Challenges concerning test case specifications in automotive software testing: assessment of frequency and criticality

Katharina Juhnke^{1,2} · Matthias Tichy² · Frank Houdek¹

Published online: 18 November 2020
© The Author(s) 2020

Abstract

Automotive test case specifications document test cases to be performed for a specific test object at a defined test level. They are a fundamental part of a structured automotive testing process, as required by the ISO 26262. The aim of our research is to identify challenges from a practitioner's point of view that lead to poor quality of test case specifications and thus negatively impact time, cost, and probability of defect detection. We designed an exploratory case study to systematically identify challenges focusing on (C) creation, (P) processing, and (Q) quality assurance related aspects of test case specifications. We conducted 17 semi-structured interviews covering a German OEM as well as three of its automotive suppliers and analyzed them qualitatively. We investigated causes and consequences arising from the challenges. Additionally, we conducted a descriptive survey to assess frequency and criticality. The identified challenges were summarized in a taxonomy consisting of nine main categories: (1) availability and (2) content-related problems with input artifacts, problems related to (3) a lack of knowledge, (4) the test case description, (5) the test case specification content, (6) processes, (7) communication, (8) quality assurance, and (9) tools. The challenges were assessed by 26 internal and 10 external employees. Hence, we identified differences between these groups in terms of access to documents, incomplete requirements, scope of model series, process, and tool-related aspects. Overall, the study results underline the necessity of quality assurance measures for test case specifications. Based on the assessments, our research indicates a broad range of test case description related challenges that are promising candidates for improving test case specification quality.

Keywords Automotive software testing · Test case specification · Embedded systems · Challenges

✉ Katharina Juhnke
katharina.juhnke@daimler.com

1 Introduction

Nowadays, innovation in vehicles is mainly realized by software and electronic systems. To verify that software works as expected, testing is an integral part of the development process in the automotive domain. Standards like ISO 26262 (2011) or Automotive SPICE (2016) must be implemented by original equipment manufacturers (OEMs). In addition to a systematic development process, standards require a mandatory documentation of the activities and relevant work products. In the context of test documentation, these work products are a verification plan and a verification specification. The latter is also called test case specification in the software testing standard ISO 29119 (2013) and this term will be used in this article.

A test case specification contains a set of test cases (see example in Fig. 4) derived from the *test basis* for a particular *test object* (ISTQB 2015). For example, in order to validate system requirements using a vehicle prototype, a number of test cases would be determined for the system PRE-SAFE¹ (*test object*), a system that provides protection in an event of danger. These test cases would contain detailed descriptions of the driving maneuvers to be performed by a test driver and the corresponding reaction description (e.g., activation of belt tensioning or automatic closing function for side windows), which were derived from the system requirement specification (*test basis*). Typical test basis documents in requirements-based testing are system or component requirement specifications, use cases, or scenarios.

A high-quality test case specification is not only required in the context of ISO 26262 but also quite necessary in order to avoid resulting errors or misinterpretations in subsequent test activities. The authors (producers) of a test case specification are usually not the testers (consumers) who execute the specified test cases. In particular, this is due to the fact that there are different application areas for these test cases. For instance, test cases from a test case specification are used as the basis for implementing test scripts for test automation regarding hardware in the loop (HiL) testing or for performing manual tests using a vehicle prototype. Manual tests are also reused as acceptance tests at the end of production. Figure 1 shows these relationships between authors and the corresponding tester roles.

Test cases can be created and executed by both internal employees (working for an OEM) and external employees (working for an external engineering partner). A high-quality test case specification would ensure that the testers understand, implement, and execute the test cases exactly as the test designer intended. This can be challenging, as all participants usually have different knowledge or assumptions about the system under test (SUT) and different experiences with test techniques and test processes.

Our own experience is that we receive feedback from practitioners that faulty test cases exist and the quality of test case specifications is poor. For instance, this is indicated by a high communication effort due to testers' questions about ambiguities in the test case specifications or by incorrectly implemented test cases. A faulty test case specification entails that testing takes too much time (e.g., until ambiguities have been clarified), is too expensive (e.g., due to redundant test cases), and, in some cases, testing has no effect and no defects are detected (e.g., if test cases are implemented incorrectly). In order to avoid these consequences and to improve the quality of test case specifications, it is important to examine these reported perceptions. Therefore, we systematically investigate challenges that imply a poor test case specification quality from a practitioner's point of view in an empirical study.

¹Mercedes-Benz PRE-SAFE®: <https://www.youtube.com/watch?v=vTmLYY-Z2rc>

Fig. 1 Relationships between authors (producers) and various tester roles (consumers) regarding the test case specification

Therefore, the following two research questions will be answered in a first study, which have already been published in Juhnke et al. (2018a, b).

RQ1: What are current challenges in practice concerning test case specifications in automotive testing?

This research question focuses on the typical life cycle of a test case specification. Usually test case specifications are written by a test designer and afterwards concrete test cases are implemented and executed by a tester. Test designers and testers are usually different people, as illustrated in Fig. 1, but they do not have to be. Between these activities of creating and processing a test case specification, quality assurance activities can exist to improve the test case specification. For example, reviews can be used to detect incomprehensible or faulty test cases or to determine the requirements coverage. In the case of a high-quality test case specification, the tester as the “consumer” of a test case specification should have few queries about the contained test cases and the implemented test cases should test what the test designer intended to test. Hence, we suspect that the challenges occur particularly in the areas of: (C) creation, (P) processing, and (Q) quality assessment related aspects of test case specifications. Therefore, we considered availability and quality of input artifacts used as well as the phrasing of test cases (C). Furthermore, we focus on identifying challenges related to negative effects in downstream development activities based on decisions and faults that occurred during the creation of test case specifications (P). We investigate challenges related to the understanding of high-quality test case specifications and which quality criteria and mechanisms are already in use or could be useful to improve quality (Q).

RQ2: Which causes and consequences of the challenges are the practitioners aware of and which solutions exist?

This research question supplements research question RQ1. It is interesting to know what causes and consequences practitioners are aware of. This makes it possible to estimate the extent of a problem and, if the causes are known, to develop suitable solutions. In addition, we considered solutions proposed by the practitioners. Such insights are interesting because existing solutions may also be applicable to other practitioners or can be applied to other situations.

In order to assess the challenges identified in the first study, we conducted a second complementary study that answers the following two research questions:

RQ3: How do practitioners assess the frequency of occurrence and the criticality of the identified challenges?

This question focuses on the assessment of the identified challenges from the exploratory case study. We assume that the challenges are also known to other practitioners with similar test responsibilities as the interviewees. Therefore, this research question intends to examine the frequency of occurrence and the criticality of the challenges. The results of the assessment are interesting because they allow a prioritization of improvement activities.

RQ4: How do the identified challenges differ between external and internal employees?

We suspect differences in the occurrence with respect to the assessments of internal employees (OEM) and external employees (engineering partner). There may be challenges that are more likely to occur for the group of external employees and less for the group of internal employees and vice versa. In addition, we investigate whether there are differences in the assessment of the criticality of a challenge between internal and external employees.

Our research questions concern the research object test case specification, as an integral part of the testing process in the automotive domain. In this context, the research questions have not yet been answered comprehensively by related work. Related work with respect to the automotive domain mentions challenges concerning the increasing complexity and heterogeneous nature of software-based systems. Grimm (2003) and Pretschner et al. (2007) emphasize the effect that with increasing system complexity, the integration and testing of such systems also becomes more complex. Kasoju et al. (2013) describe challenges mainly at the level of project and test management. Furthermore, the huge number of variants and configurations, which are an aspect of the growing system complexity, are also mentioned as a challenge in automotive testing (Pretschner et al. 2007; Tierno et al. 2016). Challenges with a more specific reference to test case specifications were named by Lachmann and Schaefer (2014). Especially they highlighted natural language-based test cases to be ambiguous and incomprehensible. Moreover, an insufficient tool support is mentioned by different researchers (Grimm 2003; Petrenko et al. 2015; Broy 2006; Garousi et al. 2017). Broy (2006) stated that tools are not integrated and therefore seamless tool chains are missing, even in the context of embedded software testing. The lack of trainings was named by Garousi et al. (2017) and Kasoju et al. (2013). Overall, the need of a test methodology to handle challenges was stated by Pretschner et al. (2007). However, there exist currently no empirical studies focusing on challenges concerning automotive test case specifications. To the best of our knowledge, related work does not provide a comparable empirical study, most of them are experience reports.

Therefore, we conducted an exploratory case study (Runeson et al. 2012) and a descriptive survey (Dresch et al. 2015) at Mercedes-Benz Cars Development and in cooperation with its suppliers. In the first study, we focus on answering **RQ1** and **RQ2**. Hence, we collected data using 17 semi-structured interviews and analyzed them qualitatively after transcription (Seaman 1999). The challenges mentioned by practitioners have been clustered systematically into a taxonomy consisting of 24 different types of challenges (ToC), which have been assigned to a total of nine main categories (M1 – M9). These categories encompass availability and content quality-related problems of input artifacts, missing knowledge about testing methods or the system under test, problems regarding test case descriptions, and the content of a test case specification, as well as problems related to processes, communication, quality assurance aspects, and tools. Additionally, we presented examples of

challenges from the individual categories and for each type of challenge. The developed taxonomy of test case specification challenges is an integral part of this work.

In our second study, we focus on answering **RQ3** and **RQ4**. We collected data from 36 questionnaires and performed descriptive analyses. Based on the assessments, we derived a scatter plot showing the challenges identified according to their frequency and criticality. Moreover, we used statistical methods, such as Fisher's exact test (Fisher 1992) and the calculation of the ϕ coefficient (Cohen 1988) to indicate the effect size. This was done to analyze differences between internal and external employees. We detected significant differences with a medium effect size between the groups in terms of missing access rights to relevant documents, incomplete requirements, the presence of different model series in a test case specification, unclear interface definition to the overall process, and non-continuous tool chains. The assessments also show that the challenges identified in the first study are valid.

The remainder of this article is organized as follows: Section 2 provides an overview of the automotive test process. The research object test case specification is described and an example of a test case is shown. Section 3 presents related work focusing on challenges in automotive software testing. Section 4 describes the research methodology used in this study. Section 5 presents the evolved taxonomy of challenges concerning automotive test case specifications as the result of the exploratory case study. The results of the descriptive survey are presented in Section 6. Finally, Section 7 concludes the results of the studies and provides suggestions for future work.

2 Background

This section provides a brief overview of important aspects of the test process in automotive software development.

Testing is one of the most important parts of automotive software development, because undetected failures in software can lead to considerable financial damage or human lives loss. Automotive software testing in general follows the fundamental test process (Spillner et al. 2014), which consists of five phases: *test planning*, *test analysis & design*, *test implementation & execution*, *evaluation & reporting*, and *test closure activities* (see Fig. 2).

Fig. 2 Phases of the fundamental test process and activities of the test planning and test analysis & design phases

Overall, test management includes activities for monitoring, administration, and adaptation of the test process and covers all test phases. The individual test phases and their activities produce defined results and documents, also referred to as work products. Figure 2 illustrates this in detail for the phases *test planning* (work product: test plan) and *test analysis & design* (work product: test case specification).

Based on the V-model established in the automotive domain (Weber 2009), as shown in Fig. 3, a developed component or function has to be tested on different test levels. At each test level, specific test platforms are used for testing, such as hardware in the loop (HiL) test systems or a vehicle prototype. For each test level, there have to be specific test cases focusing on different test objectives, such as functionality or usability (cf. Automotive SPICE (2016)). This paper focuses on test case specifications, which are necessary for system validation and system integration testing (higher test levels).

Furthermore, Fig. 3 presents the responsibilities of OEMs and suppliers. OEMs are responsible for requirements specification and system design, which is the basis for commissioning. Based on this, the development of required components is usually carried out at the supplier's premises. This means that the supplier develops the source code that runs on the electronic control units (ECUs) and tests them using component or unit tests. Finally, the OEM is responsible for final integration and system testing to ensure that the implemented software meets the specified functional and safety requirements. In this context, specification-based testing and the creation of test level-specific test cases is common practice. Our research on test case specifications focuses on those relevant for the two upper test levels (cf. Fig. 3). It is also common practice for an OEM to commission an external engineering partner A to create such test case specifications based on specified requirements. The test cases can then be implemented and executed by another external engineering partner B. Therefore, test case specifications are an important outcome of the test design phase and a required part of the test documentation, not only to trace test cases to requirements and to reproduce the execution of test cases but also as an exchange format between OEM and external engineering partners or between different external engineering partners. Therefore, the test documentation must be of high quality.

The main input artifacts for writing a test case specification are defined as *test basis* (ISTQB 2015) and contain, for example, system and component requirement specifications, use cases, functional models, software architecture design, interfaces or other required documents. Furthermore, the *test plan* (also called verification specification, see (ISO 26262 2011; Nörenberg et al. 2010) for further information) is a necessary input for writing

Fig. 3 The V-model in the automotive domain with responsibilities of OEM and supplier

Test Case Basic Attributes									Test Case Metadata					
ID	Object Type	Object Text	Precondition	Action	Expected Result	Postcondition	Priority	Requirement	Test Goal	Release State	Workflow	Vehicle Family	Test Platform	
WWC test case -TS-123	Switch on KL15R	- ISw_Stat = IGN_OFF - FWS = OFF and RWS = OFF - Front and Rear Wiper in Parking Position - WIPERCONTROL_ACTIVE = 0 - Gearbox Position = P - Rain sensor coded, no rain				- None	2	WWC-REQ-567 WWC-REQ-789	Functionality	2.0	Approved	BRXYZ	Vehicle HIL	
WWC test step -TS-124	Step 1		- FWS = 0/3 - RWS = 1		WIPERCONTROL_ACTIVE = 0 - Front and rear wipers must not start wiping									
WWC test step -TS-125	Step 2		- ISw_Stat = IGN_ACC		WIPERCONTROL_ACTIVE = 1 - Rear wiper must start wiping in continuous wipe mode 1 as soon as IGN_ACC has accepted									
WWC test step -TS-126	Step 3		- ISw_Stat = IGN_OFF		WIPERCONTROL_ACTIVE = 0 - Wipers must stop as soon as IGN_OFF has accepted (delay max. 20 ms)									

FWS = Front wiper switch, RWS = Rear wiper switch

FWS = Front wiper switch, RWS = Rear wiper switch

Fig. 4 Example of a test case definition for a system integration test using a test case specification template

a test case specification. It includes definitions of test objects, test levels (e.g., model, software, hardware, component or system integration test), test objectives, test platforms (e.g., Vehicle, HiL, SiL, MiL), testing goals (e.g., usability, functionality, robustness), and thus the testing strategy. *Test case specification templates* are also a useful input artifact and contain for example a predefined set of attributes (e.g., actions, expected results, test platform, model series).

The test case specification describes the set of all test cases that are required to implement the test objectives and contents defined in the test strategy (outcome of the *test planning* phase). The test case specification itself is again a necessary input artifact for the following test phases (e.g., *test implementation & execution*). A test case is defined by: identifier, pre- and postconditions for the execution of the test case, inputs, or actions to be performed on the test item, expected results, priority for the testing, and traceability data (e.g., references to the associated requirements) (ISO 29119 2013). Figure 4 illustrates an automotive test case for a wiper and wash system using a test case specification template with a predefined attribute set. Such table-based templates are used, for example, to specify test cases in Excel or DOORS. The attributes are represented by the columns and the rows can be used to create different object types (e.g., test case and test step). Not all attributes are relevant for each object type, so some cells should not be filled (dark gray cells in Fig. 4). Templates can also be integrated as forms in graphical user interfaces, as it is done in various test case tools (e.g., HP ALM Quality Center). Other specific attributes are required in the automotive context in addition to these typical test case attributes. In the following, these are referred to as test case metadata and include, for example, information on model series, vehicle architectures, variants, release levels, or the target test platform on which the test case is to be executed. Figure 4 shows an example of test case metadata. Enriching a test case with metadata supports the reusability of a test case across different model series or variants and is required for test case selection. To support the creation of test case specifications, templates are used to specify a basic test case structure by predefineding a set of mandatory and optional attributes.

3 Related work

In this section, we discuss related work that address challenges related to software testing in general (Bertolino 2007; Garousi et al. 2017) and automotive software testing in

particular. This also includes existing work that describes influencing factors for automotive software testing (Grimm 2003; Pretschner et al. 2007; Broy 2006) as well as existing work that focuses on the test process (Kasoju et al. 2013; Sundmark et al. 2011) and especially on challenges of test case specifications (Lachmann and Schaefer 2013, 2014) in the automotive domain.

3.1 Challenges in software testing in general

For general software testing challenges, we refer to Bertolino (2007). This paper presents a roadmap of challenges that research in software testing has to address, such as the assessment of test effectiveness or the education of software testers. The focus is more on test techniques than on challenges specific to test case specifications.

Garousi et al. (2017) characterize challenges in software testing based on a survey of 105 testing practitioners from industry. The opinion of the practitioners regarding existing challenges was obtained according to nine test activities. Thereof, the test activities *test-case design*, *test scripting*, and *test tools* refer to activities in the test analysis and design phase (cf. Fig. 2) and thus to our research focus. While test-case design and test scripting were rated as rather not challenging, practitioners perceived a need for research activities in the development of better tool support for different test activities. With regard to the three mentioned test activities relevant to our research, the following challenges recorded by Garousi et al. (2017) are interesting: quality of requirements impact quality of test-case design, correlation between domain knowledge and tester skills, need for training for various test activities, manual test cases are often too long or do not contain necessary details, tracing from requirements to test cases with a meaningful metric, documentation issues on explorative testing, and the need of better tool support.

3.2 Challenges in software testing related to the automotive domain

The review of related work in software testing especially in the automotive domain reveals that research in this area is dominated by methods and tools developed for model-based testing (Tierno et al. 2016; Petrenko et al. 2015). In addition, it is also evident that there is a lot of research activity when it comes to test automation and tools in the automotive domain (Kasoju et al. 2013; Petrenko et al. 2015). For example, this is impressively demonstrated by Kasoju et al. (2013), who collected 15 sources for solving challenges related to the lack of automation for test case generation. These approaches are mainly related to lower integration levels (e.g., model or software integration tests), which are not in the focus of our research. In contrast, higher integration levels, such as system integration tests or validation based on vehicle prototypes, are considered less frequently (Bringmann and Krämer 2008; Sundmark et al. 2011).

There is a need to better understand challenges in the domain of automotive software testing. However, only a few reports exist on how software testing processes are performed in the automotive industry, whereby test case specifications are considered only marginally. Challenges in automotive software testing are influenced in particular by the characteristics of the automotive domain (Grimm 2003; Pretschner et al. 2007; Kasoju et al. 2013; Sundmark et al. 2011). Grimm (2003) mentions the control of the increasing complexity of software-based systems as a core challenge. He highlights the impact on integration and testing of complex systems as a major challenge. Due to the distributed development of a system (Grimm 2003; Pretschner et al. 2007), the maturity of the requirements specification as well as integration and testing aspects must be considered. Pretschner et al. (2007)

consider the huge number of variants and configurations within the software as challenging. They point out that an elaborative design and test methodology is required to handle this challenge. Furthermore, the heterogeneous nature of software depending on the different domains in automotive engineering (power train, body, chassis, infotainment, or safety electronics), mentioned in Pretschner et al. (2007), also poses a challenge. This requires skills from various disciplines for test case design. The influence of the domains on the development of control devices implies influences on testing. Moreover, Grimm (2003) and Broy (2006) mention that a seamless chain of methods and tools supporting the whole development cycle from requirements specification to integration and testing is missing. Frequently, each domain and in some cases each division has its own processes, methods and tools. This statement also reflects our own experiences. Kasoju et al. (2013) in particular emphasize the lack of a uniform testing process.

3.2.1 Challenges related to the automotive testing process

Challenges focusing on the entire automotive testing process have already been addressed by several researchers (Kasoju et al. 2013; Sundmark et al. 2011). Kasoju et al. (2013) studied challenges within the entire automotive testing process with the aim of helping an organization to improve its process. They identified ten areas of challenges: (1) organizational and process-related issues, (2) time and cost constraints for testing, (3) requirement-related issues, (4) resource constraints for testing, as well as issues related to (5) knowledge management, (6) communication, (7) testing techniques and tools, (8) quality aspects, (9) defect detection, and (10) documentation. They stated that the identified challenges are more or less related to test or project management issues. For instance, 8 out of 10 challenge areas are assigned to a process area that refers to management topics. Although it is a very comprehensive view of the test process, they have not explicitly examined challenges related to test case specifications. With respect to our research focus, we see especially correlations with the challenge area (10) documentation-related issues, where test cases are not continuously updated.

Sundmark et al. (2011) focus on system testing in the automotive release process and identified challenges related to integrating system testing activities into the release process. They highlight the need for detailed identification of areas with improvement potential. Identified challenges relate to change requests that are included late in the release process, difficulties in defining exact responsibilities of the different test levels, lack of measurements for process analysis (due to heterogeneous tools), increasing complexity, and the increasing number of variants leading to test case explosions.

3.2.2 Challenges explicitly related to test case specifications

A more detailed view of challenges related to test case specifications is provided by Lachmann and Schaefer (2013) in an experience report. They describe challenges focusing on testing driver assistance systems: missing test plan, safety requirements and the resulting enormous test effort, and unstructured test case design. The latter challenge comprises problems regarding missing requirement specifications, undefined coverage criteria, or a lack of knowledge regarding test case derivation methods. Lachmann and Schaefer (2014) outline further challenges with respect to natural language specifications that can influence the comprehensibility (e.g., due to non-defined abbreviations). They also mention problems in the tool-based detection of redundancies for natural language-based test cases because natural language processing algorithms are insufficient.

In summary, the presented related work points out some challenges that can be associated with test case specifications. We summarized these findings relevant to our research context into 25 challenges from related work ($C_{RW01} - C_{RW25}$). These challenges are shown together with the corresponding references in Table 1.

In particular, Garousi et al. (2017) present a nice overview of testing challenges related to several test activities, but those are not specific to the automotive domain. Challenges specially related to automotive software testing have been mentioned in related work by Kasoju et al. (2013) for the entire testing process and by Sundmark et al. (2011) regarding the system release process. To the best of our knowledge, there is no other empirical study investigating challenges associated with the creation and further processing of test case specifications in automotive software testing. Especially challenges related to the understanding of test case descriptions were insufficiently considered in related work. However, this aspect is important in order to understand how test cases have to be described so that in a distributed development environment (involving different test designers, testers and suppliers), misunderstandings can be avoided. The only publication (Lachmann and Schaefer 2014) addressing those challenges in automotive software testing that we found was an experience report which is not based on an empirical study. However, what is lacking so far is a more comprehensive examination of the challenges concerning test case specifications.

4 Research methodology

In order to investigate challenges concerning test case specifications, we have chosen a two-stage approach. First, we conducted an exploratory case study to identify challenges. Secondly, we conducted a descriptive survey to verify the results of the previous step. The following subsections describe the study design, data collection, data analysis and threats to validity for the exploratory case study (Section 4.1), and the descriptive survey (Section 4.2).

4.1 First study: exploratory case study

Study design. The goal of our exploratory case study is to identify challenges concerning automotive test case specifications from a practitioner's point of view. Specifically, the case that is considered is Mercedes-Benz Cars Development and its suppliers. Therefore, we conducted semi-structured interviews (Runeson et al. 2012) with test managers, test designers and testers. This method (Dresch et al. 2015) was selected to get insights into the examined topic and to identify problems to be studied related to research question RQ1. An interview guide for the semi-structured interviews was designed based on some initial open-minded interviews, literature review, and a manual analysis of existing test case specifications. The interview guide used in the interviews can be found online².

Data collection. The study was conducted with employees of Mercedes-Benz Cars Development and three different suppliers. The interview participants were selected due to their involvement in the testing process. In order to obtain multiple different points of view, we selected participants with various roles from eleven different departments. This means, they all dealt with test case specifications but for different systems under test, such as exterior light, remote start, central locking, parking assistant, power management network,

²Interview guide online: <http://dx.doi.org/10.18725/OPARU-33548>

Table 1 Overview of challenges concerning test case specifications extracted from related work

ID	Challenges	References
<i>C_{RW}</i> 01	Increasing complexity of software-based systems influences testing	Grimm (2003), Pretschner et al. (2007), and Sundmark et al. (2011)
<i>C_{RW}</i> 02	Huge number of variants and configurations (test case explosion)	Grimm (2003), Pretschner et al. (2007), and Sundmark et al. (2011)
<i>C_{RW}</i> 03	Enormous test effort (especially due to safety requirements)	Lachmann and Schaefer (2013)
<i>C_{RW}</i> 04	Heterogeneous nature of software depending on the different domains	Pretschner et al. (2007)
<i>C_{RW}</i> 05	Quality of requirements impact the quality of test-case design (e.g., due to insufficient or incomprehensible requirements)	Kasoju et al. (2013) and Garousi et al. (2017)
<i>C_{RW}</i> 06	Missing test plan (and thus undocumented test strategy, test coverage criteria, etc.)	Lachmann and Schaefer (2013)
<i>C_{RW}</i> 07	Correlation between domain knowledge and tester skills (when developing test cases based on human expertise, the quality of the test cases depends on the knowledge of the tester)	Garousi et al. (2017)
<i>C_{RW}</i> 08	Lack of domain and system knowledge transfer (e.g., if testers personnel changes frequently)	Kasoju et al. (2013)
<i>C_{RW}</i> 09	Lack of basic testing knowledge	Kasoju et al. (2013)
<i>C_{RW}</i> 10	Need for training for various test activities	Garousi et al. (2017)

Table 1 (continued)

ID	Challenges	References
<i>C_{RW}</i> 11	Lack of traceability between requirements and test cases makes it hard to determine test coverage	Kasoju et al. (2013)
<i>C_{RW}</i> 12	Lack of a meaningful metric regarding traceability	Garousi et al. (2017)
<i>C_{RW}</i> 13	Poor change management (e.g., test artifacts are not updated continuously)	Kasoju et al. (2013) and Sundmark et al. (2011)
<i>C_{RW}</i> 14	Lack of a structured test process and lack of a seamless chain of methods	Grimm (2003) and Kasoju et al. (2013)
<i>C_{RW}</i> 15	Difficulties in defining the exact responsibilities of different test levels	Sundmark et al. (2011)
<i>C_{RW}</i> 16	Lack of dedicated testers or unavailability of personnel for testing	Kasoju et al. (2013)
<i>C_{RW}</i> 17	Distributed development of a system (e.g., involvement of different suppliers in the development and test process)	Grimm (2003) and Pretschner et al. (2007)
<i>C_{RW}</i> 18	Lack of regular face-to-face meetings (to avoid miscommunication)	Kasoju et al. (2013)
<i>C_{RW}</i> 19	Differences of opinion regarding test effort distribution	Sundmark et al. (2011)
<i>C_{RW}</i> 20	No unified tool for entire testing activities	Grimm (2003), Kasoju et al. (2013), Broy (2006), and Sundmark et al. (2011)
<i>C_{RW}</i> 21	Lack of documentation on how tools work	Kasoju et al. (2013)
<i>C_{RW}</i> 22	Need of better tool support	Grimm (2003) and Garousi et al. (2017)
<i>C_{RW}</i> 23	Natural language-based test cases influence comprehensibility	Lachmann and Schaefer (2014)
<i>C_{RW}</i> 24	Manual test cases are often too long or do not contain necessary details	Garousi et al. (2017)
<i>C_{RW}</i> 25	Documentation issues on explorative testing	Garousi et al. (2017)

electric drive, comfort systems, preventive safety systems, or powertrain-related systems. Each interview lasted for around 90 min. With permission of the participants, the interviews were recorded by the main author and transcribed. Overall, we interviewed 17 participants. Table 2 summarizes the characteristics of the interviewees.

Participants assigned themselves to one or more area of responsibility: (C) creating test case specifications (12 interviewees), (D) delegating the creation of test case specifications to suppliers (10 interviewees), (R) reviewing test case specifications (14 interviewees), and (I) implementing test case specifications (7 interviewees). In addition, they assessed their testing expertise themselves, based on the five-stage model of Dreyfus and Dreyfus (1980). They assigned themselves into the category of beginner (Level 1), advanced beginner (Level 2), competent practitioner (Level 3), experienced practitioner (Level 4), or expert (Level 5). On average, the interviewees rated themselves as more experienced practitioners in terms of their testing expertise (*Median* = 4, *N* = 17).

Data analysis. The analysis of the interview data is based on a qualitative content analysis according to Seaman (1999) and Runeson et al. (2012). The coding of the data was performed using the three main coding activities based on Strauss and Corbin (1998): open coding, axial coding, and selective coding. Open coding generates categories using conceptual codes. This was done by highlighting noteworthy statements and assigning a code. The relationships between the concepts and their subcategories were identified during the

Table 2 Overview of the interviewees

Role ¹	Company affiliation (Years)	Testing expertise (Level ²)	Responsibilities ³				Label
			C	D	I	R	
System manager	< 3	3		✓		✓	Int01
Test manager	3 – 5	4	✓	✓	✓		Int02
Test house manager	11 – 25	5		✓			Int03
Test manager	3 – 5	4	✓	✓		✓	Int04
Function developer	3 – 5	2	✓	✓	✓	✓	Int05
System manager	< 3	4	✓			✓	Int06
Function manager	3 – 5	3	✓			✓	Int07
Test manager	11 – 25	4		✓		✓	Int08
Function developer	11 – 25	4		✓		✓	Int09
Project manager (S)	6 – 10	4	✓		✓	✓	Int10
Test manager (S)	< 3	4	✓		✓	✓	Int11
Test manager	< 3	2	✓			✓	Int12
System manager	6 – 10	3	✓	✓			Int13
System manager	< 3	3	✓		✓	✓	Int14
Test manager	11 – 25	3	✓	✓		✓	Int15
Tester (S)	6 – 10	4	✓		✓	✓	Int16
Tester	≥ 26	3		✓	✓	✓	Int17

¹ Suppliers/external employees are marked by (S)

² Expertise levels: (1) beginner, (2) advanced beginner, (3) competent practitioner, (4) experienced practitioner, (5) expert

³ Responsibilities: (C) creating, (D) delegating, (I) implementing, (R) reviewing test case specifications

axial coding. For this purpose, the conceptual codes were supplemented by a classification into cause, consequence, current solution, or desired scenario. Finally, nine main categories were defined in the selective coding phase.

Threats to validity. In the following, we discuss threats to validity according to the four threat aspects provided by Runeson et al. (2012). To increase *construct validity*, we tried to avoid a biased selection of interviewees by considering various systems, roles and testing knowledge for the selection of interviewees as well as having a sufficient number of interviewees (see Table 2). At the beginning of the interview, the study goal was explained to the interviewees in order to avoid misunderstandings. Further, to obtain detailed results and avoid interpretations of the answers to the research question, the interview questions are based on the formulated research question. In addition, to reduce the restraint of the participants, the data was anonymized. The interview guide was reviewed by five persons and subsequently adapted and refined as a result of a pilot interview to improve the *reliability*. The anonymized interviews were literally transcribed by students and reviewed by the main author to minimize influences of the researcher and to remove transcription errors. The conceptual codes were reviewed in pairs by two of the article's authors. Everyone got familiar with the data material and used the developed coding system. The results were then discussed with three external researchers and the coding system was iteratively refined. It might not be possible to generalize the findings of the case study to automotive software engineering in general to other OEMs and suppliers. However, with respect to *external validity* we deliberately selected interviewees from eleven independent internal departments and external suppliers which work for other OEMs, too. Hence, we believe that some of our results are generalizable to a certain extent.

4.2 Second study: descriptive survey

Survey design. The goal of our descriptive survey (Dresch et al. 2015) is to validate the challenges identified in the previous exploratory case study. Our interests concerned how practitioners assessed the frequency of occurrence and the criticality of a challenge. For this purpose, we developed a questionnaire based on the identified challenges from the exploratory case study, which can be found online.³ The questionnaire consists of an introductory section with information about the purpose, the approximate time to answer the questions, and a brief overview of the topic test case specifications. The time indication was based on measured values for completing the questionnaire from a pilot test. After that, some demographic data queries follow in order to characterize the participants in relation to their working context, e.g., company affiliation and testing responsibilities. The main part consists of questions about the identified challenges, which are grouped according to the nine main categories of the taxonomy. In this respect, we derived a total of 92 challenges from the interviews (C1 – C92). For each challenge, the participant first had to assess whether it “does occur” or “does not occur” or whether he or she does not want to make any statement (“no comment”). A nominal scale was used to measure these answers. If a challenge has been marked as “does occur,” the participant was asked to provide additional information about the frequency of occurrence and criticality of the challenge. Otherwise, the participant did not have to give an assessment for this challenge. This limitation was

³Questionnaire online: <http://dx.doi.org/10.18725/OPARU-33548>

		Occurrence			Frequency					Criticality				
		no comment	does not occur	does occur	very rare	rare	sometimes	often	very often	not critical	rather not critical	partly critical	rather critical	very critical
1 Input Artifacts (Availability)	C01 Requirement Specification does not exist	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
	C02 Requirement Specification does not exist in time	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
	C03 Test plan does not exist	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
	C04 Information and documents are distributed (e.g. on different systems, storage formats, etc.)	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
	C05 No access to relevant documents (Means documents, which are relevant for creating test case specifications e.g. requirement specifications, test plans, additional documents, etc.)	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐

Fig. 5 Example of a question group from the developed questionnaire

chosen to avoid vague statements about the frequency or criticality of a challenge. Only if a participant was aware of the challenge and it actually occurs during his or her work on test case specifications it is expected that he or she can make a reliable statement about the assessment. For the assessment of frequency of occurrence and criticality, we used five-point Likert scales with an ordinal scale level (Blaikie 2003). An example of a question group with different challenges and associated scales is shown in Fig. 5.

At the end of the questionnaire, respondents had the opportunity to provide further challenges and comments as a free text answer. We implemented the questionnaire as an online survey using the survey tool LimeSurvey.⁴ This was done to minimize the effort for data collection, to ensure anonymous recording of responses and to automate the distribution of the questionnaire via e-mail.

Data collection. In general, we selected practitioners who deal with test case specifications to participate in the survey. On the one hand, survey participants were selected from the same German OEM as in the first study. Therefore, we used a company directory that lists internal employees responsible for creating or commissioning test case specifications. Furthermore, internal employees of three test teams were selected who are responsible for the implementation and execution of test cases for various systems. On the other hand, external employees of three suppliers were selected who are commissioned to create, document, implement, and execute test cases. One of the three suppliers is the same as in the first study, but the employees involved in the survey are different from those in the first study. Overall, the participants invited for the survey differ from the interview participants. This approach was chosen to ensure that the identified challenges were assessed impartially by the survey participants. Therefore, interviewees were not involved in the survey as they already knew the challenges.

Overall, 36 participants completed the questionnaire within a period of two weeks. With regard to 220 invited participants, this represents a response rate of 16%. The 36 participants consist of 26 internal employees from 20 different departments and 10 external employees from three different suppliers. Table 3 shows an overview of the characteristics of the survey participants. The areas of responsibility are divided among the participants as follows: 18 participants creating test case specifications (C), 17 participants delegating the creation of test case specifications to suppliers (D), 19 participants reviewing test case specifications (R), and 26 participants implementing test case specifications (I). On average, both groups rated themselves as competent practitioners in terms of their testing expertise (*Median* = 3, *N* = 36). The company affiliation refers for

⁴Survey Tool LimeSurvey: <https://www.limesurvey.org/>

Table 3 Overview of the survey participants

Groups of participants	Sample size (<i>N</i>)	Company affiliation (years)	Testing expertise (level ¹)	Responsibilities ²			
				C	D	I	R
Internal employees	26	12.5 years	3	13	17	14	13
External employees	10	3.0 years	3	5	0	12	6
All participants	36	5.6 years	3	18	17	26	19

¹Expertise levels: (1) beginner, (2) advanced beginner, (3) competent practitioner, (4) experienced practitioner, (5) expert

²Responsibilities: (C) creating, (D) delegating, (I) implementing, (R) reviewing test case specifications

internal employees to the time in the company as an employee and for external employees to the time they are already working for this OEM (cf. interview guide online). Thus, internal employees have an average company affiliation of 12.5 years and external employees of 3.0 years.

Data analysis. We analyzed the answers of 36 questionnaires. Incomplete questionnaires, 44 in number, were excluded from the analysis. For data analysis, we used Fisher's exact test (Fisher 1992) with a level of significance $\alpha \leq 0.05$. This is a non-parametric statistical test, which is applicable for nominal scales and is recommended if the sample size is rather small. Both applies to our case, since we use a nominal scale for the occurrence and the group of external employees is only $N = 10$. The chi-squared (X^2) test would often not provide reliable results, as the required expectation value in the cells of the cross table tends to be less than 5. In our case, the cross table is structured as follows. The two dichotomous variables company affiliation (possible values: "external" or "internal") and the assessed occurrence of the respective challenge (possible values: "does occur" or "does not occur") were considered. The 2×2 cross table contains the different groups in the rows and the answers for the respective questions in the columns. If Fisher's exact test indicates a significant difference, the ϕ coefficient was also calculated (Cohen 1988). We use this symmetrical measure to make a statement about the effect size (small effect size: $\phi = 0.1$, medium effect size: $\phi = 0.3$, large effect size: $\phi = 0.5$).

Threats to validity. In the following, we discuss the four different threats to validity aspects similar to those presented in Section 4.1.

To increase *construct validity* we tried to avoid a biased selection of survey participants by considering various systems, roles and testing knowledge for the selection of interviewees as well as having a representative number of participant. Furthermore, we conducted pilot tests to ensure the comprehensibility of the questions. At the beginning of the survey, the study goal was explained to the survey participants in an introductory part in order to avoid misunderstandings. Further, to obtain precise results, the questions about the challenges were chosen in such a way that they cover the identified challenges from the exploratory case study. In addition, the questionnaire survey was conducted anonymously in order to avoid restraint among the survey participants in answering the questions. However, we cannot completely exclude a selection bias, since the internal surveyed employees all came from the same OEM. Nevertheless, we believe that due to the measures taken construct validity threats could be limited.

In order to improve *internal validity* we tested the developed questionnaire. By using a two-round pilot study the instrumentation quality was enhanced. In order to avoid a learning

effect and maturation of the test persons, the test persons were only allowed to complete the questionnaire once. This also takes into account that participants from the interview study were not invited to the questionnaire survey in order to avoid influences from the interviews and to assess the identified challenges independently.

The questionnaire was reviewed by six persons within the pilot study. Two of them were external researchers, one internal researcher and three internal practitioners. As a result the questionnaire was subsequently adapted and refined to improve the *reliability*. In order to calculate the significance of two different groups on the basis of a hypothesis and thus ensure the reliability of such an analysis, the Fisher's exact test with a significance of 5% was used.

To increase *external validity*, it is important to note that participants from 20 independent departments and 3 external suppliers who also work for other automotive companies participated in the survey. These participants represent a wide range of different responsibilities and levels of knowledge (cf. Table 3). Considering this, we believe that despite the low response rate of 16%, the threat of non-response bias is marginal. However, there are various discussions about which response rate is sufficient for online surveys. It is known from other domains that even low response rates can be sufficient. For example, Nulty (2008) considers a response rate of 12% (10%) to be sufficient under liberal conditions, which corresponds to only 23 (24) respondents for a sample size of 200 (250).

5 Identified challenges

In this section, we present the results of our exploratory case study and the qualitative data analysis (**RQ1** and **RQ2**). We present one of the nine main categories per subsection with the associated challenge types (ToCs), which are described by means of problems/aspects (printed in bold) stated in the interviews and related to the results of the selective coding. In addition, we illustrate the identified problems and their causes by statements from the interviewees. In some cases, interviewees described consequences arising from challenging aspects or solutions that have already been put into practice. Table 4 gives an overview about the categories (M1 – M9) and the related types of challenges, such as ToC-1.1. In addition, the numbers of the corresponding concrete challenges (C1 – C92), as queried in the descriptive survey (cf. Section 6), are assigned to the respective main categories and types of challenges.

5.1 Availability problems with input artifacts (M1)

Category definition. This category groups challenges related to input artifacts that do not exist, are not available in time or in the required maturity. Furthermore, this includes distributed information or documents (e.g., stored in different systems or databases), or relevant input artifacts for which access is missing or not provided.

5.1.1 Non-existing input artifacts (ToC-1.1)

Interviewees reported that for requirement-based testing, the **specifications are often not available in time**: *“In my opinion, it is also one of the most important challenges that the whole left side of the V-model has to be in a corresponding quality and it is usually NOT. It is also one of the biggest problems in my opinion that I have to test something with non-existent requirements.”* (Int10 | Project Manager). In addition, documents that

Table 4 Taxonomy of test case specification challenges

ID	Main categories (M)/Types of challenges (ToC)	Challenge numbers
M1	Availability problems with input artifacts	C1 – C5
ToC-1.1	Non-existing input artifacts	C1 – C3
ToC-1.2	Distributed input artifacts	C4
ToC-1.3	No access to or provision of input artifacts	C5
M2	Content-related problems with input artifacts	C6 – C17
ToC-2.1	Content-related problems with requirement specifications	C6 – C11
ToC-2.2	Content-related problems with test plan	C12 – C15
ToC-2.3	Conformity of test case specification template with user needs	C16
ToC-2.4	Content-related problems with other input artifacts	C17
M3	Knowledge-related problems	C18 – C24
ToC-3.1	Lack of knowledge about the SUT	C18
ToC-3.2	Lack of knowledge about test platforms	C19, C20
ToC-3.3	Lack of knowledge about testing policies	C21 – C24
M4	Test case description related problems	C25 – C55
ToC-4.1	Language-based problems in test cases	C25, C26, C33
ToC-4.2	Phrasing-based problems in test cases	C27 – C32
ToC-4.3	Quality-based problems in test cases	C34 – C55
M5	Test case specification content-related problems	C56 – C58
ToC-5.1	Handling of variants and model series	C56, C57
ToC-5.2	Focusing on specific test platforms	C58
M6	Process-related problems	C59 – C70
ToC-6.1	Lack of standards or guidelines	C59 – C63
ToC-6.2	Change management related problems	C64
ToC-6.3	Organizational decisions	C65 – C70
M7	Communication-related problems	C71 – C77
M8	Quality assurance related problems	C78 – C86
ToC-8.1	Inadequate definition of the term quality	C78, C82,
ToC-8.2	Lack of useful metrics	C79, C80, C83
ToC-8.3	Lack of an (established) review process	C81, C84 – C86
M9	Tool-related problems	C87 – C92
ToC-9.1	Usability and function-related problems	C88, C89, C91
ToC-9.2	Heterogeneous tool chains	C87, C90
ToC-9.3	Deficiencies with the support	C92

The entries in bold are the main categories and group the types of challenges below. To visually distinguish this grouping, the main categories are printed in bold

supplement the requirements may not be up to date, for example the definition of signals in the vehicle network.

Therefore, test designers are forced to create test cases based on previous and out-of-date requirements, or test cases have to be written later, which can lead to a delay of the project. To handle missing signal names, the later signal names are replaced by temporary placeholders, but this leads to a considerable post-processing effort: “So right now, we’re

using the signals we actually need as dummies. And these signals can then only be tested once we have the real signals.” (Int05 | Function Developer).

“So, we also have to completely rewrite the test and system requirements specifications and supplement the signals.” (Int02 | Test Manager).

The **lack of test plans** has also been mentioned as a significant challenge: “The test cases were written, when there was no test plan.” (Int07 | Function Manager) The study showed that this phenomenon is due to a lack of resources (e.g., time), or lack of knowledge about the relevance of a test plan: “This is some standard document, where we don’t put work in. Personally, I think it’s silly. I don’t even know what’s actually written in a test plan.” (Int01 | System Manager).

“We don’t have something like that ... According to ISO 26262 we’ve a classification of QM. Thus we work a bit more free and leave a few aspects away, which would actually be very useful in hindsight, but due to a lack of time are not done.” (Int13 | System Manager)

Missing test plans lead to inefficient, highly redundant testing across multiple test platforms, insufficient utilization of test platforms, increasing test effort and costs: “Of course, the result is that you may have redundant tests. I may have a test case for a HiL [Hardware in the Loop] test platform and somehow later in a slightly different form for a vehicle test platform, and so on. So, that’s why there may exist redundancies.” (Int13 | System Manager).

This challenge is treated with a rudimentary standard test plan, which can still lead to an increased or insufficient testing effort. Some interviewees mentioned that they want to commission test plans in the future. This requires a deep knowledge of the supplier about the available test platforms, which is not common.

Seven external and internal interviewees, including Test House Manager, Function and System Manager, or Test Manager, remarked that a test plan would be helpful for creating the test case specification and improving their test process: “If a system manager thinks about this from the beginning - where do I test which content - and then of course takes our test platform into account, we would have gained a lot.” (Int08 | Test Manager).

5.1.2 Distributed input artifacts (ToC-1.2)

Requirements or additional **information necessary to understand the system specifications are stored in a distributed manner**: “The basis is always the system specification ... [but] the contained information is rather general. Hence, we’ve to extract missing information [from other artifacts], e.g., signal documentation.” (Int 11 | Test Manager) The distribution of information is often reflected in the use of different tools: “I also received Excel lists from colleagues as input for creating the test case specification ... [and] several DOORS documents from other colleagues.” (Int03 | Test House Manager). Test designers are forced to collect this information: “I’ve to find all the requirements. Department A writes its requirements in SMARAGD, colleagues from Department B write their requirements in POLARION and our colleagues C write them in CyberArk.” (Int15 | Test Manager).

This results in the demand: “I’d like to see requirements all written in the same tool at our company.” (Int15 | Test Manager).

Attempts have been made to enrich the specifications with the necessary additional information: “All possible figures ... which were partly included in the specification, but we also have many overview figures as PDF files.” (Int01 | System Manager). Interviewees often expressed that it would be helpful to reduce the number of possible sources or to standardize them. But due to the complexity and size of the systems to be developed, it will be difficult to combine everything in a single document.

5.1.3 No access to or provision of input artifacts (ToC-1.3)

Interviewees mentioned that **specifications contain references to other specifications** or to parts within the specification to which they have no access: “Well, we already have the classic additional applicable documents. But we usually only get the system part from the requirement specification and not the complete requirement specification. If there are links in it, then we have trouble dealing with them.” (Int11 | Test Manager).

This is often a problem in cooperation with suppliers, but was also observed during the interviews between departments of the same company. The resulting information gaps lead to a poorer system understanding and to incorrect or incomplete test cases.

Interviewees mentioned that if the lack of access rights exists for technical reasons, data is provided manually: “Sometimes I’ve to extract files explicitly and provide these files to them [the supplier], because they can’t access them.” (Int01 | System Manager). However, this can lead to poorer traceability between test cases and test basis documents.

Interviewees expressed uncertainties about what information they were allowed to provide to a supplier: “This is a difficult issue ... when I explain something to a supplier, I always have to be careful what information I share with him and what I don’t.” (Int14 | System Manager). If an insufficient amount of information is provided, suppliers are encouraged to clarify ambiguities through communication (e.g., time-consuming workshops).

5.2 Content-related problems with input artifacts (M2)

Category definition. This category summarizes challenges related to errors in requirement specifications and test plans. Moreover, challenges with the used test case specification template and other input artifacts are shown.

5.2.1 Content-related problems with requirement specifications (ToC-2.1)

The interviewees highlighted different types of **errors in requirement specifications** which frequently occur: missing, obsolete, changing, incomplete, or conflicting requirements. These errors make it difficult writing a high-quality test case specification. The risk of outdated requirements is high because typically functions from previous model series are copied and changed.

With regard to the **testability rating of requirements**, interviewees cited the incorrect or missing classification of requirements as a challenge for creating a test case specification: “For example, filling in the attribute for the testability. I believe everybody does it a little bit by gut feeling ... I would say that it is not yet specified up to 60% of the cases.” (Int11 | Test Manager).

If the testability of a requirement has not been evaluated, then it could not be tested or is incorrectly tested. Based on our observations, it can lead to an increased testing effort, because the testability had to be determined later for the affected requirements.

In order to clarify **incomplete or conflicting requirements**, test designers must communicate with the responsible persons: “I’m then dependent on the response from a system or component manager.” (Int03 | Test House Manager). In the worst case, uncertainties in the specification are identified for the first time during test execution.

Furthermore, interviewees mentioned the **increasing system complexity and size** and its impact on testing such systems: “Often, system specifications have grown greatly in the past. This makes the creation of a test specification difficult.” (Int06 | System Manager).

This is due to a large scale of functions and the corresponding number of requirements (e.g., over 6.000 requirements for only one system) as well as the constraint that each requirement has to be tested and linked with the corresponding test case: “Then you have to link a thousand times.” (Int02 | Test Manager). Interviewees cited that increasing complexity makes it more difficult to understand the requirements without additional information.

5.2.2 Content-related problems with test plans (ToC-2.2)

Interviewees named the **creation of a high-quality test plan** as challenging: “It is always a challenge to create a test plan.” (Int02 | Test Manager). Reasons for this are a lack of description of the procedure for creating a test plan or the difficulty of decomposing the system under test into smaller testable units. This requires a deeper understanding of the requirements, as there are requirements that can result in multiple test cases for different test platforms.

Consequences associated with a low-quality test plan are similar to those of a missing test plan as mentioned in Section 5.1.1: increased test effort, inefficient or incorrect usage of test platforms. In addition, it is often observed that the test plan is not taken into account: “The influence of the test plan is very, very limited. But it is practically not used.” (Int09 | Function Developer). The decomposition of the function, the assignment of variants and the assignment to test levels do not take place until the test case is created.

5.2.3 Conformity of test case specification template with user needs (ToC-2.3)

Another challenge is that an all-encompassing **test case specification template does not meet all user needs** due to the heterogeneous nature of software depending on the different domains in automotive engineering. The influence of the domains on the development implies influence on the testing. For example, interviewees complained about the large number of attributes to be filled when using an all-encompassing template: “Because it is very bloated ... the template also contains more than I actually need.” (Int13 | System Manager).

“There are certain columns that make sense and others I fill because I have to.” (Int14 | System Manager).

Furthermore, we observed different needs of how test cases and in particular test case metadata have to be specified. This includes **support of multilingual test cases**. For example, “a test case ... has two language tracks. That means, ... a German and an English version directly next to each other.” (Int08 | Test Manager). This is particularly important for test cases that are used worldwide, such as acceptance test cases that are executed at the end of production in various production plants.

Another point is the **possibility of reusing different parts of a test case**, such as preconditions or configurations. “In our case the reactions [expected results] are often very similar, but ... are executed differently depending on different environmental parameters. ... In the end, we’ve a very high copy & paste effort when it comes to the reactions, which are always the same.” (Int06 | System Manager). The global definition of these reactions and their linking with the corresponding test steps would be an example for the reuse of parts of a test case. Preconditions are already reused in this way.

In addition, interviewees mentioned the large number of test cases whose sequence is fundamentally identical but only differs in individual variables as challenging. “What I miss ... is that I can work with parameters. ... It’s always ... hard coded. For example, I’ve a coding parameter and I know I’ve to test it with values from 5 to 50. Then I’ve ... 49 test steps that I actually have to write down.” (Int10 | Project Manager). This could be avoided

by **parameterizing test cases**, which would reduce the number of test cases to be written: “And there is the consideration ... that I add a parameter module in which I define the value range and step size etc. ... and then it [the concrete test cases] will be generated automatically afterwards.” (Int10 | Project Manager).

We observed that in most cases the template was modified to support system or project-specific needs, sometimes without considering compatibility to the downstream tool chain. For example, attributes are added or their value range is supplemented in the context of variant and model series management. Interviewees emphasized the necessity of documenting the testing purpose and the origin of the test case (e.g., requirement, lessons learned, experience-based).

Based on these statements of the interviewees, we summarized the following significant challenges that a test case specification template has to deal with: parameterization of test cases, managing model series, variants, test data, reusable parts of a test case (e.g., preconditions, configurations), and supporting multi-language (e.g., German and English) test cases. In addition, it should be possible to customize the template to system or project-specific needs while still keeping compatibility to the testing tool chain.

5.2.4 Content-related problems with other input artifacts (ToC-2.4)

In addition to content-related problems with requirement specifications or test plans, content-related problems with other input artifacts also pose challenges to the error-free creation of test case specifications. This applies, for example, to the **reuse of test cases from previous versions of a test case specification**: “This [errors in test cases due to reuse] occurs when previous test case specifications are copied and modified. The old test cases contain the wrong signal names, which are no longer valid for the current system.” (Int05 | Function Developer).

Interviewees also mentioned the continuous revision of the communication matrix (C-Matrix) as challenging. As a result, test cases often contain incorrect or obsolete signal names.

To avoid outdated signal names in test cases, parameters are used, for example: “We do not write signal names in the specifications. We link to a parameter list.” (Int06 | System Manager). Signal names and other variables are managed in a separate parameter module, which means that the test case specification does not have to be explicitly adapted in the event of changes.

5.3 Knowledge-related problems (M3)

Category definition. This category points out challenges related to a lack of knowledge in terms of system understanding, test platform functionalities, guidelines for testing, and deficiencies in consulting, training, and documentation.

5.3.1 Lack of knowledge about the system under test (ToC-3.1)

Interviewees highlighted the **inadequate understanding of the system under test (SUT)** as a major source of errors in test case specifications, which can often be observed with new and inexperienced colleagues or suppliers: “It’s common to get the impression from external parties that system understanding is missing.” (Int14 | System Manager). This concerns

different roles. On the one hand, if the test designer has no knowledge about the SUT, it can lead to errors in test cases: “Then test cases were developed, about which we said: ‘Ok, this would not work at all in the real vehicle’.” (Int04 | Test Manager). On the other hand, testers can implement a test case incorrectly because they have misinterpreted it: “The test cases were misinterpreted ... and the implementation was simply wrong.” (Int01 | System Manager). Moreover, a reviewer of a test case specification without a deeper system understanding can tend to neglect semantic checks: “It’s more likely to only check formal criteria ... but do not really look at the content anymore.” (Int03 | Test House Manager).

Therefore, interviewees with different responsibilities in the test process mentioned that test designers, testers and reviewers should have a comprehensive understanding of the system to better design and understand test cases: “I think that the reviewer should have a deep understanding of the system.” (Int03 | Test House Manager).

As a possible solution to fill knowledge gaps and to establish a common understanding about the SUT, the organization of workshops with all relevant participants was mentioned.

5.3.2 Lack of knowledge about test platforms (ToC-3.2)

Another challenge arises from the **lack of knowledge about the functionalities of test platforms**, which have an impact on creating test case specifications: “I don’t know how to test [using the test platform], so I can’t write a meaningful test case specification at all.” (Int03 | Test House Manager).

Interviewees also cited the **lack of overviews of test platforms** as challenge: “I have no idea what kind of test levels and test platforms we have.” (Int15 | Test Manager).

These challenges can lead to a suboptimal assignment of test cases to test platforms and late error detection because test cases were only tested in the vehicle in a late test phase although earlier test phases would have been more suitable.

To handle these challenges, testers or testing departments try to explain functionalities of a test platform by means of example test cases: “We wrote test cases, showed him [the test designer] this, and he then said, ‘Yes, I understand how you work, how you need it!’” (Int03 | Test House Manager).

5.3.3 Lack of knowledge about testing policies (ToC-3.3)

Interviewees mentioned that (internal) **testing policies do not exist or the existing ones are rarely used**. Reasons for this are unknown sources of information, missing of a reference process, guidelines, or examples. Finding relevant information is classified as difficult by the interviewees: “Yes, I basically find it hard to find things ... Usually you either have to know where to find something or you know the people who know it.” (Int14 | System Manager).

Moreover, a majority of the interviewees stated that due to lack of time, they do not have the opportunity to read huge manuals.

To meet this challenge, the intuition and profound knowledge of a test expert was highlighted as a necessary success factor for efficient testing. In particular, test designers want one-sided instructions, continuous and short refresher courses, newsletters or an expert hotline. A cross-departmental reference process and control authorities that verifies compliance with established testing guidelines have been described as helpful and even necessary: “I do not know any guidelines. Unfortunately not. It would be good if there were such guidelines, because then we could benefit from them.” (Int08 | Test Manager).

5.4 Test case description related problems (M4)

Category definition. This category addresses challenges related to the phrasing of test cases and influences by the language used. These challenges relate in particular to the description of the test

steps with the individual actions and expected results. It also covers various quality characteristics of a test case such as completeness, reusability, comprehensibility, consistency, and unambiguity.

5.4.1 Language-based problems in test cases (ToC-4.1)

Translation and spelling errors have been mentioned by interviewees as common examples of language-based problems.

This challenge can lead to false test cases, whose meaning differs from the original test case: “We still write a lot in German and the Indians use Google Translate, which works quite well for standard text but not for technical text. This also means that, of course, incorrect translations are made and then, accordingly, wrong test cases are created.” (Int10 | Project Manager). A wrong spelling or even a missing letter can lead to a fundamentally different meaning. For example “driving at a speed of 10 m/h” would be pretty slow and hopefully the tester would notice.

Interviewees stated that using more formal approaches (e.g., describing a sequence of signal changes on the network) reduces the risk of translation errors: “But thank God, we are often on the signal level. There is works.” (Int05 | Function Developer). Signal level means that the description of the test cases is based on simple assignment operators.

5.4.2 Phrasing-based problems in test cases (ToC-4.2)

Interviewees mentioned the **phrasing of test cases as one of the most important challenges**. It depends on the test designer how a test case is specified and documented (e.g., full sentences vs. short bullets and abbreviations), which is a challenge when several different authors write a test case specification: “You can see the writing style.” (Int01 | System Manager).

“The biggest challenge is to find a common style.” (Int04 | Test Manager).

These **variations of phrasing** can lead to ambiguities and misunderstandings during test implementation: “Misunderstandings, definitely yes. I think, it is not a problem in the test case creation, but actually in the test execution.” (Int10 | Project Manager). Besides, **excessive use of abbreviations** can impair the comprehensibility and readability of test cases, if glossaries are not used: “Abbreviations can be misunderstood.” (Int11 | Test Manager).

Writing test cases in a uniform way is preferred by test designers and testers, but this requires a lot of discipline when creating test cases. For this reason, test designers often work with copy and paste or unify test cases retrospectively, which means increased workload: “In the end, I revised the test cases to unify them.” (Int13 | System Manager).

“I have defined a vocabulary within my test case specification. And that is what I use. Because many of the steps are actually the same, I copy a lot. This gives me a vocabulary for my own system or for my test case specification.” (Int16 | Tester).

In addition, the **specification of natural language test cases** is also considered a challenge due to insufficient and ambiguous test case descriptions: “I find a test case bad, if too much prose is in it and you first have to start to disassemble it. ... I find it difficult to get an overview.” (Int06 | System Manager).

“If I have such a long description, I would need to actually underline the core statements during reading.” (Int11 | Test Manager).

Furthermore, **describing test cases in a very abstract manner** was mentioned as a challenge. This case occurs when test cases have been developed in such a way that they can be used generically for several test platforms: “We also have some very nasty constructs, for example ONE test case for MiL, SiL, C-HiL AND Vehicle. We have to fix that. ... That’s rather BAD, because we have time dependencies. MiL, SiL is very much at the beginning [of the development], C-HiL tends to be in the middle and vehicle at the end. That means that if we have to change something for MiL, SiL, we would have to change it for the other platforms as well, but they are not yet so far.” (Int05 | Function Developer). In addition, excessive abstraction in test cases is a reason for interpretation in the test case implementation: “I think the generic test cases are phrased too vaguely, so that I have too much freedom afterwards when implementing them and then something completely different comes out anyway. And for me, a test specification is also essential to ensure that the test is done correctly. And if I have some freedom afterwards, I might even test without any test case specification.” (Int03 | Test House Manager).

Therefore, it was stated that the **phrasing used in test cases have to be oriented towards the target test platform**, e.g., test cases for a HiL test platform (a lot of signals and abbreviations) differ significantly from test cases in the vehicle (mainly prose): “They differ greatly because the test cases are specific with respect to the functional characteristics of a test platform ... and therefore the test cases are completely different.” (Int14 | System Manager).

“It is the task of the test designer ... to formulate test cases with respect to the target [special test platform].” (Int03 | Test House Manager). This can be problematic if a test designer assigns a test case to different test platforms, but has formulated this test case according to the needs of a specific test platform. For example, if a HiL tester receives a test case which was actually developed for the entire vehicle test, then signal names and values which he would have to test are often not specified. This means that important information may then be missing. In the worst case, the test case describes actions that cannot be performed on a HiL system at all, such as the activation of controls that are not part of the HiL system and therefore need to be simulated.

5.4.3 Quality-based problems in test cases (ToC-4.3)

The selection and application of a particular writing style or phrasing influences different characteristics of a test case. As already mentioned, an excessive usage of prose influences ambiguity and readability. Interviewees named further challenges associated to the **completeness of a test case** containing incomplete descriptions of test case purpose, missing preconditions, or unspecified test case metadata.

As a consequence, this leads to an increased communication effort, because testers have to ask for missing information. It is necessary to determine the affinity of a test case for corresponding model series and to specify the required test platform: “The test platform attribute has to be set in any case, which is very important for us. The model series are also important.” (Int05 | Function Developer). Otherwise, testers are not able to identify relevant test cases for their test platform or test cases are assigned to the wrong test platform.

The possibility to react straightforward and without much effort to changes was also named as challenge. A typical example during the development is the **change of signal names**, which are used in several test cases: “Changes in the signal names ... led to a lot of manual work.” (Int12 | Test Manager). This includes the tedious manual searching and replacement of the affected parts.

An approach to counter this challenge is the usage of placeholders or the extraction of reusable sequences into a base scenario. If a change is necessary, test cases do not have to be adjusted individually, because the change takes place at a central point: “If it’s going well, it’s just one base scenario where you have to adjust the signal.” (Int02 | Test Manager).

The interviewees also named missing strategies and mechanisms for **reusing test cases** across different model series as challenging. Therefore, test cases were often redesigned or have to be manually copied and adjusted for the new model series. In particular, the **reusability of experience-based test cases** was highlighted, because such test cases are often lost in requirement-based testing approaches because of insufficient documentation: “In terms of reusability ... I’ve made an experience [finding an error independent of requirement-based test cases] ... Such that I document it [the experience-based test case] and such a test case is not lost.” (Int09 | Function Developer).

If this information is missing and test cases are handed over or reused, this often leads to requesting: “I always ask for: ‘Was this test case created based on a requirement specification or a fault?’” (Int15 | Test Manager).

To treat this challenge, faults were documented in the requirement specification afterwards or the necessary information was documented separately (e.g., Excel), which leads to additional effort.

5.5 Test case specification content-related problems (M5)

Category definition. This category includes challenges related to the scope of content of a test case specification. The management of variants, model series and test platforms is an integral part of test case specifications and is classified here.

5.5.1 Handling of variants and model series (ToC-5.1)

Interviewees remarked the **manageability of test case specifications** including multiple model series and the enormous amount of variants as a challenge. “We now have a great variance in it. We have three variants and each variant has the corresponding five functions. Of course, the variants blow it up.” (Int12 | Test Manager).

As observed, various project-specific methods exist to handle the variety of variants and different series in one or several test case specifications but nothing of this is standardized.

The interviewees emphasized that variant handling had to be covered by the template: “That variant diversity is automatically supported in the template would be great.” (Int01 | System Manager).

5.5.2 Focusing on specific test platforms (ToC-5.2)

Another challenge is the **distribution of test cases** and their storage in many different tools. It is not uncommon that a test case specification only contains test cases for a particular test platform. For example, if only HiL test cases are contained, then “... we are currently using the test specifications primarily in the HiL test environment.” (Int03 | Test House Manager).

Equivalent to a missing test plan, the distribution of test cases encourages the redundant execution of test cases on different test platforms. The distribution of test cases makes it difficult to check the completeness of the test scope, since individual sources may not be considered. This problem arises in particular in the context of commissioning the creation of test cases: “... it is not the case that one [...] company A handles everything. Instead, there is company A, which only does that and company B only does that. [...] This means that

we have commissioned HiL test cases, ... vehicle test cases and then completely forgot that there must also be board test cases with network management.” (Int10 | Project Manager).

Furthermore, changes to the requirements have an impact on various test cases (e.g., HiL and entire vehicle tests). “The topic of distributed systems, for example ensuring a traceable processing of changed test cases, is also very, very difficult.” (Int 09 | Function Developer). As a result, not all test cases may be adapted.

5.6 Process-related problems (M6)

Category definition. This category groups challenges in terms of different understanding between various participating parties, ignorance of requirement changes, the lack of standards or guidelines, and unclear responsibilities.

5.6.1 Lack of standards or guidelines (ToC-6.1)

The actual **creation and further processing of test case specifications differs depending on the department**: “We don’t have a company-wide aligned strategy for the creation of test case specifications.” (Int08 | Test Manager).

“The problem is that each department tests differently, each department has different requirements and necessities and this leads to the fact that you will probably never be able to have a consistent testing process.” (Int17 | Tester).

Interviewees reported that project-specific guidelines are developed for the creation of test case specifications but they are rarely used across departments. Furthermore, the mandatory introduction of a test manager for each project would help to establish a consistent process.

5.6.2 Change management related problems (ToC-6.2)

In some cases, a **poor change management** was observed which leads to failing test cases because test cases no longer fit to the system under test: “... the vehicle behaves quite differently or the instrument cluster is different. Then the test case does not fit anymore because everything [e.g., the control unit integrated in the vehicle prototype, the functionality and thus also the actions and expected results of the test case] has changed in the meantime.” (Int17 | Tester). As a consequence, it is necessary to adapt the test case afterwards and to execute it again, which means a considerable effort.

In this respect, mechanisms for the maintenance of test case specifications are unsuitable or unknown: “Suppose I have a change request, which means I have some kind of change of a requirement. Ideally, I should directly adjust the test case. This is usually not done due to time and no corresponding tooling.” (Int10 | Project Manager).

5.6.3 Organizational decisions (ToC-6.3)

Company-related decisions can affect the test process. Several interviewees stated that **test-related tasks are increasingly being outsourced to suppliers**: “And now we have had a very massive change that he [system manager] just writes system specification and the test house writes a test specification separately.” (Int03 | Test House Manager).

With respect to increasing outsourcing, numerous consequences were outlined by the interviewees. For example, an increasing loss of knowledge about the test cases also leads to another communication level between system manager and test house, because no longer

the failed test cases are of interest to the system manager, but the affected requirements. This requires a consistent traceability, which in some cases has been described as insufficient by the interviewees.

Another aspect regarding organizational challenges refers to the **definition of roles and responsibilities** in the test process (e.g., system managers have a very broad range of tasks), the lack of staffing for important positions (e.g., vacant position of the test manager) or the non-regulation of responsibilities. Differences also result from system- or project-specific aspects: “That is quite different. There are systems that are relatively small. Such systems have someone in the personnel who is system and component responsible at the same time. There are systems that have one system manager and several component managers. There are also systems that have several subsystem managers.” (Int17 | Tester). Even due to department-specific differences, responsibilities are not always clear.

One solution mentioned by the interviewees to keep track of the different responsibilities is the system list: “A very helpful medium is the so-called system matrix or system list ... because it also addresses subsystems and ... sub-functions and ... who is the contact person” (Int17 | Tester).

5.7 Communication-related problems (M7)

Category definition. This category includes challenges focusing on communication problems caused by cultural background and different expectations of test designers and testers with respect to the assumed knowledge about the SUT and testing methodology.

Interviewees addressed that the **cultural background**, depending on national origin, influences how technical aspects are described and understood as well as the inhibition threshold for queries: Yes, that is a huge problem. For some reason and I don’t know why, they struggle to ask and start instead to interpret.” (Int10 | Project Manager).

“This is simply another cultural understanding. Phrases, how to describe things, are also different.” (Int03 | Test House Manager). These observations refer to the cooperation with Indian colleagues in a branch abroad.

Furthermore, **different expectations** occurred mainly between test designers and testers, based on implicit knowledge about the system under test. Therefore, some information is not explicitly documented by test designers in the test case, but is presupposed by testers: “For example, one of the actions in a test case is ‘start engine’. If the vehicle has an automatic gearbox, I have to step on the break. If the test case does not contain ‘hold the break’, then there are always testers who say: There is no ‘hold the break’ in the test case. This is an error. So, it depends on the expectation of the tester. Actually, he should know, that he has to step on the break, but not everyone knows it.” (Int08 | Test Manager).

Workshops can help to counteract these challenges and example test cases are already used to show suppliers a proper detail level of a test case. In addition, interviewees reported that it would be beneficial to communication if the participating parties were close to each other (physical proximity).

5.8 Quality assurance related problems (M8)

Category definition. This category presents challenges related to the quality assessment of test case specifications. This includes the inadequate definition of the term quality in the context of test case specifications, the lack of useful metrics, and the missing of an established review process.

5.8.1 Inadequate definition of the term quality (ToC-8.1)

We observed that **interviewees often had no idea of what quality means for their test case specification** or what could be necessary quality characteristics: “Good Question. I guess that no one can give an answer.” (Int02 | Test Manager).

“That’s difficult.” (Int05 | Function Developer).

When quality characteristics were named, then comprehensibility (named 12 times), unambiguity (12), and completeness (8) were indicated as important but without any established or known process to check or to ensure their compliance. Characteristics such as uniformity (4), atomicity, (4) and suitability for the respective test platform (3) were much less stated.

5.8.2 Lack of useful metrics (ToC-8.2)

Only requirements coverage was stated as well-known metric to evaluate the quality of a test case specification, but other metrics for quality measurement were not mentioned: “As a metric, we currently only have the traces to the system specification. Well, that’s actually the only possibility known for me, which we have right now.” (Int04 | Test Manager).

“I do not have any idea, because I think this is actually too specific to each test case specification.” (Int16 | Tester). Considering that the majority of the test case specifications are based on the same template, it is not absolutely necessary to have individual metrics. Without appropriate metrics, it is difficult to make a reliable statement about the test case specification quality.

We collected some ideas of the interviewees for possible metrics: number of mandatory attributes that are not filled, number of test steps to evaluate the size of a test case, number of reused phrases, number of prioritized test cases, number of used words in a test case, or the consistent usage of terms according to a glossary.

5.8.3 Lack of an (established) review process (ToC-8.3)

Interviewees reported that there are **no established reviews** explicitly for the test case specification, either due to lack of time or personnel: “By whom? ... Is that somehow reviewed in connection with the specification? No.” (Int06 | System Manager).

“Nobody cares for the system except me, so no one else looks at the tests except me.” (Int14 | System Manager).

In order to meet this challenge, reviews are commissioned or, if necessary, test designers review their own test case specification, whereby an independent review is considered to be better: “Well, testing somehow doesn’t yet have great significance. The system specification clearly follows the principle of dual control. And if anything in the specifications is changed, the change will be reviewed.” (Int06 | System Manager). In some cases, **reviews are only performed on a random sample** due to an increasing number of test cases related to the increased complexity of the systems. Here, requirements coverage, syntactic aspects are checked and whether all mandatory test case attributes have been specified. Checking semantic aspects (content-related compliance with the requirement) is hard to verify and, therefore, an expert with system understanding is required for reviewing the test case specification.

The possibility of performing internal walkthroughs with colleagues who have similar specialist knowledge (e.g., exterior vs. interior light system), the development of review checklists, guidelines, and stricter controls were described by the interviewees as helpful:

“This is always difficult to say, but I always find it helpful to have more control.” (Int13 | System Manager).

5.9 Tool-related problems (M9)

Category definition. This category addresses challenges related to poor usability and missing functionality of tools, the use of heterogeneous tool chains, and deficiencies with the support. The challenges presented here are not limited to a specific tool, but have been generalized as far as possible and can also occur in other tools.

5.9.1 Usability and function-related problems (ToC-9.1)

Interviewees mentioned often **usability problems with the used tools** or missing knowledge about the usage. These problems include difficulties with simple copy & paste tasks, editing test cases, creating traceability between different artifacts or exporting and importing content within the tool chain used. This is evident from the following examples:

“This is difficult to handle. I’ve to double click in, ... then a field opens, then I’ve to click the fields individually. That’s just a huge amount of clicking.” (Int10 | Project Manager).

“Unfortunately, there are no regular expressions to replace [a specific text].” (Int11 | Test Manager).

“The ... import is always a bit exciting. It breaks off more often.” (Int06 | System Manager).

Furthermore, interviewees mentioned performance problems:

“The [used tool] is very confusing and not performant.” (Int05 | Function Developer).

“The popular [irony] import that makes my life very, very difficult.” (Int04 | Test Manager).

As a result of the poor usability or even missing functionalities of the tools used, certain tasks can only be implemented with difficulty and with considerable time effort or, in the worst case, cannot be realized at all. In particular, the interviewees complained about the resulting additional time expenditure: “I can’t mark everything and change it simultaneously, I’ve to do everything individually. And that sucks, because that’s another half hour of work, where I always do the same process step.” (Int14 | System Manager).

5.9.2 Heterogeneous tool chains (ToC-9.2)

Guidelines and overviews of the functional interaction of the different tools in a tool chain are missing. This is shown by unawareness or by the fact that **test cases are not created according to guidelines** with the appropriate tools:

“Some colleagues say: ‘I’m not even aware that this [the test case specification] is also in [Tool X]’. ... Sometimes we don’t get the test cases in the ... template as we would like them to be. ... [Instead], they are written in Word [or] in Excel.” (Int03 | Test House Manager).

The employed tool chains make it difficult to **react flexibly to changes and a consistent traceability cannot always be established**: “Traceability is therefore not ensured.” (Int03 | Test House Manager).

And one is aware of this lack: “Results are only reported back in case of an error ... We want to address this in a new concept ... in order to completely ensure this traceability.” (Int08 | Test Manager).

Furthermore, the **integration of additional functionalities** to existing tools used for test management was highlighted as challenge, because non-matching subsystems delivered by different suppliers due to the lack of precise specifications.

“The compatibility across the different tools and versions, with the interface to Tool X or the import to Tool Y, is problematic.” (Int13 | System Manager).

“I’ve about nine different construction sites through these ‘import stories’ or THIS tool world ... Six of them have been worked through, the others are still open and one seems insoluble.” (Int15 | Test Manager).

Often it takes a lot of time to carry out the tasks with the available tools. It can lead to “trouble and project delay ... because I can’t provide the test cases in time.” (Int15 | Test Manager).

Interface problems are usually solved in an unconventional way, which sometimes causes new problems: “Then we make the change with an Excel exchange, because it always works. But then the traceability is even worse.” (Int13 | System Manager).

5.9.3 Deficiencies with the support (ToC-9.3)

Furthermore, interviewees reported that in cases of individual problems or project-specific requirements the **support cannot timely provide a solution**. This is reflected in statements such as: “No one really knows what the problem is and the support lacks the flexibility to change things quickly and easily.” (Int14 | System Manager).

“If you have to wait half a year until you get some change implemented, then it really can’t be.” (Int17 | Tester).

Unfortunately, this lack of support for project-specific issues often leads to the development of individual solutions: “From my point of view this leads to many people developing their own tools” (Int17 | Tester). However, such tools often do not fit seamlessly into the existing tool chain.

5.10 Discussion of the identified challenges

An overview of the developed taxonomy of test case specification challenges is initially presented in Table 4, which is our answer to **RQ1**. We provided examples of challenge types (ToCs) for each main category (M1 – M9) in the previous subsections. We pointed out causes and consequences as well as applied solutions, which were known to the practitioners (**RQ2**).

In the following, we summarize our results and relate the identified challenges by means of the main categories to the suspected areas, as shown in Table 5. We highlight the areas that are significantly affected by the challenges in the table using a ✓. Furthermore, we assigned challenges identified in Section 3 from related work ($C_{RW01} - C_{RW25}$) to the main categories.

Challenges related to the availability of input artifacts (**M1**) occur mainly while creating (C) test case specifications and inhibit test case design. Similar to Lachmann and Schaefer (2014, 2013), we identified the lack of test plans as an important challenge. As a solution in practice, we have observed rudimentary standard test plans. We assigned this challenge to the type of non-existing input artifacts (**ToC-1.1, 5.1.1**). We identified the distribution (**ToC-1.2, 5.1.2**) and access restrictions of input artifacts (**ToC-1.3, 5.1.3**) especially in collaboration with suppliers as new challenges concerning test case specification, which were not mentioned in related work.

Table 5 Challenge categories related to areas and related work

Categories									
Area	M1	M2	M3	M4	M5	M6	M7	M8	M9
(C) Creation	✓	✓	✓	✓	✓	✓			✓
(P) Processing		✓	✓	✓	✓	✓	✓		✓
(Q) Quality			✓	✓	✓			✓	✓
Related work challenges	<i>C_{RW}06</i>	<i>C_{RW}01</i> , <i>C_{RW}03</i> , <i>C_{RW}04</i> , <i>C_{RW}05</i>	<i>C_{RW}07</i> , <i>C_{RW}08</i> , <i>C_{RW}09</i> , <i>C_{RW}10</i>	<i>C_{RW}11</i> , <i>C_{RW}23</i> , <i>C_{RW}24</i> , <i>C_{RW}25</i>	<i>C_{RW}02</i>	<i>C_{RW}13</i> , <i>C_{RW}14</i> , <i>C_{RW}15</i> , <i>C_{RW}16</i> , <i>C_{RW}17</i> , <i>C_{RW}18</i>	<i>C_{RW}19</i>	<i>C_{RW}12</i>	<i>C_{RW}20</i> , <i>C_{RW}21</i> , <i>C_{RW}22</i>
References	Lachmann and Schaefer (2014, 2013)	Grimm (2003), Pretschner et al. (2007), (2013), Kasoju et al. (2017), Garousi et al. (2017), Sundmark et al. (2011), and Lachmann and Schaefer (2013)	Kasoju et al. (2013), Petrenko et al. (2015), Garousi et al. (2017), and Lachmann and Schaefer (2013)	Kasoju et al. (2013), Lachmann and Schaefer (2014), and Garousi et al. (2017)	Grimm (2003), Pretschner et al. (2007), and Sundmark et al. (2011)	Grimm (2003), Pretschner et al. (2007), Kasoju et al. (2013), Sundmark et al. (2011), and Lachmann and Schaefer (2013)	Kasoju et al. (2013), and Sundmark et al. (2011)	Kasoju et al. (2013), and Garousi et al. (2017)	Grimm (2003), Kasoju et al. (2013), Broy (2006), Garousi et al. (2017), and Sundmark et al. (2011)

Content-related problems with input artifacts (**M2**) concerning requirement specifications (**ToC-2.1**, **5.2.1**), such as the lack of requirements clarity and incomplete or conflicting requirements, lead to re- and misinterpreting were mentioned by Garousi et al. (2017) and Kasoju et al. (2013). Such problems have a significant impact of the quality of test design and therefore occur mainly while creating (C) test case specifications. In the worst case, problems can also occur in the further processing (P) for the first time. We identified new content-related challenges for test case specifications regarding test plans (**ToC-2.2**, **5.2.2**), test case specification templates (**ToC-2.3**, **5.2.3**) and incorrect previous test cases (**ToC-2.4**, **5.2.4**). Interviewees highlighted the usage of customized templates and we identified some aspects that should be supported by a template, i.e., variant handling, managing reusable parts.

Similar to Kasoju et al. (2013), we identified challenges related to a lack of knowledge (**M3**), which affects all areas and roles (C, P, Q). Test designers, testers and reviewers must have sufficient knowledge of the system under test (**ToC-3.1**, **5.3.1**), test platforms used (**ToC-3.2**, **5.3.2**), and test policies (**ToC-3.3**, **5.3.3**) in order to create, process, and review high-quality test cases. This also means that there must be a correlation between domain knowledge and tester skill, as indicated in Garousi et al. (2017) and Kasoju et al. (2013).

Test case description related problems (**M4**), such as challenges related to the phrasing of test cases (**ToC-4.2**, **5.4.2**) and in particular natural language-based test cases may cause problems (e.g., lot of manual activities to uniform test cases, understanding and completeness of test cases in processing test cases and extensive quality checks) in all areas (C, P, Q). Similar to Lachmann and Schaefer (2014), we detect also a variation of phrasing and problems related to the excessive use of abbreviations. In addition, we identified the consequences: e.g., ambiguity and misunderstanding during test implementation. We further identified a new challenge concerning the correlation between test case description and the target test platform. Besides, language-based problems are mainly associated with translation and spelling errors. In addition, we identified challenges related to language-based problems (**ToC-4.1**, **5.4.1**), which mainly arise in terms of translation and spelling errors. We agree with Garousi et al. (2017) and Kasoju et al. (2013) that methods for documenting and tracing experience-based test cases are still missing. Moreover, we stated typical quality-based problems in test cases (**ToC-4.3**, **5.4.3**) related to the completeness (e.g., missing preconditions and undefined test case meta data), changeability, and reusability of test cases.

The huge number of variants is still challenging in the automotive domain, as stated by Pretschner et al. (2007), even in the creation of a test case specification (**ToC-5.1**, **5.5.1**). We observed that this cannot be solved by the existing test case specification template so far. Hence, different departments develop individual solutions. This reflects the well-known challenge, also stated by Grimm (2003), that each discipline has its own processes, methods, and tools. Interdisciplinary exchange between different departments rarely takes place. Furthermore, we identified the new type of challenge regarding the focus on a specific test platform (**ToC-5.2**, **5.5.2**) in the category test case content-related problems (**M5**). In this case, the test platform-dependent distribution of test cases intensifies the problem of redundant test case execution.

We identified various process-related problems (**M6**). The lack of standards or guidelines (**ToC-6.1**, **5.6.1**) particularly influences the creation (C), while missed change requests (**ToC-6.2**, **5.6.2**) mainly occur in the processing (P) of a test case specification (e.g., test cases do not match the SUT). Similar to Kasoju et al. (2013) and Sundmark et al. (2011) this is due to poor change management and a lack of traceability management. The increasing outsourcing of audit-related tasks and the resulting consequences was emphasized by

the interviewees and perceived by us as a new challenge in the context of organization-related decisions (**ToC-6.3**, **5.6.3**). Furthermore, similar to Lachmann and Schaefer (2013), we observed the lack that no test manager is nominated, which affect the area of processing (P).

Communication-related problems (**M7**, **5.7**) occur mainly between test designers and testers and therefore they are related to the further processing of a test case specification (P). Similar challenges were identified by Kasoju et al. (2013) regarding the lack of interaction between costumers and test personnel. However, they rather describe process-related communication problems and do not refer directly to test case specification issues.

Quality-related problems (**M8**, **5.8**) mainly address quality assessment activities and affect only quality aspects (Q). We would like to emphasize that answering the question “What is a high-quality test case specification for you?” was difficult for many interviewees. This suggests an insufficient (known) definition of quality (**ToC-8.1**, **5.8.1**). Quality aspects are taken into account in operational activities, but they could not be named explicitly. For example, test designers pay attention to writing test cases in a comprehensible and complete manner, but they only perceive it unconsciously as a quality feature. Moreover, there is a lack of useful quality metrics (**ToC-8.2**, **5.8.2**) for checking test case specifications. Quality metrics for test case specifications could make a valuable contribution to a better understanding of quality and to support test case specification reviews. In this context, similar to Kasoju et al. (2013), the interviewees reported a lack of established review processes and review guidelines (**ToC-8.3**, **5.8.3**).

The challenge of insufficient tools (**M9**, **5.9**) was mentioned as a still existing problem for various test activities (C, P, Q), stated in multiple related works (Grimm 2003; Kasoju et al. 2013; Lachmann and Schaefer 2014; Petrenko et al. 2015; Garousi et al. 2017). In particular, we identified a challenge in the lack of tool support for an automation of quality assessments especially for test case specifications. Furthermore, we observed that a major part of the creation of a test case specification is still done manually. This aspect was also mentioned in Tierno et al. (2016) using the example of various existing model-based testing concepts, which are still not used by OEMs. We distinguish between challenges regarding the usability and functionality of a tool (**ToC-9.1**, **5.9.1**), the use of heterogeneous tools (**ToC-9.2**, **5.9.2**), and deficiencies in the assistance provided by the support (**ToC-9.3**, **5.9.3**).

We have encountered similar challenges in all four companies as well as in the different departments and assume that these challenges may be similar for other departments and companies in the same environment. Moreover, we believe that these challenges may also occur in other industries with similar conditions as in the automotive domain, such as outsourcing of development activities, distributed teams, or handling of different product lines.

The identified challenges that form the basis for the further investigation in the descriptive study are listed in Table 6. Furthermore, this table provides a summary of the consequences and named solutions for each challenge, which is our answer to **RQ2**. If the interviewees did not name any existing solutions, this is marked as *not indicated* in the table. If they mentioned ideas, these are marked as *suggestions* in the table.

6 Results of the descriptive survey

In this section, we present the results of our descriptive survey. For this purpose, we analyzed all 36 completed questionnaires. We investigated how practitioners assess the identified

Table 6 Overview of consequences and solutions proposed by the practitioners for the identified challenges

ID	Challenge	Consequences	Named solutions
M1: Availability problems with input artifacts			
C1, C2	Requirements specification does not exist/does not exist in time	Delays the creation of test cases	Test cases are created based on previous test cases, using placeholders
C3	Test plan does not exist	Lead to inefficient, highly redundant testing across multiple test platforms, insufficient utilization of test platforms, increasing test effort and costs	Using a rudimentary standard test plan
C4	Information and documents are distributed	Gathering information takes time	Enriching specifications with necessary additional information
C5	No access to relevant documents	Information gaps, poorer system understanding, incomplete or incorrect test cases	Communication is necessary, data is provided manually
M2: Content-related problems with input artifacts			
C6	Requirement specification is outdated	Necessary information for creating test cases is missing	<i>Not indicated</i>
C7 – C11	Incorrect, conflicting, obsolete, incomplete, or unintelligible requirements exist	Influences the creation of high-quality test case specifications, increasing test effort	Communication with those responsible for the requirement specification is necessary, provision of additional information to encourage understanding of the requirements
C12 – C15	Test plan is too general/extensive, insufficient description of the test object, undefined test end criteria	Increasing test effort, inefficient or incorrect usage of test platforms, missing test cases for specific test platforms	Conducting test plan reviews
C16	Template does not fit	Test cases cannot be adequately documented, test case duplicates exist (e.g., for different variants)	Modification of the template, parameterizing test cases
C17	Faulty test cases of previous versions	Test cases contain incorrect or obsolete information	Parameterizing test cases, using placeholders

Table 6 (continued)

ID	Challenge	Consequences	Named solutions
M3: Knowledge-related problems			
C18	Insufficient knowledge about SUT	Derivation of incorrect test cases, errors in test case specifications	Workshops with all relevant participants to fill knowledge gaps regarding the SUT
C19, C20	Insufficient knowledge about available test platforms/test platform functionalities	Lack of meaningful test cases for specific test platforms, suboptimal assignment of test cases to test platforms and late error detection, test cases do not fit the test platform	Testers provide sample test cases for specific test platforms
C21 – C24	Insufficient knowledge about testing guidelines, lack of training, consulting and contact persons, insufficient documentation of the template	Test cases are not documented template compliant, missing information in test cases	<i>Suggestions:</i> One-sided instructions, continuous and short refresher courses, newsletter, expert hotline
M4: Test case description related problems			
C25, C26	Translation and spelling errors in test cases exist	Lead to false test cases	Using formal approaches (e.g., description of test cases at signal level)
C27	Several authors are working on a test case specification	Ambiguities and misunderstandings during test implementation, non-uniform test case descriptions	Subsequent revision and unification of phrasing in test cases, using previously defined phrase blocks
C28	Phrasing of test cases in prose	Ambiguities and misunderstandings during test implementation	Subsequent revision and unification of phrasing in test cases, using previously defined phrase blocks
C29, C30	Use of (undefined) abbreviations	Impair comprehensibility/readability of a test case	Using glossaries
C31	Abstract test case description leads to inaccuracies in test cases	Interpretation during test case implementation	<i>Not indicated</i>
C32	Phrasing of a test case is not specific to the respective target test platform	Necessary information is missing, test cases cannot be performed on the assigned test platform	<i>Suggestion:</i> Phrasing should be adapted to target test platform
C33	Typing errors in test cases	Usually no effects, may affect the content of a test case (e.g., fundamentally different meaning)	Communication with responsible persons necessary if deficiencies are noticed

Table 6 (continued)

ID	Challenge	Consequences	Named solutions
C34 – C43	Incomplete test cases exist (missing documentation of: preconditions, actions, expected results, test purpose, prioritization, model series, test platforms, origin, test case derivation procedure)	Increasing communication effort to clarify missing information, testers are not able to identify relevant test cases for their test platform, filtering of test cases not possible (e.g., by model series), lack of traceability of the test case	Communication with responsible persons necessary if deficiencies are noticed
C44, C46, C49	Test case description is incomprehensible, ambiguities in test cases exist, poor readability/comprehensibility of a test case	Interpretation during test case implementation	Communication with responsible persons necessary
C45	Sequence of test case actions to be executed is unclear	Interpretation during test case implementation, lead to incorrect test cases implementation	Communication with responsible persons necessary
C47, C48, C50	Quality parameters of a test case: too many functions, linked requirements, or parameters	Impair the verification of the (content) completeness of a test case	<i>Not indicated</i>
C51	Test case already contains implementation details	Restriction of the tester/implementer	<i>Not indicated</i>
C52, C53	(Experience-based) Test cases are not reused	Test cases were often redesigned instead of using existing ones, experience-based test cases are lost	Manual copying and adaptation of test cases, documentation of experience-based errors in requirement specification
C54	Changes to a test case are time-consuming	Changes are often not made immediately	Tedious manual searching and replacement of affected parts, usage of placeholders, extraction of reusable sequences into a base scenario
C55	Test case is not consistent with requirements	Incorrect test case is executed and does not contribute to the detection of errors	<i>Not indicated</i>
M5: Test case specification content-related problems			
C56	Insufficient handling of variants in the test case specification	Lack of documentation of the variants, lack of clarity of the assigned variant per test case	Various project-specific methods exist (e.g., parameterizing test cases)
C57	Many different model series in one test case specification	Lack of documentation of the model series	Various project-specific methods exist (e.g., one test case specification per model series)
C58	Only test cases for a specific test platform are contained	Redundant execution of test cases on different test platforms, difficulties in verifying the completeness of test scopes	<i>Suggestion:</i> Test specification contains test cases for all relevant test platforms

Table 6 (continued)

ID	Challenge	Consequences	Named solutions
M6: Process-related problems			
C59, C61	Guidelines are missing	Testing process differs (e.g., depending on the department)	Developing project-specific guidelines <i>Suggestion:</i> Mandatory introduction of a test manager for each project
C60	Templates for test reports are missing	difficult to make conclusions about the status of a system's test progress	<i>Suggestion:</i> Uniform guidelines/templates for reporting test progress
C62, C63	A defined reference process for testing is missing, unclear interface definition to the overall process	Unclear responsibilities	<i>Suggestion:</i> A company-wide reference process
C64	Lack of change management	Leads to ignorance of changes	Own processes are developed
C65	Test process does not fit the project/system	Required test activities are not executed (e.g., test case specification is not available in time)	<i>Suggestion:</i> there should be a test manager
C66	Test manager is vacant	Test-relevant tasks (e.g., reviews) are not processed	<i>Suggestion:</i> Revision of responsibility descriptions (e.g., for system managers)
C67	Responsibilities and roles are unclear	Leads to knowledge gaps, increased effort and test execution takes longer than in-house	<i>Not indicated</i>
C68 – C70	Increasing outsourcing		
M7: Communication-related problems			
C71, C72	Communication problems with the supplier, communication via representatives	Lead to misunderstandings (“Chinese Whispers”-problem)	Workshops with all relevant participants
C73, C74	Communication problems due to distributed teams/spatial distance	Reduced communication	<i>Not indicated</i>
C75	Communication problems due to cultural differences	Lead to misunderstandings (e.g., due to shyness to ask in case of ambiguities)	<i>Not indicated</i>
C76	Communication problems due to language barriers	Lead to misunderstandings (e.g., due to translation errors)	Using previously defined phrase blocks
C77	Different expectations	Lead to misunderstandings (e.g., due to different assumptions about the common knowledge base)	<i>Not indicated</i>

Table 6 (continued)

ID	Challenge	Consequences	Named solutions
M8: Quality assurance related problems			
C78	Quality characteristics for a test case specification are unknown	Unclear which criteria have to be checked, e.g., in a review	<i>Not indicated</i>
C79, C80	Established metrics for quality assessment are missing (only requirement coverage is well-known)	Complicates the quality assessment of a test case specification	<i>Suggestions:</i> Some ideas for metrics (e.g., number of test steps or number of used words in a test case, etc.)
C81	Tools used do not generate quality reports	Only requirement coverage is calculated by the tools and therefore the only well-known metric	<i>Not indicated</i>
C82	Company guidelines for quality measurement are missing	Review process varies (e.g., depending on the department)	<i>Suggestions:</i> Providing review checklists and guidelines
C83 – C86	Lack of established review processes, due to shortage of manpower, lack of time or test case specifications are too large to conduct (full) reviews	Authors of a test case specification perform a review themselves (no independent review), reviews are only performed on a random sample	Commissioning of reviews <i>Suggestions:</i> Performing internal walkthroughs with colleagues with similar specialist knowledge, providing review checklists and guidelines
M9: Tool-related problems			
C87, C90	Heterogeneous tools are used, non-continuous tool chains (interface problems) exist	Time-consuming, project delays to carry out the tasks with the available tools, difficult to react flexibly to changes, consistent traceability cannot always be established	If export and import do not work, data are exchanged manually (e.g., via Excel)
C88	Tools used are complex and functions are unknown	Considerable time effort necessary to complete tasks	<i>Suggestion:</i> Using another tool
C89	Usability deficiencies	Tasks are difficult to perform	<i>Suggestion:</i> Using another tool
C91	Project-specific adaptations	Lead to problems if adaptation does not fit into existing tool chain	<i>Not indicated</i>
C92	Deficiencies with the tool support	Support cannot help, changes to tools cannot be implemented on time	Developing own tools, which often do not fit seamlessly into existing tool chains

challenges (**RQ3**) in terms of the frequency of occurrence and criticality. These results are presented in Section 6.1. In order to identify differences between external and internal employees regarding the occurrence of challenges (**RQ4**), we used Fisher's exact test as described in Section 4.2. In addition, to determine the effect strength for identified significant differences between the groups, we calculated the ϕ coefficient as a correlation measure. These statistical results are presented in Section 6.2.

6.1 RQ3: Assessment of the identified challenges

For answering **RQ3**, we present the results of the descriptive survey according to our developed taxonomy of test case specification challenges (cf. Table 4). Therefore, the following subsections are structured in accordance to the main categories (M1 – M9). For each main category, we consider the assessments of the associated challenges based on their frequency of occurrence and criticality. We present the results of all challenges per category with one diagram for frequency of occurrence and one for criticality, such as Figs. 6 and 7 for category M1. On the left side of the figures, the challenges are listed, which have been identified in the exploratory case study (C1 – C92). To the right of each challenge is the percentage of survey participants who answered the questions. In the case of frequency of occurrence, the response scale ranges from “does not occur” to “very often.” In this case we extended the original five-point Likert scale, as shown in Fig. 5, by the value “does not occur.” For the purpose of a complete presentation of the results regarding the frequency of occurrence of the challenges, we considered the integration of the answers “does not occur” as necessary in the following diagrams. The number of survey participants who did not rate a challenge (“no comment”) is shown on the right side of each figure (gray bars). These survey participants did not have to provide any further information about frequency or criticality. If the survey participants rated a challenge as does occur, which means that this challenge occurs at least “very rare,” we were interested in how critically they assessed this challenge. We assume that survey participants who actually face a challenge, even if only very rarely, can make a more precise statement about their criticality than others. These results are presented in a similar diagram, which shows the percentage of survey participants who rated each challenge as “not critical” and “rather not critical” (left percentage), “partly critical” (middle percentage), or “rather critical” and “very critical” (right percentage). The diagram next to the criticality assessment shows how many participants have answered this question (“rated”). Participants who did not assess a challenge (“no comment”) or assessed the challenge as “does not occur” are summarized as “not rated.”

6.1.1 M1: Availability problems with input artifacts

Survey participants rated the challenges from category M1 as more likely to occur. As illustrated in Fig. 6, between 15 and 76% rated the frequency of occurrence of the challenges

Fig. 6 Frequency assessment for category M1: Availability problems with input artifacts

Fig. 7 Criticality assessment for category M1: Availability problems with input artifacts

from “sometimes” to “very often.” Two out of five challenges (**C3** and **C4**) were rated over 50% as at least “sometimes” occurring. For survey participants, challenge **C4** (information and documents are distributed) is the most common, resulting in an overall score of 76%. Challenge **C3** (test plan does not exist) occur more frequently with 52%. However, both challenges are not classified as very critical overall (cf. Fig. 7). As it was mentioned in the interviews, missing test plans (**C4**) leads to increased testing effort, which could explain why 25% of 31% consider it to be “very critical.” The frequent occurrence of the challenge **C4** can be attributed to the high complexity of the systems to be developed. The opposite low level of criticality (33% rather not critical) may be due to the fact that the parties involved have developed strategies for gathering information, such as enriching specifications with additional information.

As shown in Fig. 6, challenge **C1** was rated with 85% as predominantly rare occurring to does not occur. However, this challenge is considered most critically with 50% (cf. Fig. 7). This statement can be regarded as convincing, since requirements-based testing procedures are still widely established in the automotive industry and in this context requirement specifications are a fundamental input artifact for creating test case specifications.

6.1.2 M2: Content-related problems with input artifacts

Challenges from category M2 were mainly rated as occurring by survey participants. Half of the challenges were rated at least 50% as “sometimes” to “very often,” as shown in Fig. 8. In particular, the frequency of challenges with respect to requirement specification issues were more frequently rated with “sometimes” to “very often.” This can be observed for the challenges **C10** (incomplete requirements), **C11** (unintelligible requirements) and

Fig. 8 Frequency assessment for category M2: Content-related problems with input artifacts

Fig. 9 Criticality assessment for category M2: Content-related problems with input artifacts

C6 (entire requirement specification is outdated) with 53 to 57%. It is more likely that individual requirements are outdated than the entire requirement specification.

That the three most common challenges are related to requirement specification issues can be explained as both the test designer and the tester are usually familiar with the requirement specification. Hence, this document is also known to a larger number of survey participants. In contrast, the test plan is more relevant for system or test managers who, for example, commission the creation of a test case specification. Testers seldom have insight into it. At position four of the most frequently occurring challenges is challenge **C12** (test plan is too general) with 52%. The high frequency of this challenge can be explained by the increased usage of standard test plans mentioned in the interviews. These contain a comprehensive test strategy that usually has to be adapted to the system to be developed, which is sometimes not done. The criticality of this challenge is rated lowest with 5%, as shown in Fig. 9. This rating could be explained as it is better to have a general test plan instead of none at all (cf. challenge **C3**). The rare occurrence of conflicting (**C8**) and incorrect (**C7**) requirements between 29 and 37% is noteworthy. This could be a sign of the effectiveness of the quality checks for requirements that have been introduced in the investigated OEM in recent years. Similarly, the rare occurrence of challenge **C14** (insufficient description of the test object) indicates that if a test plan exists, the contained description of the test object is sufficient.

The assessment of criticality also reveals that problems regarding the requirements (**C6**, **C7**, **C8**, **C9**) tend to have critical effects on the test case specification (cf. Fig. 9). It is noteworthy that challenge **C16** (template does not fit project-specific needs) was considered rather uncritical (43%). It is known from the interviews that numerous individual solutions exist which extend the standard test case specification template or reinterpret the template guidelines.

6.1.3 M3: Knowledge-related problems

Figure 10 shows that survey participants predominantly rated the challenges of category M3 as not occurring (cf. large percentage of “does not occur” ratings). This shows that the majority of survey participants are not aware of any gaps in their knowledge. However, these statements may be influenced by the fact that individuals are reluctant to admit their own shortcomings. In contrast, if survey participants are confronted with knowledge-related

Fig. 10 Frequency assessment for category M3: Knowledge-related problems

problems, they rate them as critical. This can be seen in Fig. 11, where five out of seven challenges were assessed as predominantly critical (between 35 and 50%).

As illustrated in Fig. 10, the high demand for trainings (C22) regarding test case specifications and insufficient knowledge about the SUT (C18) between 53 and 68% is noteworthy. In addition, both challenges are regarded as rather critical between 43 and 50% (cf. Fig. 11). It is obvious that the lack of knowledge about the system to be tested (C18) is assessed most critically. If a test designer does not know the system, it is difficult to develop meaningful test cases. Furthermore, the survey results underpin the statements from the interviews on the described lack of company-wide testing guidelines, such as the lack of training (C22, 68%), lack of consulting and contact persons (C23, 47%) or the insufficient knowledge about testing guidelines (C21, 45%). These challenges are also assessed as rather critical (between 35 and 43%), as can be seen from Fig. 11. In addition, there is the insufficient documentation of the test case specification template (C24), which is also assessed as rather critical (42%).

6.1.4 M4: Test case description related problems

Figure 12 shows that survey participants rated the frequency of challenges of category M4 between 18 and 67% as “sometimes” to “very often.” From a total of 31 challenges, 17 were assessed as predominantly occurring, which is approximately 55% of the challenges of category M4. Predominantly occurring means that these 17 challenges were rated by at least 50% of the survey participants from “very rare” to “very often” (applicable for C26, C27, C29 – C34, C89, C39, C42, C44, C46, C47, C49, C54, and C55). The other 14 challenges were assessed as predominantly “does not occur.”

As can be seen in Fig. 12, the first nine challenges occur more frequently than others (between 51 and 67%). This includes that changes to a test case are often time-consuming

Fig. 11 Criticality assessment for category M3: Knowledge-related problems

Fig. 12 Frequency assessment for category M4: Test case description related problems

(C54), and information about the used procedure for test case derivation (C43), the origin (C42), and the prioritization of a test case (C39) is often missing. Moreover, this also concerns incomprehensible test case descriptions (C44) and incomplete test cases (C34) or that the phrasing of a test case is not specific to the respective test platform (C32). Spelling errors in test cases (C26) lead the ranking with 67%, but are rated as rather uncritical (88%, cf. Fig. 13). Regarding criticality, the same applies to typing errors in test cases (C33, 85%). Another language-based problem concerns translation errors in test cases (C25), which have been rated as rather rare (68%) and uncritical (57%). The low frequency and criticality of this challenge could be explained by other characteristics of the survey participants. We also asked in the questionnaire for the native language and the common language of the test case specifications. The majority of the survey participants deal with German test case specifications and are native speakers (50%), whereas only 6% are not German native speakers. Thirty-six percent of the specifications are written in English by non-native speakers and 8% bilingual in German and English by German native speakers. However, as we know from the interviews, the possible effects of challenges C25 and C26 should not be underestimated.

Figure 13 shows that most challenges in category M4 are rated as rather uncritical by the survey participants (22 of 31 challenges). The challenges of undocumented preconditions (C35), actions (C36) and expected results (C37) are considered rather critical by most

Fig. 13 Criticality assessment for category M4: Test case description related problems

survey participants (between 46 and 62%). This is quite obvious, since these attributes contain the main information of a test case: the description of the test case. Another challenge that has been rated as rather critical (42%) concerns incomprehensible test case descriptions (C44). Moreover, this challenge seems to occur more frequently (55% rather often, cf. Fig. 12). Insufficient and incomprehensible test case descriptions mean an increased effort, as for example necessary queries by the tester before a test case is implemented. Especially in the interviews, the use of phrases in prose was cited as the cause of incomprehensible test case descriptions. An interesting aspect is that 70% of the survey participants assessed that the challenge of misunderstandings caused by test cases formulated with prose is less likely to occur (C28, cf. Fig. 12). The majority (58%) of the survey participants rated this challenge as “does not occur” occurring. Moreover, this challenge was assessed as rather uncritical by 43%. Misunderstandings do not seem to be the biggest problem with prose test cases. As interviewees explained, readability and changeability play a decisive role in this context. In particular, the survey seems to confirm that changing a test case is usually a time-consuming task (C54). Sixty-two percent agree that it is more time-consuming and 35% rated it critical.

A challenge identified for the first time in the interviews is also reflected in the survey results: the formulation of a test case should be test platform-dependent (C32). According to the reports from the interviews, the survey results also show with 55% that the phrasing

Fig. 14 Frequency assessment for category M5: Test case specification content-related problems

of test cases is often not adapted to the target test platform. However, a test case that is not formulated in accordance with the test platform is predominantly regarded as partially critical by 45%. There are two reasons for this average assessment of criticality. Either test cases cannot be executed on the test platform and are rejected by the test platform (less critical) or such test cases lead to considerable additional effort for clarification (more critical).

Furthermore, the challenge of not reusing test cases (C52 and C53), already described in the interviews, becomes clear from the survey results. Thirty-eight percent of survey participants rated these challenges as more frequent and especially C53 with 46% as rather critical. This criticality is obvious, since knowledge is lost through non-reused experience-based test cases, whereby these test cases are usually very useful test cases.

6.1.5 M5: Test case specification content-related problems

Figure 14 shows that the challenges C56 (insufficient handling of variants) and C57 (many different model series) occur more frequently between 58 and 68%. The increasing system complexity is a well-known challenge and can be perceived here on the basis of model series and the problem of inadequate variant management.

All challenges (C56 – C58) seem to be rather uncritical, between 35 and 59% (cf. Fig. 15). Only 29% of the survey participants consider variant management to be a critical challenge. The focus of a test case specification on test cases for a specific test platform and thus the distribution of the total number of test cases to different documents is rated as rather uncritical with 59%.

6.1.6 M6: Process-related problems

As shown in Fig. 16, survey participants predominantly rated process-related challenges (M6) as “sometimes” to “very often” occurring. Eight out of twelve challenges exceed the 50% threshold (between 52 and 70%).

Figure 16 shows that the identified challenges are very common for the survey participants. In particular, the first five challenges tend to occur between 67 and 70% more frequently. All challenges are assessed as rather critical (between 29 and 68%). The negative effects of outsourcing test activities mentioned in the interviews are also assessed in this way by the majority of the survey participants. Most of the survey participants (between 68

Fig. 15 Criticality assessment for category M5: Test case specification content-related problems

Fig. 16 Frequency assessment for category M6: Process-related problems

and 70%) answered that outsourcing related challenges occur “sometimes” to “very often,” such as the loss of knowledge (C68), the longer duration of the test execution (C70) and the increased effort (C69). In particular, all three outsourcing-related challenges were assessed most critically (more than 64%, cf. Fig. 17). In the case of losing knowledge about the test case specification when outsourcing, none of the survey participants assessed this as rather uncritical or not critical. In addition, the survey participants seem to consider the role of a test manager to be important, as 56% consider the lack of this role to be rather critical (C66). This is an interesting aspect considering that this challenge occurs rather frequently with 54%.

6.1.7 M7: Communication-related problems

For each question of category M7, the answers are shown in Fig. 18. Five of the seven challenges tend to occur more frequently (between 61 and 76%). Between 6 and 19% did not answer the questions. Hence, category M7 has the highest response rate. This shows that most survey participants face communication-related challenges. In general, most of the survey participants (76%) answered that communication problems with suppliers (C71) occur “sometimes” to “very often” (cf. Fig. 18). However, the aspect that communication

Fig. 17 Criticality assessment for category M6: Process-related problems

Fig. 18 Frequency assessment for category M7: Communication-related problems

with the supplier usually takes place via representatives (C72) was rated by 68% as rather frequent.

In Fig. 19, it is shown that a majority of the survey participants (48%) rated this challenge as rather critical. This also reflects the findings from the interviews that communication via representatives is perceived as fundamentally challenging. Furthermore, the survey participants rated the challenge of spatial distances (C74) as more frequent with 68%. This aspect also often affects suppliers and can be a reason for communication problems with the supplier. Challenges relating to communication problems due to cultural differences (C75) or language barriers (C76) occur less frequently (55% and 55%), but are assessed as rather critical with 47%. That was also mentioned in the interviews that when other cultures or non-native speakers are involved in projects, this is challenging. It can be assumed that a high proportion of survey participants who never face challenges C75 and C76 are not involved in international projects or work alone on the test case specification. Unfortunately, this is not evident from the recorded data.

6.1.8 M8: Quality assurance related problems

Figure 20 shows that survey participants are predominantly confronted with the identified challenges. Basically, challenges regarding quality assurance related problems (M8) can be considered as frequently occurring, because they were answered between 57 and 81% with “sometimes” to “very often.” It is impressive that all of the identified challenges have been rated as occurring more frequently by the survey participants with more than 57% (cf. Fig. 20). This underlines the fact that quality assurance measures in particular have to be introduced to improve the quality of automotive test case specifications and it also confirms the necessity of our research. This is also indicated by the fact that in our investigated research context, the lack of existing metrics for quality evaluation of test case specifications (C79 and C80) is assessed by 61% and 75% of the survey participants as rather frequent.

Fig. 19 Criticality assessment for category M7: Communication-related problems

Fig. 20 Frequency assessment for category M8: Quality assurance related problems

Furthermore, challenge **C79** was rated as most critical by 67% of the survey participants (cf. Fig. 21). The lack of established review processes (**C83**) was rated as particularly frequent by 64%. In this context, we assume correlations with challenges **C78**, **C79**, and **C80**. In addition, organizational challenges, such as lack of staff (**C84**, 81%) and time (**C86**, 80%), were rated as occurring most frequently. Moreover, effects of system complexity can also be seen here. Test case specifications partly comprise several 1000 test cases and are very large. One interviewee mentioned that it would take longer to review a test case specification than to write it and therefore not all test cases can be reviewed. The survey results underpin this statement that the size of test case specifications often means that only random reviews can be conducted (**C85**, 67%). The challenge that tools do not provide automated quality reports (**C81**) was rated as rather uncritical by 50%. This could be related to the lack of understanding of quality metrics (**C79**). Since other metrics than requirement coverage rarely seem to be known, no tool that calculates quality characteristics is missing either.

6.1.9 M9: Tool-related problems

Figure 22 shows that survey participants, with the exception of challenge **C92**, answered most questions of category M9 as more frequently occurring (between 50 and 81%). This confirms the impression gained from the interviews that tools as a whole are often perceived as a challenge. Only 18 to 22% of the survey participants did not answer.

At the top of the ranking regarding frequency are the challenges **C87** (heterogeneous tools) and **C90** (non-continuous tool chains) with 81% and 72%. In contrast, the use of heterogeneous tools (**C87**) is considered to be rather uncritical (37%), as shown in Fig. 23. However, if interface problems occur during the exchange of data between heterogeneous tools (**C90**), this is evaluated as rather critical (36%). From the free text answers, it can be

Fig. 21 Criticality assessment for category M8: Quality assurance related problems

Fig. 22 Frequency assessment for category M9: Tool-related problems

seen that especially in case of problems with the tool chains the support is usually not able to help quickly. Increasingly, interface problems were mentioned, which may be caused by the tool chains used in the company. This is also due to the fact that the survey participants rated the challenge **C92** as rather critical (54%).

6.1.10 Concluding remarks on the assessment of the identified challenges

The results of the descriptive study show that the identified challenges are assessed differently in terms of their frequency of occurrence and criticality. In order to give a comprehensive overview of the assessments of all 92 challenges, we have calculated the mean values for the frequency of occurrence and criticality for each challenge and plotted them in a scatter plot (cf. Fig. 24). This reveals the differences between the assessments of challenges for each category. Table 7 shows the data associated with Fig. 24.

We are aware that for ordinal data the median should be used to measure the central tendency instead of the mean. However, the treatment of ordinal scales as interval scales is controversially discussed (Knapp 1990) but it has become common practice (Blaikie 2003). Therefore, we interpret the Likert scales as “measurement per fiat” and treat them as interval scaled in this section.

Figure 24 illustrates that the data points in the upper right quadrant have been rated as “very critical” by the survey participants and occur “very often.” Hence, challenges in this quadrant can be considered as riskier than data points in the lower left quadrant (“not critical” and “does not occur” occurring). For example, the challenge **C68** (increasing outsourcing leads to loss of knowledge) is rather critical and occurs more frequently, while challenge **C26** (spelling errors in test case specifications) occurs less frequently and is rather uncritical. Additionally, for a separate consideration of the assessments, the fifth most frequently and most critically assessed challenges are printed in bold in Table 7.

A closer look at the upper right quadrant in Fig. 24 shows that a total of 7 of the 17 challenges (41%) contained there are quality-related problems (cf. yellow triangles). 5 of these 17 challenges (29%) are process-related problems. In particular, it is interesting that

Fig. 23 Criticality assessment for category M9: Tool-related problems

Fig. 24 Distribution of challenges in terms of frequency of occurrence and criticality

challenges **C68**, **C69**, and **C70** all related to outsourcing test activities, have been rated most critical and most frequently, although it is common practice in the automotive industry to commission such activities and not do them in-house.

Furthermore, the scatter plot shows that the assessments of challenges per category are rather closer together and form clusters. For example, process-related problems (M6) are expected more in the two right quadrants (occur between “rare” and “sometimes” and between “partly critical” and “rather critical”), as well as communication-related problems (M7), quality assurance related problems (M8) and tool-related problems (M9). Content-related problems with input artifacts (M2) tend to occur less frequently and are assessed very differently in their criticality. Test case description related problems (M4) tend to occur rarely, but are assessed very differently in their criticality. Problems of no other category were rated so differently. They have the characteristic that they strongly differ in their assessment of criticality between “not critical” (e.g., **C26**, spelling errors in test cases) and “rather critical” (e.g., **C35**, missing preconditions in test cases). There are also greater differences in the frequency of occurrence regarding availability problems with input artifacts (M1). Although all problems are predominantly rated as “partly critical,” they differ in their occurrence between “very rare” and “sometimes.” Furthermore, test case specification content-related problems (M5) differ very strongly and are distributed over three quadrants. However, the similar assessments of challenges belonging to the same category (except categories M1, M4, and M5) could also be a threat that challenges of a category might have been assessed similarly, as they were shown to survey participants in groups according to the categories.

Table 7 List of challenges with assessment of frequency and criticality according to Fig. 24

#	Challenge	Freq.	Crit.
M1	Availability problems with input artifacts		
C01	Requirement specification does not exist	0.848	3.500
C02	Requirement specification does not exist in time	1.303	3.250
C03	Test plan does not exist	2.148	3.250
C04	Information and documents are distributed	3.118	2.867
C05	No access to relevant documents	1.829	3.000
M2	Content-related problems with input artifacts		
C06	Requirement specification is outdated	2.147	3.292
C07	Incorrect requirements exist	1.900	3.273
C08	Conflicting requirements exist	1.464	3.000
C09	Obsolete requirements exist	2.233	3.130
C10	Incomplete requirements exist	2.300	3.130
C11	Unintelligible requirements exist	2.281	2.852
C12	Test plan is too general	1.966	2.650
C13	Test plan is too extensive	1.355	3.000
C14	Insufficient description of the test object	1.310	3.071
C15	Undefined test end criteria	1.548	2.765
C16	Template does not fit project-specific requirements	1.935	2.667
C17	Influences due to faulty previous test cases	2.033	2.545
M3	Knowledge-related problems		
C18	Insufficient knowledge about the SUT	1.972	3.417
C19	Insufficient knowledge about available test platforms	1.031	2.727
C20	Insufficient knowledge about test platform functionalities	1.563	2.588
C21	Insufficient knowledge about testing guidelines	1.758	3.150
C22	Lack of training on test case specifications	2.516	3.333
C23	Lack of consulting/contact person	1.813	3.294
C24	Insufficient documentation of the template	1.321	3.083
M4	Test case description related problems		
C25	Translation errors in test cases exist	1.321	2.429
C26	Spelling errors in test cases exist	2.367	1.542
C27	Several authors are working on a test case specification	1.844	2.579
C28	Phrasing of test cases in prose leads to misunderstandings	1.242	2.714
C29	Use of abbreviations impairs comprehensibility	1.688	2.421
C30	Use of undefined abbreviations impair comprehensibility	1.813	2.714
C31	Abstract test case description leads to inaccuracies in test cases	1.909	3.174
C32	Phrasing of a test case is not specific to the respective target test platform	2.121	3.182
C33	Typing errors in test cases	2.313	1.704
C34	Incomplete test cases exist	2.171	3.269
C35	Preconditions of a test case are not documented	1.455	3.688
C36	Actions of a test case are not documented	0.909	3.417
C37	Expected results of a test case is not documented	1.091	3.462
C38	Test purpose of a test case is not documented	1.926	2.000

Table 7 (continued)

#	Challenge	Freq.	Crit.
C39	Prioritization for a test case is not documented	2.393	1.850
C40	Model series is not documented for a test case	1.259	2.750
C41	Test platform is not documented for a test case	1.367	2.571
C42	Origin of a test case cannot be traced	2.069	2.333
C43	Procedure used to determine the test case is not documented	2.263	2.000
C44	Test case description is incomprehensible	2.242	3.423
C45	Sequence of test case actions to be executed is unclear	1.117	2.786
C46	Ambiguities in test cases exist	1.735	2.727
C47	A test case tests too many functions	1.625	2.556
C48	A test case has too many linked requirements	1.720	2.500
C49	Poor readability/comprehensibility of a test case	2.118	2.885
C50	Test case contains too many parameters	1.438	2.438
C51	Test case already contains implementation details	1.034	1.909
C52	Test cases are not reused	1.269	2.455
C53	Experience-based test cases are not reused	1.577	3.077
C54	Changes to a test case are time-consuming	2.250	2.950
C55	Test case is not consistent with requirements	1.750	3.000
M5	Test case specification content-related problems		
C56	Insufficient handling of variants in the test case specification	2.269	3.059
C57	Many different model series in one test case specification	3.000	2.500
C58	Only test cases for a specific test platform are contained	1.719	2.353
M6	Process-related problems		
C59	Guidelines are missing	2.000	3.368
C60	Templates for test reports are missing	2.067	3.118
C61	Guidelines for handling the tools/tool chains are missing	1.931	3.588
C62	A defined reference process for testing is missing	1.966	3.467
C63	Unclear interface definition to the overall process	2.933	3.500
C64	Lack of change management leads to ignorance of changes	2.879	3.414
C65	Test process does not fit the project/ system	1.742	3.353
C66	Test manager is vacant	2.385	3.625
C67	Responsibilities and roles are unclear	2.061	3.556
C68	Increasing outsourcing leads to knowledge gaps	2.742	4.045
C69	Increasing outsourcing leads to increased effort	3.067	3.870
C70	Increasing outsourcing means that test execution takes longer than in-house	2.962	4.000
M7	Communication-related problems		
C71	Communication problems with the supplier	2.655	3.640
C72	Difficult communication due to communication via representatives	2.765	3.400
C73	Communication problems due to distributed teams	2.333	3.000
C74	Reduced communication due to spatial distance	2.676	3.192
C75	Communication problems due to cultural differences	1.839	3.235
C76	Communication problems due to language barriers	1.794	3.263
C77	Different expectations lead to misunderstandings	2.471	3.154

Table 7 (continued)

#	Challenge	Freq.	Crit.
M8	Quality assurance related problems		
C78	Quality characteristics for a test case specification are unknown	2.655	3.300
C79	Established metrics for quality assessment are missing	2.609	3.533
C80	Requirement coverage is the only known quality metric	3.150	3.267
C81	Tools used do not generate quality reports	2.652	2.625
C82	Company guidelines for quality measurement are missing	2.391	3.231
C83	Lack of established review processes	2.800	3.389
C84	Shortage of manpower to conduct reviews	3.406	3.370
C85	Test case specification is too large for conducting a full review	2.556	3.400
C86	Lack of time to conduct reviews	3.343	3.567
M9	Tool-related problems		
C87	Heterogeneous tools are used	3.188	2.815
C88	Tools used are complex and functions are unknown	2.333	3.045
C89	Usability deficiencies	2.125	3.150
C90	Non-continuous tool chains (interface problems) exist	2.844	3.240
C91	Project-specific adaptations of the template lead to problems	2.179	3.222
C92	Deficiencies with the tool support	1.586	3.769

The entries in bold are the main categories and group the types of challenges below. To visually distinguish this grouping, the main categories are printed in bold

Overall, the assessment of the challenges shows that the challenges identified in the interview study (1) are also known to other practitioners and (2) differ in frequency and criticality. The first aspect underlines the necessity of our research and confirms our experience with the existing poor quality of test case specifications. The latter enables the prioritization of challenges that can be used to determine the order in which they should be solved. Above all, quality assurance related problems (M8) are to be mentioned here, as these are increasingly being evaluated more frequently and critically than others. In this respect, the development of reliable metrics and of suitable review processes for automotive test case specifications are emerging as future research directions.

6.2 RQ4: Differences of identified challenges between external and internal employees

In order to validate the individual challenges of a category, the survey participants were first asked whether the challenge in question occurred or not, to which they could respond with “does occur” or “does not occur” (cf. Fig. 5). Regarding the answers to these questions, we were able to identify differences between the groups regarding the occurrence of each challenge. Therefore, we used Fisher’s exact test as we described in Section 4.2. We found statistically significant differences between external and internal employees for the following challenges.

There is a significant difference in the occurrence of the challenge **C5: No access to relevant documents** between external and internal employees. Results show a statistically significant correlation with a medium effect size between external and internal employees concerning missing access to relevant documents, $p = 0.022$, $\phi = 0.420$.

Fig. 25 Frequency of answers for challenge C5: No access to relevant documents

As can be seen in Fig. 25, the frequency to which survey participants of each group rated this challenge as “does occur” and “does not occur.” In relation, external employees answered more often with “does occur” ($\%f_{ext} = 90\%$, $N_{ext} = 10$) than internal employees ($\%f_{int} = 44\%$, $N_{int} = 25$). For this reason, it is assumed that the challenge of missing access rights to relevant documents (C5) occurs more often for external employees than for internal employees. This also confirms statements by external employees from the interviews that, for example, parts of the requirement specifications are not provided to which links exist. In contrast, internal employees are sometimes unsure what may or may not be given to external employees.

Another significant difference can be observed in relation to challenge **C10: Incomplete requirements exist**. Results show a statistically significant correlation with a medium effect size between internal and external employees and the occurrence of incomplete requirements in specifications, $p = 0.014$, $\phi = -0.499$.

Figure 26 shows that internal employees answered more often with “does occur” ($\%f_{int} = 90.5\%$, $N_{int} = 21$) and external employees more less ($\%f_{ext} = 44.4\%$, $N_{ext} = 9$). This means that external employees are less likely to be confronted with incomplete requirements than internal employees. Requirement specifications that are given to suppliers for the creation or implementation of test cases, may have a better quality of requirements than internal requirement specifications. On the one hand, this may be due to the fact that internal requirements specifications often contain more innovative topics and thus usually have a lower degree of maturity. On the other hand, requirement specifications, which are part of an contract, may be examined more closely in order to reduce possible misinterpretations on the external side.

We detected a significant difference concerning challenge **C57: Many different model series**. Results show a statistically significant correlation with a medium effect size between internal and external employees and the contained number of model series in a test case specification, $p = 0.031$, $\phi = -0.403$.

Fig. 26 Frequency of answers for challenge C10: Incomplete requirements exist

As shown in Fig. 27, internal employees answered more often with “does occur” ($\%f_{int} = 87.5\%$, $N_{int} = 24$) than external employees ($\%f_{ext} = 50\%$, $N_{ext} = 10$). This means that for internal employees, the challenge of several model series contained in one test case specification occurs more often than for external employees. This phenomenon can

Fig. 27 Frequency of answers for challenge C57: Many different model series

be explained by the fact that the creation of test case specifications is usually commissioned based on one model series and thus the complexity has always been reduced for external employees.

Furthermore, there is a significant difference in the occurrence of the challenge **C63: Unclear interface definition to the overall process**. Results show a statistically significant correlation with a medium effect size between internal and external employees and the occurrence of incomplete requirements in requirement specifications, $p = 0.029$, $\phi = -0.452$.

Similar to Figs. 27 and 28 shows that internal employees answered more often with “does occur” ($\%f_{int} = 90.9\%$, $N_{int} = 22$) than external employees ($\%f_{ext} = 50\%$, $N_{ext} = 8$). For internal employees, the process-related challenge that defined interfaces between individual test teams are missing occurs more often than for external employees. This also corresponds to the statements from the interviews. Sometimes test cases are executed redundantly on different test platforms and the different test platforms usually do not know anything about the execution status of the test cases. For example, it is often unnecessary to execute a test case at system level if it has already failed at component level. However, this very often happens in reality because an established approval process is missing or the interfaces to the overall process are unclear.

The last significant difference we found relates to challenge **C87: Non-continuous tool chains**. Results show a statistically significant correlation with a medium effect size between internal and external employees and problems related to the usage of heterogeneous tools, $p = 0.005$, $\phi = -0.567$.

Figure 29 shows that internal employees answered more often with “does occur” ($\%f_{int} = 91.7\%$, $N_{int} = 24$) than external employees ($\%f_{ext} = 62.5\%$, $N_{ext} = 8$). For internal employees, the challenge that there is no continuous tool chain occurs more often

Fig. 28 Frequency of answers for challenge C63: Unclear interface definition to the overall process

Fig. 29 Frequency of answers for challenge C87: Non-continuous tool chains

than for external employees. This phenomenon is not explicitly evident from the interviews. It is possible that the number of heterogeneous tools used by the suppliers surveyed is lower than those used by internal employees. However, it becomes clear from the free text answers that the tool chain used internally has larger deficits. Suppliers are usually only aware of a small proportion of this, which could explain the differences.

For all other challenges, no significant differences can be determined, which means that the null hypothesis could not be rejected. As a result, the assessment of the occurrence of all other challenges is to be considered equal for both groups and there are no differences between the two groups. In addition, no significant differences between the groups could be identified for the assessment of the criticality of the challenges. Based on the survey results, it can be stated that the differences between internal and external employees regarding the identified challenges can be considered minimal.

7 Conclusion and future work

In this article, we described the design and implementation of two studies in order to identify and describe challenges in the field of automotive test case specifications. Our research is guided by the key question of what are the challenges regarding test case specifications in automotive testing (**RQ1**). For this purpose, we conducted an exploratory case study based on qualitative data collection by means of 17 semi-structured interviews. We interviewed 14 employees of Mercedes-Benz Cars Development and three employees from three different suppliers and thus systematically identified challenges concerning automotive test case specifications. In particular, we considered aspects related to the areas of (C) creating (usually responsible by the test designer), (P) further processing (usually responsible by the

tester), and (Q) quality assessment (ideally responsible by an independent reviewer person who is not the author) of a test case specification.

We identified various real-world challenges and classified them in a taxonomy consisting of 28 challenge types (ToCs) and the following nine main categories: (M1) availability problems with input artifacts, (M2) content-related problems with input artifacts, knowledge related problems, (M3) lack of knowledge, (M4) the test case description, (M5) the test case specification content, (M6) processes, (M7) communication, (M8) quality assurance, and (M9) tools. We assigned these categories to the expected problem areas (C, P, and Q). In addition, the exploratory case study examined to what extent the interview participants are aware of causes, consequences, or even suggestions for solutions (**RQ2**). If they were present and named by the interviewees, they were presented in relation to the identified challenges. Some examples for practical solutions are: standard test plans (ToC-1.1), enriching specifications with additional information (ToC-1.2), project-specific modifications (e.g., of the test case specification template, ToC-2.3) or guidelines (ToC-6.1), additional documentation for the reuse of experience-based test cases (ToC-4.3), workshops (ToC-3.1), or performing walkthroughs instead of comprehensive reviews (ToC-8.3).

In our second study, we focused on assessing the identified challenges (**RQ3**) in terms of occurrence, frequency of occurrence, and criticality regarding the successful completion of the project. For this purpose, we conducted a descriptive survey using a questionnaire. A total of 36 survey participants, including 26 internal employees from Mercedes-Benz Cars Development and 10 employees from three suppliers, completed the questionnaire. The results show that in particular process-related problems (M6), communication-related problems (M7), tool-related problems (M9), and especially quality assurance related problems (M8) are assessed as more frequently occurring and more critical by the survey participants. The assessed criticality of challenges with respect to test case description related problems (M4) differs greatly. In particular, the high criticality and frequency of challenges regarding the outsourcing of development activities have to be emphasized, although outsourcing is common practice in the automotive domain. In addition, the results regarding the lack of an established review process and appropriate metrics for test case specifications point to necessary research directions.

In addition, we attempted to identify differences between employees of the OEM and the suppliers (**RQ4**). In order to detect significant differences between the groups, we used Fisher's exact test. In the end, the groups only differ in five challenges, which are as follows: (C5) No access to relevant documents, (C10) incomplete requirements exist, (C57) many different model series, (C63) unclear interface definition to the overall process, and (C87) non-continuous tool chains. Survey participants who belong to the supplier rated the challenge C5 more often with "does occur" than employees of the OEM. All other challenges seem to be OEM specific problems, because they were rated by the suppliers rather as "does not occur."

On the one hand, we identified well-known challenges in the automotive domain (e.g., complex systems and processes, problems with tools) (Broy 2006; Grimm 2003; Pretschner et al. 2007; Kasoju et al. 2013; Sundmark et al. 2011) also for test case specifications. On the other hand, we identified new challenges specific to test case specifications, such as test case description related problems. We are the first who report on an assessment of these challenges in terms of occurrence, frequency of occurrence, and criticality, focusing on automotive test case specifications.

In our future work, we focus on challenges that address quality aspects of test case specifications. We aim to develop measures to improve the quality of test case specifications. For this purpose, it is necessary to understand how automotive test cases are described and

to examine how the quality of these test cases can be measured, as test cases are a central part of the test case specification. Hence, we are currently working on analyzing how test designers describe test cases, e.g., what vocabulary and writing style they use and identify typical phrases. Initial results suggest that test designers unconsciously use a controlled language that makes it possible to derive templates. These templates seem to be a useful basis to support test designers in writing and documenting test cases and to enable reviewers to check test case specifications more quickly. Therefore, we are working on a methodology to derive system-specific templates for automotive test cases.

Furthermore, the external validity of the taxonomy of challenges can be further improved if it is confirmed by further OEMs. In addition, we see potential in further studies to explore the interrelation between the challenges identified. While our study was rather wide-ranging and included the viewpoints of various practitioners responsible for different systems, it would be interesting to look at individual systems. This would allow the evaluation of individual challenges along the software development cycle.

Funding Information Open Access funding provided by Projekt DEAL.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Bertolino, A. (2007). Software testing research: achievements, challenges, dreams. In *Proceedings of the Future of Software Engineering (FOSE '07)* (pp. 85–103).
- Blaikie, N. (2003). Analyzing quantitative data: from description to explanation. Sage Publishing.
- Bringmann, E., & Krämer, A. (2008). Model-based testing of automotive systems. In *Proceedings of the 1st International Conference on Software Testing, Verification, and Validation (ICST '08)* (pp. 485–493).
- Broy, M. (2006). Challenges in automotive software engineering. In *Proceedings of the 28th International Conference on Software Engineering (ICSE '06)* (pp. 33–42).
- Cohen, J. (1988). Statistical power analysis for the behavioral sciences. 2nd ed. Lawrence Erlbaum Associates.
- Dresch, A., Lacerda, D.P., Antunes, J., Antônio, V. (2015). Design science research. Springer International Publishing.
- Dreyfus, S.E., & Dreyfus, H.L. (1980). A five-stage model of the mental activities involved in directed skill acquisition Operations Research Center. University of California, Berkeley.
- Fisher, R.A. (1992). On the interpretation of χ^2 from contingency tables, and the calculation of p. *Journal of the Royal Statistical Society*, 85(1), 87–94.
- Garousi, V., Felderer, M., Kuhrmann, M., Herkiloğlu, K. (2017). What industry wants from academia in software testing?: hearing practitioners' opinions. In *Proceedings of the 21st International Conference on Evaluation and Assessment in Software Engineering (EASE '17)* (pp. 65–69).
- Grimm, K. (2003). Software technology in an automotive company: major challenges. In *Proceedings of the 25th International Conference on Software Engineering (ICSE '03)* (pp. 498–503).
- ISO 26262 (2011). Road vehicles – functional safety.
- ISO 29119 (2013). Software and systems engineering – software testing.
- ISTQB (2015). Standard glossary of terms used in software testing Online available: <http://glossar.german-testing-board.info>.
- Juhnke, K., Tichy, M., Houdek, F. (2018). Challenges with automotive test case specifications. In *Poster Track of the 40th International Conference on Software Engineering (ICSE SEIP '18)* (pp. 131–132).

- Juhnke, K., Tichy, M., Houdek, F. (2018). Challenges concerning test case specifications in automotive software testing. In *Proceedings of the 44th Euromicro Conference on Software Engineering and Advanced Applications (SEAA '18)* (pp. 33–40).
- Kasoju, A., Petersen, K., Mäntylä, M.V. (2013). Analyzing an automotive testing process with evidence-based software engineering. *Information and Software Technology*, 55(7), 1237–1259.
- Knapp, T.R. (1990). Treating ordinal scales as interval scales: an attempt to resolve the controversy. *Nursing Research*, 39(2), 121–123.
- Lachmann, R., & Schaefer, I. (2013). Herausforderungen beim Testen von Fahrerassistenzsystemen. In *11th Workshop Automotive Software Engineering (ASE)*.
- Lachmann, R., & Schaefer, I. (2014). Towards efficient and effective testing in automotive software development. *GI-Jahrestagung*, pp 2181–2192.
- Nörenberg, R., Reissing, R., Weber, J. (2010). ISO 26262 conformant verification plan. In *Informatik 2010, GI-Editon lecture notes in informatics, Ges. für Informatik* (pp. 515–520).
- Nulty, D.D. (2008). The adequacy of response rates to online and paper surveys. *Assessment & Evaluation in Higher Education*, 33(3), 301–314.
- Petrenko, A., Timo, O.N., Ramesh, S. (2015). Model-based testing of automotive software: some challenges and solutions. In *Proceedings of the 52nd Design Automation Conference (DAC '15)* (pp. 1–6).
- Pretschner, A., Broy, M., Kruger, I.H., Stauner, T. (2007). Software engineering for automotive systems: a roadmap. In *Proceedings of the 29th International Conference on Software Engineering (ICSE '07)* (pp. 55–71).
- Runeson, P., Höst, M., Rainer, A., Regnell, B. (2012). *Case study research in software engineering: guidelines and examples*, 1st ed. Wiley.
- Seaman, C.B. (1999). Qualitative methods in empirical studies of software engineering. *IEEE Transactions on Software Engineering*, 25(4), 557–572.
- Tierno, A., Santos, M.M., Arruda, B.A., da Rosa, J.N.H. (2016). Open issues for the automotive software testing. In *Proceedings of the 12th IEEE International Conference on Industry Applications (INDUSCON '16)* (pp. 1–8).
- Spillner, A., Linz, T., Schaefer, H. (2014). *Software testing foundation: a study guide for the certified tester exam*. Rocky Nook Inc.
- Strauss, A., & Corbin, J.M. (1998). *Basics of qualitative research: techniques and procedures for developing grounded theory*. 2nd ed. SAGE Publications.
- Sundmark, D., Petersen, K., Larsson, S. (2011). An exploratory case study of testing in an automotive electrical system release process. In *Proceedings of the 6th IEEE International Symposium on Industrial Embedded Systems (SIES '11)* (pp. 166–175).
- VDA QMC Working Group 13 / Automotive SIG (2016). *Automotive SPICE: process reference model process assessment model*. 3rd edn.
- Weber, J. (2009). *Automotive development processes*. Springer.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Katharina Juhnke is a Ph.D. student at Ulm University in cooperation with the Daimler AG as industrial partner. She received her M.Sc. in Computer Science at Leipzig University of Applied Sciences (HTWK Leipzig) and worked as an IT Consultant, Requirements & Usability Engineer. At present, she works at Mercedes-Benz Passenger Car Development focusing on improving the quality of test case specifications that are relevant for system and system integration testing of embedded systems. Her research interests are primarily embedded software testing, domain-specific languages, and also usability engineering and empirical research methods.

Matthias Tichy is a full professor and head of the Institute of Software Engineering and Programming Languages at Ulm University, Germany. His main research focus is on model-driven software engineering for complex technical systems, specifically in the automotive domain. His research addresses all phases of modeling language engineering, e.g., identification of needs, syntax and semantics definition, analysis approaches, and modeling tools with high user experience, complemented by empirical research methods.

Frank Houdek is manager for requirements engineering and testing at Mercedes-Benz Passenger Car Development. He received his Ph.D. at the University of Ulm. He worked in and headed various research and transfer projects with company internal customers in the passenger car and commercial vehicles business units of Daimler AG. He is a founding member of IREB (International Requirements Engineering Board), member of the IREB council, and head of the IREB exam group.

Affiliations

Katharina Juhnke^{1,2} · Matthias Tichy² · Frank Houdek¹

Matthias Tichy
matthias.tichy@uni-ulm.de

Frank Houdek
frank.houdek@daimler.com

¹ Group Research, MBC Development, Daimler AG, Ulm, Germany

² Institute of Software Engineering and Programming Languages, Ulm University, Ulm, Germany