] Cook, R., D.: Detection of Influential Observation in Linear Regression, vol. 19, no. 1, p. 15, 1977

[24] Cooper, R., G.: Third-Generation New Product Processes, pp. 3–14, Journal of Product Innovation Management, volume 11, issue 1, Chicago, U.S., 1994

[25] Dallinger, P.: Entwicklung eines Datenmodells einer innovativen Analysemethode für mechatronische Systeme im Fahrzeug, master thesis, Institute of Automotive Engineering, Graz University of Technology, Graz, Austria, 2015

[26] Dallinger, P.; Ernst, M.; Fabian, J.; Schnellbach, Adam: Development of a Data Model for an Innovative Analysis Method for Mechatronic Systems in the Field of Automotive Engineering, Proceedings of 2015 International Conference on Advances in Software, Control and Mechanical Engineering, Antalya, Turkey, 2015

[27] Danzer, H., H.: Qualitätsmangement, Vorlesung, Graz University of Technology, Graz, Austria, 2015

[28] Delonga, M.: Zuverlässigkeitsmanagementsystem auf Basis von Felddaten,Universität Stuttgart, Germany, 2007

[29] Deming, W., E.: Out of the Crisis, p. 88, MIT Press, Cambridge, U.S., 1986

[30] Dernoscheg, S.: Konzeptionelle Risikoanalyse für Produktfreigaben in der Automobilindustrie, bachelor thesis, Institute of Automotive Engineering, Graz University of Technology, Graz, Austria, 2014

[31] Deutsches Institut für Normung: DIN 25424 Fehlerbaumanalyse, Methode und Bildzeichen, Beuth, Germany, 1981

[32] Deutsches Institut für Normung: DIN 40041:1990-12 Zuverlässigkeit; Begriffe, Berlin, Germany, 1990

[33] Deutsches Institut für Normung: DIN 55350-11:2008-05 Begriffe zum Qualitätsmanagement, Berlin, Germany, 2008

[34] Dold, A.: Implementation of Requirements From ISO 26262 in the Development of E/E Components and Systems – Challenges & Approach, Automotive Electronicsand Electrical Systems Forum 2008, Stuttgart, Germany, 2008

[35] Dorfman, M.; Thayer, R., H.: Software requirements engineering, IEEE Computer Society Press, Los Alamitos, U.S., 1997

[36] Duane, J., T.: Learning Curve Approach to Reliability Monitoring, IEEE Transaction on Aerospace, vol.2, no.2, U.S., 1964

[37] Ebner, C.: Rückkopplung von Erkenntnissen aus dem Einsatzverhalten von Fahrzeugen auf die Entwicklung von Automobilkomponenten, bachelor thesis, Institute of Automotive Engineering, Graz University of Technology, Graz, Austria, 2014

[38] Eclipse Foundation Europe GmbH: OpenMDM, [online] Available: https://www.openmdm.org/about-openmdm, Germany

[39] Edler, A.: Nutzung von Felddaten in der qualitätsgetriebenen Produktentwicklung und im Service, doctoral thesis, Berichte aus dem Produktionstechnischen Zentrum Berlin, IPK, Berlin, Germany, 2001

[40] Eigner, M.; Stelzer, Ralph: Product Lifecycle Management, Ein Leitfaden für Product Development und Life Cycle Management, VDI, 2nd ed., Springer, Berlin, Heidelberg, Germany, 2013

[41] Erlachner, S.: Einsatz von Entwicklungs- und Testdaten zur Bewertung automotiver Software, master thesis, Institut für Softwaretechnologie, Graz University of Technology, Graz, Austria, 2015

[42] Ernst, M.: Marktpotentiale eines innovativen Analysetools für erzeugende Branchen am Markt Österreich und Deutschland, master thesis, Institut für Handel Absatz und Marketing, Johannes Kepler University, Linz, Austria, 2015

[43] Ernst, M.: 1. Zwischenbericht zur Analyse und Prognosemodellierung mechatronischer Fahrzeugkomponenten in der Entwicklung und im Feld, research report Nr. FTG-2014/01, Institute of Automotive Engineering, Graz University of Technology, Graz, Austria, 2014

[44] Ernst, M.: Analysis Methods in the Development Process of Mechatronic Drivetrain Systems with Special Focus on Automotive Software, presentation, The 19th World Multi-Conference on Systemics,Cybernetics and Informatics, WMSCI, Orlando, Florida, U.S., 2015

[45] Ernst, M. et al.: Key Process/Performance Indicators (KPIs) in der automotiven Softwareentwicklung, Research report FTG-2015/10, Institute of Automotive Engineering, Graz University of Technology, Graz, Austria, 2015

[46] Ernst, M.: The Potential of Key Process/Performance Indicators (KPIs) in Automotive Software Quality Management, presentation, SAE World Congress, Detroit, Michigan, U.S., 2016

[47] Ernst, M.; Dallinger, P.; Fabian, J.; Hirz, M.; Schnellbach, Adam: Innovative Analysemethode in der Entwicklung mechatronischer Systeme am Beispiel elektrifizierter Antriebsstränge, Elektrotechnik und Informationstechnik, DOI 10.1007/s00502-015-0292-7, vol. 132, no. 3, pp. 134–141, Springer, Vienna, Austria, 2015

[48] Ernst, M.; Dallinger, P.; Hirz, M.; Fabian, J.; Schnellbach, Adam: Potentiale einer innovativen Analysemethodik und deren Datenmodellierung, FMEA KONKRET, 02/2014, ISSN 2198-6851, FMEA plus Akademie, Ulm, Germany, 2014

[49] Ernst, M.; Erlachner, S.; Hirz, M.; Fabian, Jürgen: Optimisation of Automotive Software Quality Management by Use of Analysis Methods in the Development Process of Mechatronic Systems, International Conference on Advances in Software, Control and Mechanical Engineering, ICSCME, Antalya, Turkey, 2015

[50] Ernst, M.; Erlachner, S.; Hirz, M.; Fabian, J.; Wotawa, Franz: Analysis Methods in the Development Process of Mechatronic Drivetrain Systems with Special Focus on Automotive Software, The 19th World Multi-Conference on Systemics, Cybernetics and Informatics, WMSCI, Orlando, Florida, U.S., 2015

[51] Ernst, M.; Fabian, J.; Hirz, Mario: Innovative Methoden zur Analyse elektrifizierter Antriebsstränge zukünftiger Fahrzeugkonzepte, 13. Symposium Energieinnovation, p. 408, Graz University of Technology, Graz, Austria, 2014

[52] Ernst, M.; Hirz, M.; Fabian, Jürgen: The Potential of Key Process/Performance Indicators (KPIs) in Automotive Software Quality Management, 2016-01-0046, SAE World Congress, Detroit, Michigan, U.S., 2016

[53] European Comission: Verordnung (EU) Nr. 19/2011 der Kommission, [online] available: http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=OJ:L:2011:008:0001:0013:DE:PDF, accessed on: Oct. 25 2016, 2011

[54] European Parliament and of the Council: Directive 2001/95/EC on general product safety, available: http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=OJ:L:2002:011:0004:0017:DE:PDF, accessed on: Aug. 06 2016, 2001

[55] Federal Office of the Bundeswehr for Information Management and Information Technology: Federal Office of Bundeswehr Equipment, Information Technology and In-Service Support (BAAINBw), accessed on: Oct. 18 2016

[56] Federal Republic of Germany: Der Beauftragte der Bundesregierung für Informationstechnik, Federal Ministry of the Interior, Central Office for Information Technology Coordination in the Federal Administration (BMI-KBSt), [online] available: http://www.cio.bund.de/Web/DE/Architekturen-und-Standards/V-Modell-XT/vmodell_xt_node.html, accessed on: Apr. 13 2016, Germany

[57] Fehrer, F.: Entwicklungs- und Testdatenanalyse mechatronischer Komponenten im Automobilbau, master thesis, Institute of Automotive Engineering, Graz University of Technology, Graz, Austria, 2016

[58] Fischer, R.; Jürgens, G.; Küçükay, F.; Najork, R.; Pollak, Burkhard: Das Getriebebuch, Der Fahrzeugantrieb, Springer, available: http://site.ebrary.com/lib/alltitles/docDetail.action?docID=10580386, Vienna, Austria, 2012

[59] Fröhlich, P.: Überwachung verfahrenstechnischer Prozesse unter Verwendung eines qualitativen Modellierungsverfahrens. doctoral thesis, Institut für Automatisierungs- und Softwaretechnik, University of Stuttgart, Germany, 1997

[60] Gadatsch, A.: Grundkurs Geschäftsprozess-Management, Methoden und Werkzeuge für die IT-Praxis: Eine Einführung für Studenten und Praktiker, 7th ed., Vieweg+Teubner, available: http://dx.doi.org/10.1007/978-3-8348-2428-8, Wiesbaden, Germany, 2012

[61] Gaudoin, O.: Outils statistiques pour la construction et le choix de modèles en fiabilité des logiciels, thèse de doctorat, Université de Joseph Fourier – Grenoble 1, France, 1990

[62] Gembrys, S.-N.; Herrmann, Joachim: Qualitätsmanagement, Taschen-Guide, Haufe, vol. 137, Planegg, München, Germany, 2007

[63] Göltz, A.: Murphy´s Law, murphyslaws.net, available: http://murphyslaws.net/, accessed on: Jul. 06 2016, 1996/97

[64] Gscheidle, R.; Fischer, Richard: Fachkunde Kraftfahrzeugtechnik, Europa-Fachbuchreihe für Kraftfahrzeugtechnik, 30th ed., Europa-Lehrmittel Nourney Vollmer, Haan-Gruiten, Germany, 2013

[65] Hab, G.; Wagner, Reinhard: Projektmanagement in der Automobilindustrie, Effizientes Management von Fahrzeugprojekten entlang der Wertschöpfungskette, 4th ed., Springer, available: http://dx.doi.org/10.1007/978-3-8349-4369-9, Wiesbaden, Germany, 2013

[66] Hedderich, J.; Sachs, Lothar: Angewandte Statistik, Methodensammlung mit R, 15th ed., Springer Spektrum, available: http://dx.doi.org/10.1007/978-3-662-45691-0, Berlin, Heidelberg, Germany, 2016

[67] Highsmith, J.; Cockburn, A.: Agile Software Development: The Business of Innovation, pp. 120–122, IEEE Computer, U.S., 2001

[68] Hillenbrand, M.: Funktionale Sicherheit nach ISO 26262 in der Konzeptphase der Entwicklung von Elektrik/Elektronik Architekturen von Fahrzeugen, doctoral thesis, Steinbuch series on advances in information technology, KIT Scientific Publishing; vol. 4, available: http://edok01.tib.uni-hannover.de/edoks/e01fn12/716638142.pdf, Karlsruhe, Hannover, Germany, 2012

[69] Hinsch, M.: Qualitätsmanagement in der Luftfahrtindustrie, Ein Praxisleitfaden für die Luftfahrtnorm EN 9100, Springer Vieweg, available: http://dx.doi.org/10.1007/978-3-642-53707-3, Berlin, Germany, 2014

[70] Hope, J.; Fraser, Robin: Beyond budgeting, How managers can break free from the annual performance trap, Harvard Business School Pr, Boston, U.S., 2003

[71] IABG, I.-B. m.: Fundamentals of the V-Model, Accessed on: Apr. 13 2016, Ottobrunn, Germany, 2006

[72] Iceberg Software Ltd.: SimpleKPI, [online] available: https://www.simplekpi.com/, accessed on: May 07 2016, Andover, U.K.

[73] IEEE Guide for Developing User Organization Open System Environment (OSE) Profiles, 1998

[74] Imagix: [online] available: http://www.imagix.com/products/source-code-analysis.html, accessed on: Jun. 10 2015, San Luis Obispo, U.S.

[75] InetSoft: [online] available: https://www.inetsoft.com/, accessed on: Jan. 02 2016, New Jersey, U.S.

[76] Institut für Technologie und Arbeit: [online] available: http://www.optimus-spitzencluster.de/entwicklungsprozessvorgehensmodellnachvdi.pdf, accessed on: May 09 2016, Germany

[77] Institut für Technologie und Arbeit: Leitprojekt OPTIMUS-SC, [online] available: http://www.optimus-spitzencluster.de/qualitygateszurunterstuetzungdesprojektmanagements.pdf, accessed on: Aug. 06 2016, Germany

[78] Institute of Electrical and Electronics Engineers: 610.12-1990 IEEE Standard Glossary of Software Engineering Terminology, IEEE / Institute of Electrical and Electronics Engineers Incorporated, U.S

[79] Institute of Electrical and Electronics Engineers: IEEE standard glossary of software engineering terminology (IEEE Std 610.12-1990), U.S., 1990

[80] International Organization for Standardization: ISO Management System Standards, [online] available: http://www.iso.org/iso/home/standards/management-standards/mss-list.htm, accessed on: May 30 2016, Geneva, Switzerland

[81] International Organization for Standardization: ISO/IEC 15504-1:2004 Information technology -- Process assessment, Geneva, Switzerland, 2004

[82] International Organization for Standardization: ISO 9004:2009 Managing for the sustained success of an organization - A quality management approach, Geneva, Switzerland, 2009

[83] International Organization for Standardization: ISO/TS 16949:2009 Quality management systems -- Particular requirements for the application of

ISO 9001:2008 for automotive production and relevant service part organizations, Geneva, Switzerland, 2009
[84] International Organization for Standardization: ISO/IEC/IEEE 24765:2010 - Systems and Software Engineering — Vocabulary, Geneva, Switzerland, 2010
[85] International Organization for Standardization: ISO 26262-1:2011 Road vehicles - Functional safety, Geneva, Switzerland, 2011
[86] International Organization for Standardization: ISO/IEC 25000:2014 Systems and software engineering -- Systems and software Quality Requirements and Evaluation (SQuaRE) -- Guide to SQuaRE, Geneva, Switzerland, 2014
[87] International Organization for Standardization: ISO/IEC 90003:2014 Software engineering -- Guidelines for the application of ISO 9001:2008 to computer software, Geneva, Switzerland, 2014
[88] International Organization for Standardization: ISO 9000 - Quality management systems - Fundamentals and vocabulary, Geneva, Switzerland, 2015
[89] International Organization for Standardization: ISO 9001:2015 Quality management systems — Requirements, Geneva, Switzerland, 2015
[90] Isermann, R.: Fahrdynamik-Regelung, Modellbildung, Fahrerassistenzsysteme, Mechatronik ; mit 28 Tabellen, Kraftfahrzeugtechnik, 1st ed., Friedr. Vieweg & Sohn, GWV Fachverlage GmbH, available: http://site.ebrary.com/lib/alltitles/docDetail.action?docID=10192028, Wiesbaden, Germany, 2006
[91] Ishikawa, K.: Guide to Quality Control, Asian Productivity Organization, JUSE Press Ltd., Tokyo, Japan, 1968
[92] ITEM Software, I.: Reliability Prediction Basics, [online] available: http://www.reliabilityeducation.com/ReliabilityPredictionBasics.pdf, Accessed on: Nov. 01 2015, 2007
[93] Jung, B.; Schweißer, S.; Wappis, Johann: 8D und 7STEP, Systematisch Probleme lösen, 1st ed., Carl Hanser Fachbuchverlag, available: http://www.hanser-elibrary.com/isbn/9783446425712, Germany, 2011
[94] Junglas, M.: Methodische Entwicklung hochintegrierter mechatronischer Systeme unter funktionalen, zuverlässigkeits- und sicherheitstechnischen Aspekten - Analyse und Quantifizierung, doctoral thesis, Fakultät für Ingenieurwissenschaften, Abteilung Maschinenbau und Verfahrenstechnik, University Duisburg-Essen, Germany, 2012
[95] Kaimer, S.: Actimax Transfer Case Family - Modular and Scalable Product Platform, EAWD Congress, Graz, Austria, 2015

[96] Kazmi, A.: Strategic Management, 4th edition, Mc Graw Hill Education, U.S., 2015

[97] Kiuwan: [online] available: https://www.kiuwan.com/, accessed on: May 10 2015, Maitland, U.S.

[98] Klausnitzer, R.: Das Ende des Zufalls, Wie Big Data uns und unser Leben vorhersagbar macht, 1st ed., Ecowin, Salzburg, Austria, 2013

[99] Klostermann, M.: MKS Handbook - Introduction 2, MAGNA Powertrain AG & Co KG, Lannach, Austria

[100] Klostermann, M.; Pfeffer, E.; Mandic, I.; Münzer, A.: Anforderungen KMI für FSI, MAGNA Powertrain AG & Co KG, Lannach, Austria, 2015

[101] Kneuper, R.: Was ist eigentlich Prozessqualität?, Software Engineering Institute, [online] available: http://www.kneuper.de/Publikationen/Vortraege/Prozessqualitaet-GI2011.pdf, accessed on: Aug. 06 2016, Germany, 2011

[102] KPIFire: [online] available: https://www.kpifire.com/, accessed on: May 07 2016, Provo, U.S.

[103] Kraftfahrt-Bundesamt: Jahresbericht 2012, Germany, 2013

[104] Kremer, H.,-J.: Das Messdatenmanagement optimieren. Nicht nur für Automobilhersteller: große Fortschritte bei openMDM, Bd. 60. 0013-5658., TEMA - Technik und Management, 2011

[105] Kremer, H.,-J.: Testergebnisse für den Produktlebenszyklus erschließen. Messdatenmanagement mit openMDM, Bd. 19., beam, Marburg, Germany, 2011

[106] Kronthaler, F.: Statistik angewandt, Datenanalyse ist (k)eine Kunst, Springer-Lehrbuch, Springer Spektrum, available: http://dx.doi.org/10.1007/978-3-642-53740-0, Berlin, Germany, 2014

[107] Kügler, P., M.: Analyse von Methoden für Entwicklung, Produktion und Einsatz von technischen Produkten, bachelor thesis, Institute of Automotive Engineering, Graz University of Technology, Graz, Austria, 2016

[108] Lach, S.; Polly, Sebastian: Produktsicherheitsgesetz, Leitfaden für Hersteller und Händler, Springer Fachmedien Wiesbaden, available: http://gbv.eblib.com/patron/FullRecord.aspx?p=3109918, Wiesbaden, Germany, 2015

[109] Lindemann, U.: Methodische Entwicklung technischer Produkte, Methoden flexibel und situationsgerecht anwenden, VDI, Springer, available: http://dx.doi.org/10.1007/b137764, Berlin, Heidelberg, Germany, 2005

[110] Liu, D.: Systems engineering, System design principles and models, CRC Press, Boca Raton, Florida, U.S., 2015, cop. 2016

[111] Maalej, W.; Thurimella, Anil Kumar: Managing requirements knowledge, Springer, available: http://www.loc.gov/catdir/enhancements/fy1410/2013940938-b.html, Berlin, Germany, 2013

[112] Mamrot, M.; Marchlewitz, S.; Winzer, P.: Verknüpfung des Prozess- und Produktsystemmodells zur modellbasierten Felddatenrückführung, Bergische Universität Wuppertal, Germany, 2012

[113] Manko, J.: Konzeptionelle Vergleichbarkeitsanalyse von Entwicklungs- und Felddaten, bachelor thesis, Institute of Automotive Engineering, Graz University of Technology, Graz, Austria, 2015

[114] Meeker, W., Q.; Escobar, L., A.: Statistical methods for reliability data, Wiley series in probability and statistics. Applied probability and statistics section,Wiley, available: http://site.ebrary.com/lib/alltitles/docDetail.action?docID=10418978, New York, U.S., 1998

[115] Meinke, K.: Software Lifecycle Models, School of Computer Science and Communication, KTH Royal Institute of Technology, [online] available: http://www.nada.kth.se/~karlm/prutt05/lectures/prutt05_lec6.pdf, accessed on: Jun. 27 2016, Stockholm, Schweden, 2006

[116] Mertens, P.: Prognoserechnung, 7th ed., Physica, Heidelberg, Germany, 2012

[117] Meyer, M.; Meyna, A.; Pauli, Bernhard: Zuverlässigkeitsprognose für Kfz-Komponenten bei zeitnahen Garantiedaten, ATZ Automobiltech Z, vol. 105, no. 3, pp. 262–267, Germany, 2003

[118] Meyna, A.; Pauli, Bernhard: Taschenbuch der Zuverlässigkeits- und Sicherheitstechnik, Quantitative Bewertungsverfahren ; mit 37 Tabellen und 98 Beispielen mit Lösungen, Praxisreihe Qualitätswissen, Hanser, Munich, Germany, 2003

[119] Mizuno, S.: QFD, The customer-driven approach to quality planning and deployment, Asian Productivity Organization,Tokyo, Tokyo, Japan, 1994

[120] Müller, C.; Denecke, Liesa: Stochastik in den Ingenieurwissenschaften, Eine Einführung mit R, Statistik und ihre Anwendungen, Springer Vieweg, Berlin, Germany, 2013

[121] Müller, E.: Qualitätsmanagement für Unternehmer und Führungskräfte, Was Entscheider wissen müssen, Springer Gabler, Berlin, Germany, 2014

[122] Myers, R. H.: Generalized linear models, With applications in engineering and the sciences, Wiley series in probability and statistics, 2nd ed., Wiley, available: http://lib.myilibrary.com/detail.asp?ID=344602, Hoboken, New Jersey, U.S., 2010

[123] Naunheimer, H.; Bertsche, B.; Lechner, G.; Ryborz, Joachim: Fahrzeuggetriebe, Grundlagen, Auswahl, Auslegung und Konstruktion, 2nd ed., Springer, available: http://site.ebrary.com/lib/alltitles/docDetail.action?docID=10189346, Berlin, Heidelberg, Germany, 2007

[124] Nelder, J., A.; Wedderburn, R., W., M.: Generalized Linear Models, Journal of the Royal Statistical Society. Series A (General), vol. 135, no. 3, pp. 370–384, Wiley for the Royal Statistical Society, Hoboken, New Jersey, U.S., 1972

[125] Obermeier, S.; Fischer, H.; Fleischmann, A.; Dirndorfer, Max: Geschäftsprozesse realisieren, Ein praxisorientierter Leitfaden von der Strategie bis zur Implementierung, SpringerLink, 2nd ed., Springer Vieweg, available: http://dx.doi.org/10.1007/978-3-8348-2303-8, Wiesbaden, Germany, 2014

[126] Ōno, T.; Bodek, Norman: Toyota production system, Beyond large-scale production, Productivity Press, New York, U.S., 2008

[127] O'Regan, G.: Introduction to software quality, Undergraduate topics in computer science, Springer, Cham, Germany, 2014

[128] Pahl, G.; Beitz, W.; Feldhusen, J.; Grote, Karl-Heinrich: Konstruktionslehre, Grundlagen erfolgreicher Produktentwicklung ; Methoden und Anwendung, 7th ed., Springer, available: http://dx.doi.org/10.1007/978-3-540-34061-4, Berlin, Heidelberg, Germany, 2007

[129] Parmenter, D.: Key performance indicators, Developing, implementing, and using winning KPIs, John Wiley & Sons, available: http://search.ebscohost.com/login.aspx?direct=true&scope=site&db=nlebk&db=nlabk&AN=190542, Hoboken, New Jersey, U.S., 2007

[130] Passegger, W.: Datenmanagement für Reklamationen und Befundungsinspektionen in der Automobilindustrie, master thesis, Institute of Automotive Engineering, Graz University of Technology, Graz, Austria, 2016

[131] Passegger, W.; Ernst, M.; Pferscher, A.; Hofer, C.; Fabian, Jürgen: Data storage, management and analysis for complaints and teardown inspections – Proposal of analysis methods, internal document, unpublished, Institute of Automotive Engineering, Graz University of Technology, Graz, Austria, 2016

[132] Pauli, B.; Meyna, Arno: Zuverlässigkeitsprognosen für Kfz-Komponenten bei unvollständigen Daten, ATZ Automobiltechnische Zeitschrift, vol. 102, no. 12, pp. 1104–1107, Germany, 2000

[133] Pecnik, H.: History / Driver Advantages / Vehicle Categories / AWD systematics, EAWD Congress, Graz, Austria, 2015

[134] Perner, P.: Bewertung von Zuverlässigkeitsmodellen für automotive Software, bachelor thesis, Institute of Automotive Engineering, Graz University of Technology, Graz, Austria, 2014

[135] Pfeifer, T. e. a.: Untersuchung zur Qualitätssicherung, Stand und Bewertung - Empfehlung für Maßnahmen. Forschungsbericht KfK-PFT 155, Kernforschungszentrum Karlsruhe, Germany, 1990

[136] Pferscher, A.; Hofer, C.; Ernst, M.; Fabian, Jürgen: Data storage, management and analysis for complaints and teardown inspections, research report, FTG-2016/05, Institute of Automotive Engineering, Graz University of Technology, Graz, Austria, 2016

[137] Pickl, T., G.: Untersuchung und Evaluierung von Analysetools, bachelor thesis, Institute of Automotive Engineering, Graz University of Technology, Graz, Austria, 2015

[138] Prefi, T.: WZL - RWTH University Aachen, TFB 57, [online] available: http://www.tfb57.rwth-aachen.de/de/e9fc75581b61155ac125709e0026af68/Vortrag1_2_Prefi.pdf, accessed on: Aug. 06 2016, Germany, 2004

[139] Prodinger, M.: Klassifizierung und Anwendung von Lizenzmodellen für Analysetools, Institute of Automotive Engineering, Graz University of Technology, Graz, Austria, 2015

[140] Project Management Institute: A guide to the project management body of knowledge, (PMBOK guide), 5th ed., available: http://proquest.tech.safaribooksonline.de/9781935589679, U.S.

[141] PWC: Industry 4.0: Building the digital enterprise, [online] available: https://www.pwc.com/gx/en/industries/industries-4.0/landing-page/industry-4.0-building-your-digital-enterprise-april-2016.pdf, accessed on: Dec. 06 2016, 2016

[142] Quehenberger, J.: All-Wheel-Drive Couplings & Controls, EAWD Congress, Graz, Austria, 2015

[143] Raubold, U.: Lebenszyklusmanagement in der Automobilindustrie, Ein Optimierungsansatz auf Basis der auf den Lebenszyklus wirkenden Einflussfaktoren, Cottbus, University of Technology, doctoral thesis, 2010, Beiträge zur Produktionswirtschaft, 1st ed., Gabler, Springer Fachmedien, available: http://dx.doi.org/10.1007/978-3-8349-6613-1, Wiesbaden, Germany, 2011

[144] Rausand, M.; Høyland, A.: System reliability theory, Models, statistical methods, and applications, Wiley series in probability and statistics Applied probability and statistics, 2nd ed., Wiley-Interscience, available:

http://www.loc.gov/catdir/bios/wiley046/2003057631.html, Hoboken, New Jersey, U.S., 2004

[145] Rehermann, J.; Blumenau, M.: datapine, [online] available: http://www.datapine.com/, accessed on: Jun. 01 2016, Berlin, Germany

[146] Reif, K.: Konventioneller Antriebsstrang und Hybridantriebe, Bosch Fachinformation Automobil, Vieweg+Teubner, GWV Fachverlage GmbH, available: http://dx.doi.org/10.1007/978-3-8348-9711-4, Wiesbaden, Germany, 2010

[147] Reif, K.: Bosch Autoelektrik und Autoelektronik, Bordnetze, Sensoren und elektronische Systeme ; mit 43 Tab, Bosch Fachinformation Automobil, 6th ed., Vieweg+Teubner, Springer Fachmedien, available: http://dx.doi.org/10.1007/978-3-8348-9902-6, Wiesbaden, Germany, 2011

[148] Reif, K.: Sensoren im Kraftfahrzeug, Bosch Fachinformation Automobil, 2nd ed.,Vieweg+Teubner, available: http://dx.doi.org/10.1007/978-3-8348-2208-6, Wiesbaden, Germany, 2012

[149] Reif, K.: Automotive Mechatronics, Automotive Networking, Driving Stability Systems, Electronics, Bosch Professional Automotive Information, Springer Fachmedien, available: http://dx.doi.org/10.1007/978 -3-658-03975-2, Wiesbaden, Germany, 2015

[150] Reiter, H.: Reduktion von Integrationsproblemen für Software im Automobil durch frühzeitige Erkennung und Vermeidung von Architekturfehlern, Technical University of Munich, Germany, 2010

[151] Rigdon, S. E.; Basu, A., P.: Statistical methods for the reliability of repairable systems, A Wiley-Interscience publication, Wiley, available: http://www.loc.gov/catdir/bios/wiley043/99057532.html, New York, U.S., 2000

[152] Rinne, H.; Mittag, H.-J.: Statistische Methoden der Qualitätssicherung, 3rd ed., Hanser, Munich, Germany, 1995

[153] Royce, W., W.: Managing the development of large software systems, pp. 1–9, TRW, Proceedings of IEEE Westcon, 1970

[154] Ruge, P.; Birk, C.; Wermuth, Manfred: Das Ingenieurwissen: Mathematik und Statistik, Ingenieurwissen, Springer Vieweg, available: http://dx.doi.org/10.1007/978-3-642-40474-0, Berlin, Germany, 2014

[155] Schäfer, A.: Grundlagen der Zuverlässigkeitstechnik, Modul T7 Softwarezuverlässigkeit, Technical University of Munich, Germany, 2011

[156] Schäuffele, J.; Zurawka, Thomas: Automotive Software Engineering, Grundlagen, Prozesse, Methoden und Werkzeuge effizient einsetzen, ATZ / MTZ-Fachbuch, 5th ed., Springer, available: http://dx.doi.org/10.1007/978-3-8348-2470-7, Wiesbaden, Germany, 2013

[157] Schlummer, M. H.: Beitrag zur Entwicklung einer alternativen Vorgehensweise für eine Proven-in-Use-Argumentation in der Automobilindustrie, doctoral thesis, Bergische Universität Wuppertal, Germany, 2012

[158] Schlund, S.; Winzer, Petra: DeCoDe-Modell zur anforderungsgerechten Produktentwicklung, Gabler Research, pp. 277–293, 1st ed., Gerhard Bandow, and Holzmüller, Hartmut H., Das ist gar kein Modell!, Unterschiedliche Modelle und Modellierungen in Betriebswirtschaftslehre und Ingenieurwissenschaften, Gabler, Wiesbaden, Germany, 2010

[159] Schmitt, R.: Lehrstuhl für Fertigungsmesstechnik und Qualitätsmanagement, Werkzeugmaschinenlabor WZL der RWTH Aachen, Fraunhofer IPT, [online] available: http://www.wzl.rwth-aachen.de/www_files/Produktmanagement_am_WZL_DE.pdf, accessed on: Mar. 06 2016,

[160] Schmitt, R.; Pfeifer, Tilo: Qualitätsmanagement, Strategien, Methoden, Techniken, 4th ed., Carl Hanser Fachbuchverlag, available: http://www.hanser-elibrary.com/action/showBook?doi=10.3139/9783446426474, Munich, Germany, 2010

[161] Schnellbach, A.; Werdich, M.: FMEA Evolution, FMEA-Forum Osnabrück, Germany, 2012

[162] Schnurr, R.: Six Sigma Black Belt, [online] available: http://www.sixsigmablackbelt.de/fehlerkosten-10er-regel-zehnerregel-rule-of-ten, accessed on: Nov. 06 2015, Ottenhöfen, Germany

[163] Schuh, G.: Innovationsmanagement, Handbuch Produktion und Management 3, VDI-Buch, 2nd ed., Springer, available: http://dx.doi.org/10.1007/978-3-642-25050-7, Berlin, Heidelberg, Germany, 2012

[164] Schwaber, K.: SCRUM Development Process, Oopsla conference Austin, Texas, U.S., 1995

[165] Schwaber, K.; Sutherland, J.: The Scrum Guide, [online] available: http://www.scrumguides.org/docs/scrumguide/v2016/2016-Scrum-Guide-US.pdf#zoom=100, accessed on: Feb. 08 2016, 2016

[166] Section of Safety Theory and Traffic Engineering, University of Wuppertal: [online] available: http://sitheorie.uni-wuppertal.de/FG/index.php?id=121&L=3, accessed on: Dec. 08 2016, Wuppertal, Germany

[167] Shewhart, W. A.: Economic control of quality of manufactured product, American Society for Quality Control, Milwaukee, U.S., 1980

[168] Shingo, S.; Dillon, A. P.; Bodek, Norman: Zero quality control, Source inspection and the poka-yoke system, Productivity Pr, Cambridge, U.S., 1986

[169] Sibbertsen, P.; Lehne, Hartmut: Statistik, Einführung für Wirtschafts- und Sozialwissenschaftler, Springer-Lehrbuch, Springer, available: http://dx.doi.org/10.1007/978-3-540-36476-4, Berlin, Heidelberg, Germany, 2012

[170] Sisense: [online] available: http://www.sisense.com/, accessed on: Feb. 02 2016, New York, U.S.

[171] Software Diagnostics: [online] available: http://www.softwarediagnostics.com/, accessed on: Jun. 10 2015, Potsdam, Germany

[172] Software Engineering Institute: CMMI® for Development, Version 1.3, Improving processes for developing better products and services, technical report,, [online] available: http://www.sei.cmu.edu, accessed on: Oct. 18 2016, Pittsburgh, U.S.

[173] Spark, N. T.; Abrahams, Marc: A History of Murphy's Law, [Lulu.com], Morrisville, U.S., 2006

[174] Squoring Technologies SAS: [online] available: http://www.squoring.com/en/produits/squore-automobile/, accessed on: Jul. 10 2015, Toulouse, France, Gilching, Germany

[175] Stage-Gate International: [online] available: http://www.stage-gate.com/resources_stage-gate_full.php, accessed on: Jan. 06 2016, U.S., 2000-2016

[176] Statista GmbH: [online] Available: http://de.statista.com, accessed on: Sep. 03 2016, Germany

[177] Steinkellner, M.: Entwicklungs- und Testdatenanalyse automotiver Software - Mitentwicklung eines Analysetools, bachelor thesis, Institute of Automotive Engineering, Graz University of Technology, Graz, Austria, 2016

[178] Sturz, W.: Life Cycle Management. Von der Produktidee bis zum Produktrecycling, Steinbeis Transferzentrum für Wissensmanagement & Kommunikation, Berlin, Germany, 2005

[179] Taucraft Ltd.: Targetprocess, [online] available: https://www.targetprocess.com/product/, accessed on: May 10 2015, New York, U.S.

[180] Thaller, G. E.: Software-Test, Verifikation und Validation, Heise, Hannover, Germany, 2000

[181] The Association of German Engineers: VDI- 2221 Methodik zum Entwickeln und Konstruieren technischer Systeme und Produkte, Düsseldorf, Germany, 1993

[182] The Association of German Engineers: VDA 6 - Qualitätsstandard der Automobilindustrie, Germany, 1995

[183] The Association of German Engineers: VDI 2206, Design methodology for mechatronic systems, Düsseldorf, Germany, 2004

[184] The Federal Ministry of Defence: [online] available: https://www.bmvg.de/portal/a/bmvg, accessed on: Oct. 18 2016, Germany

[185] The Standish Group: Chaos Report, [online] available: https://www.projectsmart.co.uk/white-papers/chaos-report.pdf, accessed on: Oct. 18 2016, U.K., 1995

[186] T-systems International GmbH: [online] available: https://www.t-systems.com/blob/44922/0d2a02c4e9197f4ad75c49ec544ea239/dlo-big-data-connected-car-data.pdf, accessed on: Jun. 20 2016, Germany

[187] van Venrooy, M.: Produktrückrufe in der deutschen Automobilindustrie, Motivations-, Kommunikations- und Verständnisproblematiken, Chemnitz University of Technology, doctoral thesis, 2014, Markenkommunikation und Beziehungsmarketing, Springer Gabler, available: http://dx.doi.org/10.1007/978-3-658-09499-7, Wiesbaden, Germany, 2015

[188] Wagner, S.: Software Product Quality Control, Springer, available: http://dx.doi.org/10.1007/978-3-642-38571-1, Berlin, Heidelberg, Germany, 2013

[189] Werdich, M.: FMEA - Einführung und Moderation, Durch systematische Entwicklung zur übersichtlichen Risikominimierung (inkl. Methoden im Umfeld), 2nd ed., Vieweg+Teubner, available: http://dx.doi.org/10.1007/978-3-8348-2217-8, Wiesbaden, Germany, 2012

[190] Westkämper, E.: Produkt Life Cycle Management als Beitrag für ein nachhaltiges Wirtschaften, EXPO 2000 - Global Dialogue, Science and Technology - Thinking the Future, Universität Stuttgart, Institut für Industrielle Fertigung und Fabrikbetrieb, Stuttgart, Germany, 2001

[191] Wilfinger, M.: Interaktion mechatronischer Antriebsstrangkomponenten, bachelor thesis, Institute of Automotive Engineering, Graz University of Technology, Graz, Austria, 2015

[192] Witter, A.: Entwicklung eines Modells zur optimierten Nutzung des Wissenspotentials einer Prozeß-FMEA, Hannover, Universiy, doctoral thesis, Fortschritt-Berichte VDI Reihe 20, Rechnerunterstützte Verfahren, VDI, vol. 176, Düsseldorf, Germany, 1995

[193] Wolf, J.: Entwicklung eines Analysetools in VisualBasic, bachlor thesis, Institute of Automotive Engineering, Graz University of Technology, Graz, Austria, 2016

[194] Wrike Inc.: [online] available: https://www.wrike.com/de/, accessed on: May 10 2015, San, Jose, U.S., 2006-2016