ISBN 978-93-84422-37-0
2015 International Conference on Advances in Software, Control and Mechanical Engineering (ICSCME'2015)
Antalya (Turkey) Sept. 7-8, 2015 pp. 1-5

Optimisation of Automotive Software Quality Management by Use of Analysis Methods in the Development Process of Mechatronic Systems

Markus Ernst[1], Stefan Erlachner[1], Mario Hirz[1] and Jürgen Fabian[1]
[1]Graz University of Technology, Institute of Automotive Engineering,
Inffeldgasse 11/II, 8010 Graz, Austria,

***Abstract*:** *Due to a steady increase of mechatronic applications in automotive technology and various interactions of systems, modules and components, efficient quality management and enhanced development processes are strongly required. In this context, automotive manufacturers and suppliers are continuously faced to avoid and decrease faults in mechatronic systems. When dealing with automotive software, the number of electronic control units (ECUs) and the variety of implemented functions and complex interactions represent a major key challenge. Therefore, fault prevention by introducing new analysis methods based on objective quality evaluations and specific metrics already have to be applied in early development phases. The application of specific analysis methods is able to support an objective quality evaluation and risk-analysis for the release process of automotive software. The present publication includes an approach, which enables an objective analysis of the development process by the use of stochastic methods to gain an estimation of software reliability.*

Keywords: *automotive software, automotive mechatronics, quality management*

1. Introduction

Steady growing numbers of functionalities and electronic control units lead to increasing complexity in automotive mechatronics, whereby the number of failures tends to increase also with complexity and functional content. One of the advantages of state of the art mechatronic systems is the flexibility to adapt to boundary conditions. Therefore, a part of the functions and its precision becomes changeable and programmable within very short timelines. This also includes a faster time-to-market, because basic elements can be developed in parallel, whereby the functional integration results from the software, [1], [2]. Besides the advantages of modern mechatronic systems, different internal and external failures can be lethal to the operating devices, leading to rising reliability concerns. Faults represent any type of malfunction within a system, leading to an undesired performance and to an inability to fulfil the intended purpose, [3]. Failure modes of mechatronic systems can result from mechanical, electrical, computational and control subsystems, classified as hardware and software failures, [4]. Any fault should be prevented by suitable development process, especially considering functional safety faults are an unacceptable deviation and can lead to uncontrollable circumstances. In this context, a reduction of software failures, which are closely connected to failures within the complete mechatronic systems, plays an important role. For this purpose, effective quality management states an important task in automotive software development to avoid fail operation in customer use. To reach this target, possible malfunctions have to be detected and avoided from early development phases on, which requires effective and efficient analysis methods.

2. Challenges in the development process and the evaluation of automotive software

With target to fulfil the huge number of requirements in automotive mechatronics system development, different types of qualitative and quantitative analysis methods have to be applied. The guideline VDI 2206 (Design Methodology for Mechatronic Systems) [4] delivers a basis for the development process of mechatronic systems. Besides the V-model at macro-level and the general problem-solving cycles at micro-level, the development process is also split into interdisciplinary subareas of mechatronic systems, precisely mechanical, electrical and IT-related development phases. Whereas the left side of the V-model shows the design and preparation steps from system, over subsystem to component level, the right side focusses on different types of testing procedures, e.g. Mil, Sil, Hil, and hardware-based investigations, as well as the implementation of subareas into the complete mechatronic system. In view of quality improvement the evaluation and control of requirements, tests and fault handling during the development process plays an essential role. Focusing software development under safety-related aspects, at it is included in different guidelines, such as ISO 26262 (Road Vehicles – Functional Safety) [5], a permanent check of requirements and consideration of safety-relevant aspects has to be done. Typical qualitative approaches are FMEA (Failure Mode and Effects Analysis), FTA (Fault Tree Analysis) and SQMA (Simulation-based Qualitative Modelling and Analysis). These methods are based on system architecture related data acquisition, modelling of incomplete or imprecise information, as well as the continuous consideration during the development process and provide prognosis of malfunctions, risks and effects of potential failures. With the help of propagation models and the consideration of effects, possible faults are detected, evaluated and measures for improvements are derived. Quantitative methods concern statistically and stochastically approaches to determine probability of variables. So-called regression models enable a reliability prediction of systems and the derivation of lifetime distribution out of test data. Depending on the applied methods, different influencing factors, e.g. load cases, environmental impacts, boundary conditions and system complexity are modelled to describe hazard behaviour. Limitations of these traditional analysis methods can be found in restricted possibilities for complex systems due to restricted consideration of multiple-failure effects, limited possibilities of crosslinking of components and drawbacks in the description of signal and data flows, as they occur in automotive applications. The main target in the development of new analysis approaches is to optimise workflows and processes, as well as to minimise the residual error rate in field-use of mechatronic systems, under consideration of all requirements regarding complexity and interactions.

3. Analysis Methods for Automotive Software Quality Management

One challenge of mechatronic system development lies in the integration of mechanical, electrical and software components. Faults due to mechanical reasons appear because of attrition or dimensioning faults, by contrast deviations due to automotive software occur because of undetected residual errors. By keeping track of minimising the residual error rate in field-use issued by automotive software, fault prevention by introducing new analysis methods has to be done as soon as possible in development process. These new analysis methods base on objective quality evaluations and specific metrics, and demand a continuous monitoring of requirements, tests and deviations during the entire software development. Different parties are involved in the development of mechatronic systems, e.g. quality assurance, project management and general management. By implementing analysis methods, the different views and questions of those areas due to the development process have to be considered, as shown in **Error! Reference source not found.**.

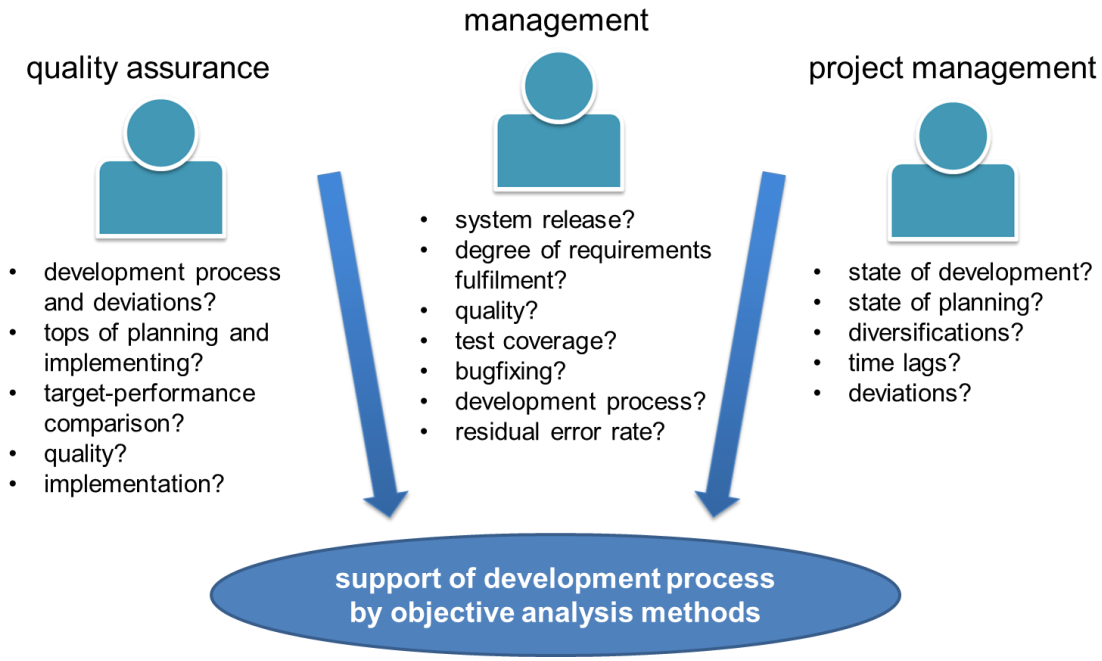

Fig. 1: different views and tasks during the development process

By analysing these attributes, their chronological sequence in the development process is important. For this purpose the progress of planning elements, deviations in view of possible time lags and target-performance comparisons deliver objective statements. One important question includes the possibility of releasing a software status to a certain point of time. Hence, effective strategies have to provide a risk-analysis by use of stochastic methods to enable objective quality evaluation and an estimation of software reliability.

An approach of risk analysis in the process of releasing automotive software by the use of analysis methods is shown in **Error! Reference source not found.**. In the first steps, a combination of defined risk levels in a traffic light matrix and a target-performance comparison resulting in a set of indicators supports a fast evaluation of the possibility for releasing the system. Special attention has to be paid to functional safety relevant aspects. As knockout criteria, all planning elements, tests or faults linked to functional safety have to be implemented or passed correctly for a possible system release. Considering the residual error rate, zero faults are only theoretically attainable with an infinite number of perfect tests, because every specification would need a test against every malfunction which is not possible in profit oriented industry. Even if the test strategy includes MiL, SiL and HiL levels, especially mechatronic systems especially deliver a challenge to get test conditions close to reality. Hence, in practice the residual error rate at a certain point of time is very significant. For this purpose the next step in **Error! Reference source not found.** shows a modelling of faults issued by testing which enables a definition of the residual error rate, a trend analysis of the further development and a support for defining the end of testing.

In case of automotive software a complex basic approach of modelling has to be chosen, which enables the consideration of a variable mean time between failures, a changing fault rate and a permanent repairing process. This is possible by the use of the non-homogenous poisson process (NHPP) with a power law intensity function to simulate a realistic process in automotive software engineering, [6]:

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

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

The parameter estimation could be done using maximum likelihood method, [7]:

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

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

The inputs of the modelling are the detected faults during the development process. To define the confidence interval of the model distribution a fisher information matrix could be used to calculate the expected maximum, minimum and average residual error rate at a system release point, [8]. Considering the different number of implemented requirements and executed tests over various software releases, modelling have to be done at every certain point of time, such as release milestones. This multiple recharging enables an additional step to realistic failure modelling and a better prognosis of the residual error rate. Beside that the modelling allows a trend analysis of the further development tests and the possibility of end of test definition. At least the approach provides a combination of process and software metrics helps investigating the software quality. For this purpose a set of software and process indicators are aggregated to four categories, such as quality, quantity, complexity and productivity, shown by a net diagram to visualise the software time transformation. This approach as a combination of various analysis methods supports the decision of the possibility to release the automotive software to a certain point of time.

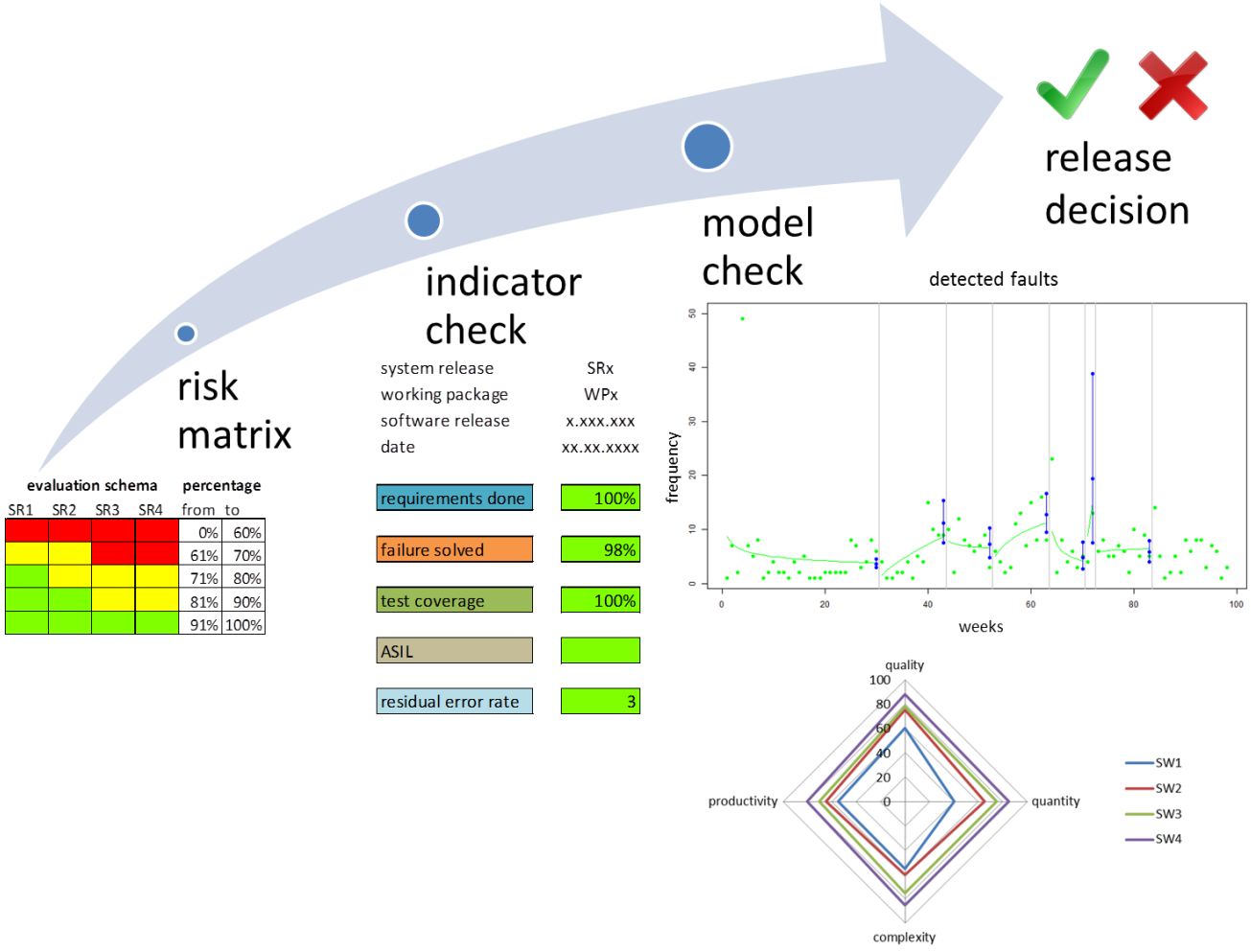

Fig. 2: approach of risk analysis in automotive software release process

4. Conclusion

In the automotive industry manufacturers and their suppliers are permanently faced with the reduction of faults in mechatronic systems, especially related to functional safety aspects. The complex interaction of systems and subcomponents represents an additional challenge. Focusing automotive software, enhanced development is supported by new comprehensive analysis and evaluation methods. To prevent faults in early development phases of automotive systems, a combination of stochastic considerations, an evaluation using metrics and a permanent target-performance comparison have to be implemented to consider different views and questions. A combination of indicators, metrics and stochastic methods deliver a risk analysis for software release process.

5. References

[1] S. Y. Nof: "Springer Handbook of Automation", Springer, ISBN 978-3-540-78830-0, 2009 http://dx.doi.org/10.1007/978-3-540-78831-7

[2] R. Isermann: "Mechatronic Systems: Fundamentals", Springer, ISBN 978-1-85233-930-2, 2005

[3] C. W. de Silva: "Mechatronic Systems: Devices, Design, Control, Operation and Monitoring", CRC Press, ISBN 0849307767, 2014

[4] Association of German Engineers (VDI): "VDI 2206 Design methodology for mechatronic systems", 2004

[5] International Organization for Standardization: "ISO 26262 Road Vehicles – Functional Safety", 2011

[6] J. T. Duane: "Learning Curve Approach to Reliability Monitoring", IEEE Transaction on Aerospace, Vol.2 No.2, 1964 http://dx.doi.org/10.1109/TA.1964.4319640

[7] H. Bauke: "Parameter estimation for power-law distributions by maximum likelihood methods", The European Pysical Journal B, Vol. 58, pp. 167-173, Springer, 2007

[8] W. Q. Meeker, L. A. Escobar: "Statistical method for reliability data", John Wiley & Sons Inc., U.S., 1998

[9] C. Radhakrishna Rao, "Linear Statistical Inverence and its Applications", John Wiley & Sons Inc., ISBN 0-471-21875-8, 1973