An Exploratory Case Study of Testing in an Automotive Electrical System Release Process

Daniel Sundmark
MRTC, Mälardalen University
Västerås, Sweden
daniel.sundmark@mdh.se

Kai Petersen
Blekinge Institute of Technology
Karlskrona, Sweden
kai.petersen@bth.se

Stig Larsson
MRTC, Mälardalen University
Västerås, Sweden
stig.larsson@mdh.se

Abstract—The release process is a crucial element in the development of software-intensive systems, as it bridges the gap between the development of a system and its operational use. A short release process enables a fast time to market, but also puts high demands on the efficiency of integration and testing, which typically constitue principal release process steps. This paper reports findings from an exploratory industrial case study focusing on system testing in an automotive electrical system release process. We provide a description of how system testing is performed and integrated in the release process in the automotive domain, and identify a set of challenges observed in the studied setting. The case being studied is Scania, a major Swedish automotive company.

I. Introduction

In system development, a *release process* dictates the steps that are to be undertaken within the development organization to make sure that a well-packaged high-quality system is delivered timely to the awaiting customer. As such, the release process bridges the gap between the development of a system and its operational use [20]. As the release process details the final steps before the system is delivered to the customer, there is a natural inclusion of verification and validation activities. In this study, we particularly focus on *system testing* in the release process. Beizer [4] describes system testing as follows: "*System testing is aimed at revealing bugs that cannot be attributed to components as such, to the inconsistencies between components, or to the planned interactions of components and other objects. System testing concerns issues and behaviors that can only be exposed by testing the entire integrated system or a major part of it.*"

We admittedly make use of a broader definition of system testing than Beizer does, partially including what he would refer to as integration testing. In this paper, the term *system testing* is used to denote all testing where (1) the testing is performed by individual organizational entities with an explicit focus on electrical system testing, and (2) the testing is performed on a system or sub-system that is the result of the integration of several different sub-systems or modules.

With regard to system testing in the context of the automotive domain, with a tight hardware-software connection, and their specifics in the testing process, little evidence on how the testing is performed in this context exists, and the challenges in system testing in the automotive domain have not been evaluated [6]. Accordingly, the objectives of the study are (1) to describe how system testing is performed in an automotive domain company, and how it connects to a frequent release cycle of three months, and (2) to identify challenges related to system testing with frequent releases in the context of the automotive domain.

The research method used was an exploratory case study [25]. The data collected is based on open interviews and the study of documentation. In total 16 open-ended and non-structured interviews have been performed.

The paper is structured as follows: Section II discusses related work and Section III describes the research method used in the study. Section IV provides the study results, Section V discusses the theoretical and practical implications of these results, and Section VI concludes the paper.

II. Related Work

A review of research in test, verification, and validation in the automotive and vehicular domains reveals that contributions in this area are dominated by methods developed for the purpose of low-level model-based testing and verification (see e.g., [5], [13], [18]). Studies focusing on integration- and system level testing of automotive systems are sparse, and this shortage is also highlighted by other researchers [6]. However, based on the studies that do exist in this area, it is possible to derive a set of main challenges that have been reported regarding system testing of automotive systems. Challenges with respect to high level testing that were mentioned in studies with an automotive focus are summarized in Table 1, and are discussed in detail below:

C_1: First and foremost, there is reportedly a strong tradition in the automotive industry of building cars by integration of modular third-party components. This is a tradition that has also been adopted by automotive software engineering [3], [8], [16], [17]. While financially beneficial from a development cost per component-perspective (since standard automotive components can be developed once for several OEMs (i.e., vehicle manufacturers)), a consequence of this subcontracting culture is that the development process, including requirements engineering, implementation, integration and testing, needs to cross organizational borders. According to Grimm [8], having subcontractors part of the development process might significantly complicate system integration and testing, e.g., in

TABLE I
CHALLENGES IN AUTOMOTIVE SOFTWARE SYSTEM TESTING AS DESCRIBED BY RELATED WORK.

ID	Challenges Reported in Related Work
C_1	To a large extent, automotive software is subcontracted rather than developed in-house, making system verification more difficult [3], [8], [16], [17].
C_2	Early integration testing is hampered by hardware dependencies and lack of hardware availability [2], [12].
C_3	Mass-customization of vehicles calls for massive configuration and variant testing [17].
C_4	The safety-critical nature of some vehicular subsystems poses additional requirements on their verification [8], [17], including safety concerns of early system testing in a real setting [22].
C_5	A steadily increasing number of functions are distributed and require integration for testing [17].
C_6	Overlapping tests at different levels cause waste of testing time and resources [11].

the form of communication being hampered by organizational and geographic distribution.

C_2: Insights into the exponentially increasing corrective costs relative to the time of defect detection in the development process in general software engineering has emphasized the importance of early testing, even at higher levels of integration. In the case of automotive software engineering, as in most embedded software development, system integration early in the development process is often hindered by lack of target hardware access [2], [12].

C_3: Another challenge, reported by Pretschner et al. [17] relates to variant handling. As a customer, you often want the opportunity to customize your vehicle. Different alternative selections of, e.g., gearbox, engine or driver interface will also lead to corresponding selections in the electrical system configuration. This means that each individual subsystem selection needs to function properly in every possible integrated system configuration (or that the organization keeps track of which combinations that are incompatible and impossible to select). Naturally, the list of available subsystem configurations also varies over time, and with the long lifetime of automotive systems, it is required that backward compatability spans over decades. Hence, when you are testing an automotive system, you are really testing a family of systems, most often subject to combinatorial explosion.

C_4: Automotive systems are heterogeneous in nature, but nearly all such systems include subsystems with strict requirements on safety and reliability (e.g., braking and engine control) [8], [17]. Testing plays a major part in the safety assurance of these subsystems. This challenge is further complicated by the level of integration between the non-safety critical subsystems and the safety critical ones. It is of utmost importance that the non-safety critical systems are not allowed to hinder the safe operation of the safety critical ones. In addition, this safety criticality also restricts the possibility of early testing in a real setting, since such testing would potentially endanger the safety of both the driver and other individuals [22].

C_5: Related to the above, in the automotive domain, there is a trend towards more and more complex functionality, whose implementation is distributed over several previously isolated subsystems [17]. Examples include electronic stability control, which, in the most complex case, require interaction between the braking, engine control and transmission systems. Functions distributed over several subsystems requires performing

(sub)system integration before function testing, which in turn adds to the responsibility of integration testing.

C_6 Finally, Perez and Kaiser report on an observed overlap between different test levels in automotive systems development [11]. The authors state that *"The strict separation of test levels results in similar or even identical test cases being separately specified, implemented, and executed at different test levels."*.

III. RESEARCH METHOD

The research method used is an exploratory case study. In contrast to descriptive case studies, exploratory case studies do not require or assume a priori formulation of hypotheses or theories. The reason for choosing an exploratory case study method was that there is little evidence reported on how automotive companies approach testing in short release cycles.

A. Research Questions

The research questions can be directly linked to the two objectives formulated in the introduction.

1) How are integration and system testing performed in the context of software development in the automotive domain, with frequent releases to the market that are tightly coupled with the release of hardware (vehicles)?
2) What challenges does the automotive domain in the described context face, and how do they compare to already known challenges in the domain?

B. Case, Context, and Unit of Analysis

The case being studied is Scania CV[1], a Swedish truck and bus manufacturer. The context of the case study can be characterized based on the checklist provided in Petersen and Wohlin [15]: Scania is developing for a mass market, i.e. their development is market-driven with a large set of potential customers. The products being developed are real-time and safety critical systems, and are based on product-lines with a high degree of variability. The development is completely co-located at one single site. The unit of analysis is the Scania Electrical System.

The Scania Electrical System: SESAMM (i.e., *Scania Electrical System Architecture made for Modularization and Maintenance*) is the common electrical system used by all vehicles produced by Scania. SESAMM is a system of distributed Electronic Control Units (ECUs) that operate over
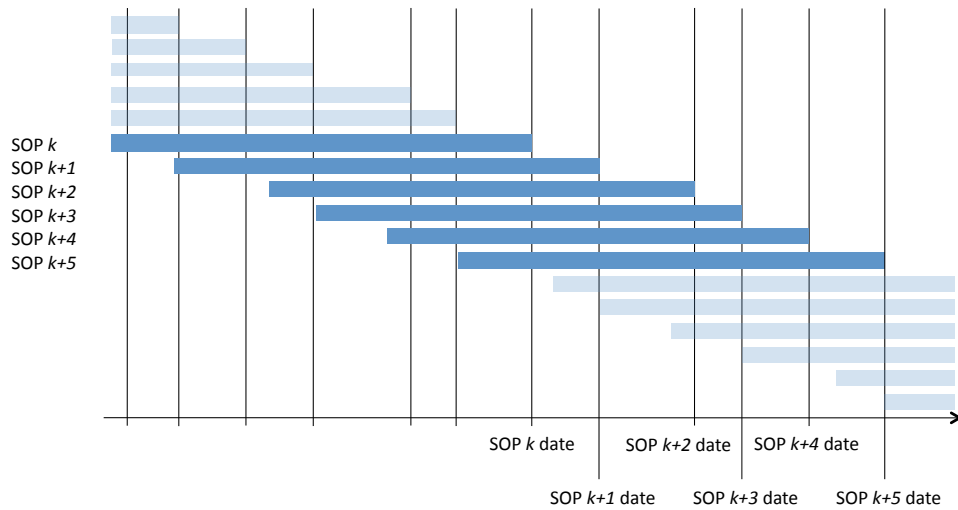
---

[1]http://www.scania.com

Fig. 1. A set of parallel SOPs over a period of time.

three different CAN buses, interconnected by a coordinator ECU system. The buses are categorized mainly based on the level of mission criticality of the ECU systems connected to each bus (or more specifically, the severity of consequences should the CAN bus communication fail). The *Green bus* connects ECU systems related to informatics, heaters, and climate control. The ECU systems on the *Yellow bus* relate to support functionality, e.g., visibility, diagnostics and instrumentation, and the *Red bus* mainly connects core functionality ECU systems, such as powertrain control and brake management. Note that we use the term *ECU system* to refer to an embedded ECU microcontroller, including hardware and software, within SESAMM. This is only partly compliant with the internal Scania terminology, which commonly uses the term *system* to refer to both individual ECUs, and to SESAMM as a whole.

ECU System and User Function Owners: Each ECU system in SESAMM is "owned" by a person or role in the Scania organization (i.e., *system owners*). ECU system ownership indicates responsibility for that particular ECU system, and its development and evolution. Whenever changes to SESAMM affect a particular ECU system, the owner of that ECU system should be involved.

At the level of abstraction above ECU systems in SESAMM lies the concept of *user functions*. While ECU systems mainly are isolated in their concern to a particular structural part of the vehicle (e.g., engine, driver interface, etc.), user functions focus on the fulfillment of (user initiated) tasks. As such, user functions are implemented through the collaborative functionality of the underlying ECU systems. In some cases, the implementation of a user function is isolated to a single ECU system, but often the implementation of user funtions requires the support of several ECU systems. As an example, the *Retarder lever activated braking* user function is triggered by the activation of retarder supported braking. The function should make sure that the brake lights are ignited, and that a notification of the retarder brake activation is shown on the

driver interface. Moreover, the user function needs to notify the master brake system of the retarder braking. In order to carry out all required activities, the retarder user function is supported by several underlying ECU systems.

Similarly to ECU systems, user functions are owned by persons or roles in the Scania organization (i.e., *user function owners*). It is not uncommon for a ECU system owner to also hold ownership for a number of user functions that are mainly implemented on that particular ECU system.

Customization: Customization is a primary matter for Scania. Basically, customers are allowed to customize their vehicles in great detail, including a high degree of freedom in the selection of several different variants of driver interface, gearbox, engine, exhaust system, etc. The definition of a particular vehicle customization is given in the form of a identifier string, which act as a vehicular "DNA string". This string greatly affects the configuration of the SESAMM electrical system, since different vehicular components require different configurations of the corresponding controlling ECU systems, and different versions of the software embedded in those ECU systems.

C. Data Collection and Analysis

Triangulation, i.e., consulting multiple sources of evidence, is an important concept in case study research [25]. There have been three major sources of data for this study:

Documentation: Company-internal documentation of the release process, including integration and system testing, as well as internal documentation of the electrical system. This documentation mainly consists of presentation material developed by subfield experts for the purpose of company-internal process communication and dissemination, but also includes internal technical reports, meeting notes, and the internal web pages.

Open-Ended Interviews: 16 informal 1-2 hour interviews with the system test group-, and ECU system test group leaders, system architects, and other specialists within the
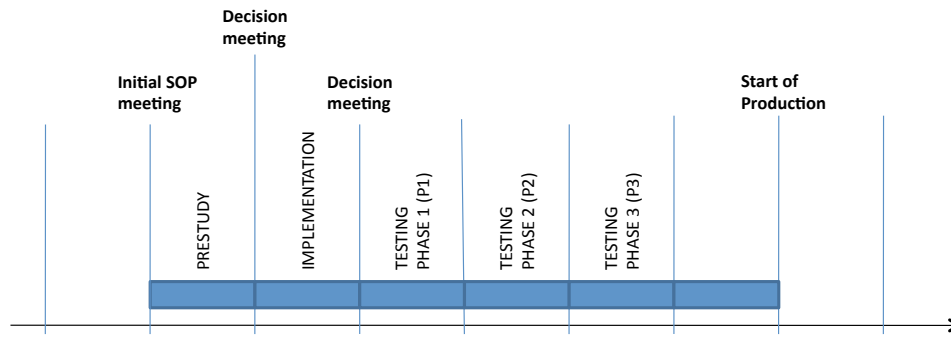
Fig. 2. The life cycle of a SOP.

organization. The interviews were non-structured, and the interview questions were generally open-ended. However, all interviewees were informed of the purpose of the interview (i.e., a mapping of system testing in the release process), and were asked to describe the role of their respective organizational unit in the release process. Moreover, frequently asked questions include questions about the internal release frequency, planning and prioritization processes, and the testing practices in the interviewee's organizational unit. Interview notes were recorded by hand.

Member Checking: All interviewees received a copy of an early version this report, and were given the opportunity to comment on any misunderstandings or inconsistencies that ware included. Five interviewees provided feedback, out of which three stated nothing but satisfaction with the contents. Two interviewees provided a number of minor suggestions for improvement. These suggestions were later incorporated in the final version.

The analysis was done in three steps. First, a description of the release process was created based on interviews and process documentation. Thereafter, challenges and desired improvements were extracted from the notes being taken. In the third step, the challenges were synthesized and their potential implications for research and practice analysed.

D. Validity threats

Four types of validity threats are generally distinguished, namely construct validity, internal validity, external validity, and reliability [24].

Construct Validity: Construct validity aims at obtaining the right measures for the concept being studied. One threat was to obtain appropriate people to answer the research questions. The threat was reduced by having support from the company in selecting people with good knowledge of the release and testing process. An open threat was that the presence of the researcher might influence the outcome of the study due to that the researcher is being perceived as external.

Internal Validity: Internal validity is primarily a concern of descriptive case studies and experiments. As this study does not seek to confirm a theory or proposition, this type of validity is not threatened.

External Validity: External validity is concerned with the generalization of the results. This is always a challenge in industrial case studies. The threat is reduced by carefully describing the context and units of analysis, so that the degree of generalizability becomes explicit. We believe the results are generalizable to a high degree within the automotive domain, as similar challenges are observed. For example, in general the automotive industry deals with the issue of customizability and short releases.

Reliability: Reliability concerns the ability to replicate the study. As the interviews were open-ended, they cannot be exactly replicated. However, the interview goals were clearly stated, and hence with these goals in mind the study should be replicable in other contexts. Another threat is that the interpretation of the researcher affects the outcome. To mitigate this risk two actions have been taken. Several sources of information were consulted, and member checking was conducted to confirm whether the information collected was interpreted correctly.

IV. RESULTS

This section presents the results of the exploratory case study, and provides answers to research question 1 (Section IV-A) and research question 2 (Section IV-A2).

A. The Release Process and System Testing

1) The Electrical System Release Process: At the topmost level of integration, a new increment of the SESAMM electrical system is released quarterly. However, internal releases of individual ECU systems may occur more frequently (or infrequently).

Start of Production A central aspect of the release process of the electrical system at Scania is the concept of Start of Production (SOP). By definition, a *SOP date* is a point in time where a certain version of the electrical system goes into production, but the term *SOP* is also used to denote the package of new and existing functionality that is scheduled to be released at the sop date. Additionally, the term "SOP" may be used to refer to the process leading up to the release of this functionality (i.e., the particular instance of the release process). Basically, any planned change to the system is defined in the form of a change request, which in turn is
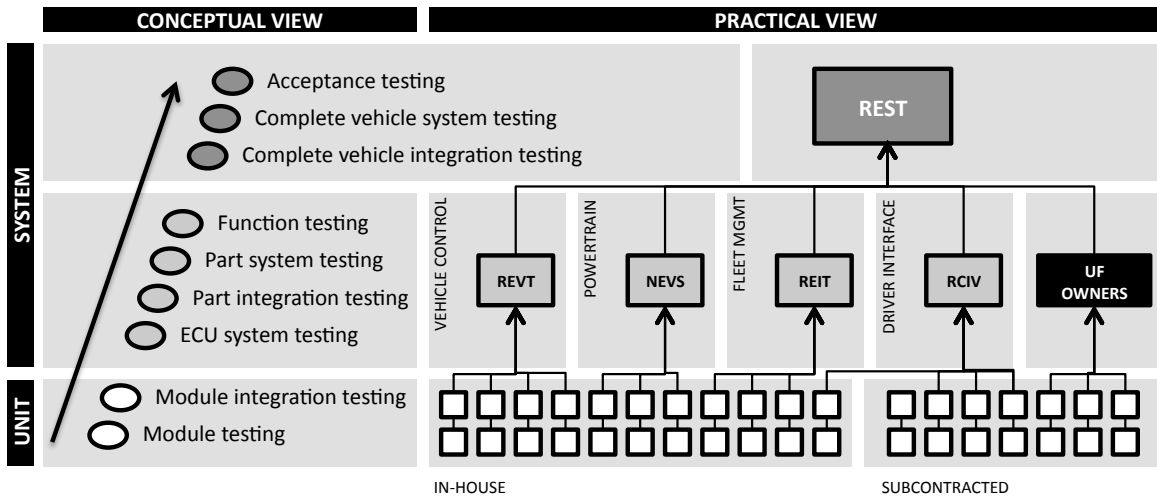
Fig. 3.   Levels of testing at Scania.

allocated to a specific SOP. At any point in time, roughly 5-6 SOPs are being developed and tested in parallel (see Figure 1 for an example).

The SOP Life Cycle The life cycle of a SOP (depicted in Figure 2) is initiated at a startup meeting. At the startup meeting, proposed change requests are evaluated, and a preliminary decision is made regarding which change requests might be eligible for inclusion in the SOP. The startup meeting also marks the starting point for the first phase of the SOP, i.e., the *prestudy* phase. The focus of the prestudy phase is the systematic identification of which functions and ECU systems that are affected by the change requests included in the SOP.

At the end of the prestudy phase, a decision meeting is held. This meeting also marks the beginning of the *implementation* phase. In the implementation phase, a first version of the functionality to be included in the SOP is developed at the different ECU system development units. The idea is that a first version of each change request in the SOP should be ready in time for the first SOP integration and system test period.

The implementation phase is followed by three iterative system-level verification phases (P1-P3), which in practice are three consecutive relese increment integrations. A change request, whose implementation has not reached a sufficient level of quality at a specific time before the SOP date is moved to a later SOP.

2) System Testing in the Release Process: At Scania, there are basically three main categories, or levels, of system testing: First, during *ECU system test*, individual ECU systems are verified and validated. At this level, the main concern is the integration of software modules within the ECU being tested, and not the correct integration with other ECU systems, even though connected ECU systems might be included in the testing in order to provide a suitable test environment. Second, above the ECU system testing level, the testing with the goal of verifying and validating user functions (i.e., *function testing*) is performed. As previously discussed, function testing may

concern user functions that are implemented solely in one ECU system, but often, user functions make use of several ECU systems in order to perform their tasks.

Third, and at the complete vehicle level of integration, *full system integration testing* is performed. The full system integration testing mainly concern issues related to regression testing (i.e., does a change in one ECU system maliciously and unintentionally affect another ECU system?), and validation testing (i.e., are the top-level requirements met for the newly implemented change requests?). An overview visualization of all test levels in is provided in Figure 3. Note that the conceptual view (left side of the figure) adheres to the organizational structure, in that ECU system test and function test are clustered together. In the following sections, the levels of system testing at Scania are described in further detail.

ECU System, Part Integration, and Part System Testing At Scania, there are four ECU System Test groups (G_1 - G_4). Each ECU system test group is responsible for the verification and validation of a subset of the SESAMM ECU systems. The activities, focus and resources of each group is described in detail below.

- G_1 is the group performing the testing of the ECU systems responsible for powertrain control. The powertrain control system is internally released with a frequency three times the release frequency of the full SESAMM system. There are currently three testing labs for engine, gearbox, and test of the integration of engine and gearbox ECU.
- G_2 is the group responsible for testing of the coordinator and body work ECU systems. Release of these ECU systems is limited to every other SOP. At G_2, testing is mainly performed in a relatively new hardware in the loop lab. Due to the fact that the lab, and the way of working with the lab, is still under development, testing is currently labour-intensive and quite time-consuming.
- G_3 is the group assigned to test the fleet management system. Basically, the fleet management system consists of
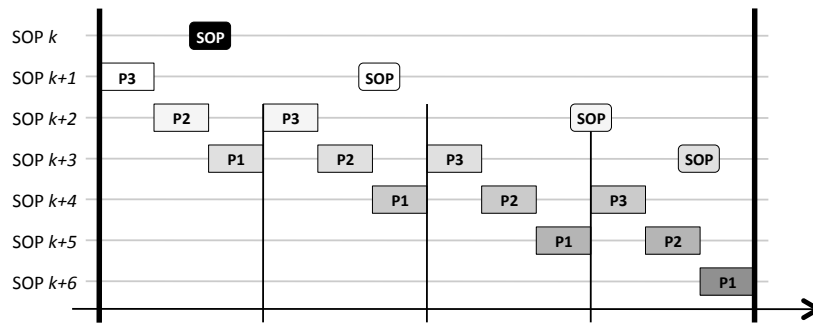
Fig. 4. The interleavings of SOP test periods (simplified).

two parts: a fleet management ECU system with the main task of providing various vehicular field information, and a standalone desktop fleet management server application with the main task of collecting and presenting the provided information from a fleet of vehicles. The fleet management ECU system is a part of SESAMM, whereas the fleet managment server application is not. Hence, only a minor part of the system testing performed at G_3 falls within the scope of this paper.

- G_4 is the group testing the driver interface ECU systems. Mainly, releases in these systems are synchronized with the SESAMM release process. However, minor in-between releases, mainly concerning bug fixes, are made when required. At G_4, testing is primarily performed using various standalone versions of the driver interface connected to a desktop computer managing testware and test cases.

- *Other ECU Systems:* ECU system testing of a minor set of SESAMM ECU systems do not fall under the responsibility to any particular ECU system testing group. These are mainly subcontracted ECU systems, where ECU system testing is performed on location at the ECU system vendor. An example of such a system is the braking ECU system. From a Scania perspective, these systems are included in the system testing process at the function testing level (see below).

Function Testing mainly concerns the validation of user functions, often testing the integrated functionality of a set of ECU systems. It is primarily driven by the user funtion owner. Even though they are conceptually different, ECU system testing and user function testing are not different testing stages or phases in practice. In most cases, function testing and ECU system testing are performed by the same ECU system test group. The reason for this is that the corresponding development group often holds ownership of both the user function, and the ECU system(s) on which the user function is mainly implemented. Also, as stated above, function testing is the first level of system testing performed for some of the subcontracted systems, where the ECU system testing is performed by the ECU system vendor.

Complete Vehicle Integration and System Testing At the topmost level of integration (i.e., including and considering

the electrical system in its entirety), testing is performed in testing labs and in actual vehicles. The organizational group responsible for full system integration testing, G_5, performs testing in three different periods in a SOP: P1, P2 and P3. The rationale behind having three test periods, separated in time, is that it is considered important to get a first early top-level integration testing, and that the results from this and the second test period are highly valuable feeback for further development and finalization of the SOP. Consequently, for the first test period (P1), all "critical" functionality of the SOP should be implemented.

Full system integration testing is the only instance of system testing that fully considers the correct versioning of all integrated ECU systems, particularly considering what should be included in the SOP to be tested. Consequently, there are two main objectives for full system integration testing: First, the aim is to perform verification and validation of newly implemented functionality at the topmost level of integration. Second, the aim is to perform a cost-efficient regression testing to provide evidence that the newly implemented functionality does not affect existing functionality in any harmful way.

Partly due to testing lab and test vehicle availability, testing is performed in pseudoparallel, i.e., the testing resources are time shared between different SOPs (see Figure 4). Whenever a SOP is in a testing phase, but not actively being tested in lab or field testing, it is being refined, and defects detected during testing are being corrected. Lab and field testing are most often synchronized. In other words, whenever a particular SOP enters a particular test period in the lab, the same SOP test period is started in the field. However, in certain situations, lab and field testing may be dislocated in time.

There are two labs for performing full system integration testing at Scania. The older lab, *Lab1*, is primarily a lab for manual testing, built by numerous versions of physically interconnected SESAMM ECU systems. The newer lab, *Lab2*, is a dSPACE[2] hardware in the loop lab that supports a higher degree of test automation.

Field Testing By the term *field testing*, we refer to testing performed in complete operational vehicles in controlled situations. Field testing at this level of integration concerns testing

---

[2]www.dspace.com

TABLE II
CHALLENGES IN AUTOMOTIVE SOFTWARE DEVELOPMENT FOUND IN THE STUDIED SETTING.

ID	Challenges Found in the Studied Setting	Related Work
CC_1	Change requests are included late in the release process.	
CC_2	Field quality issues do not always pass through all steps of system testing.	
CC_3	Misalignment between dependent and concurrent processes.	C_6
CC_4	Difficulties of assessing the quality assurance value of field testing.	
CC_5	Difficulties of assessing the quality assurance value of entire test periods.	
CC_6	Difficulties in defining the exact responsibilities of different test levels.	C_6
CC_7	Differences of opinion regarding test effort distribution.	
CC_8	Waste of effort in test infrastructure reconfiguration.	C_4
CC_9	Lack of legacy test cases.	
CC_{10}	Complexity increase caused by distribution of functionality.	C_5
CC_{11}	Lack of measurement support.	
CC_{12}	Mass customization results in combinatorial explosion in testing.	C_3

of the whole vehicle, including mechanical parts, and is not limited to electrical system testing. There are a certain number of principles being applied to field testing. For example, it is desireable that the functionality of each new release should run a certain number of hours in field operation. Furthermore, each release should span over at least one specialized field test in winter and summer conditions.

Even though field testing is conceptually visible in the release process only at the complete vehicle level of integration testing, the field testing fleet, or parts thereof, is rather used as a generic verification and validation resource throughout the development and release processes. Also individual ECU subsystem developers might test new implementations directly by integration of prototype ECU system into the the electrical system of field test vehicles (even if this requires a special dispensation, cf. challenge C_4). Several interviewees note that Scania's basic principles ascribe field testing, at all organizational levels, an inherent value.

Lab and Test Vehicle Reconfiguration The test labs and field test vehichles need to be reconfigured based on the nature of the change requests of a particular SOP, and which functions and ECU systems they primarily affect. These reconfigurations can be seen as context switches between testing of different SOPs, using the same labs and vehicles. Reconfiguration of the labs and vehicles must be performed each time SOP goes into a test period. Hence, three test periods in a SOP calls for three rounds of lab and vehicle reconfiguration.

B. Company Challenges

During the course of collecting information regarding the current state of system testing in the release process, personal opinions and notable discrepancies have inevitably emerged. This section reports on challenges in the studied setting, based on observations made during the interviews and during the review of in-house documentation. Table 2 provides an overview of the identified company challenges (CC_1 to CC_{12}). Challenges CC_1 to CC_3 are caused by discrepancies between the intended (see Section IV-A) and the actual process. Challenges CC_4 and CC_5 are special in the sense that there are differences in opinion regarding these challenges between practitioners. Challenges CC_6 to CC_{10} have been brought up by interviewees, while CC_{11} and CC_{12} are based

on observations made by the authors.

CC_1: According to the defined ways of working, all nontrivial changes to the system that are implemented in a SOP should be defined in the form of change requests or in the early phases of a SOP. In addition, all main change requests of a SOP should be implemented in a first version before the first system test period (P1). However, for, e.g., strategic and practical reasons, it is quite common for change requests to be implemented and even included in a SOP after P1. Some development groups rarely deliver at all to P1, but focus on P2 as the first delivery date of the SOP.

CC_2: Related to the CC_1, field quality requests, which basically are change requests related to corrective maintenance, are often highly prioritized, and may be introduced at any point in time, independent of the release process. In a testing perspective, the main difference between late change requests and field quality requestss would be that a change request always need to be formally assigned to a SOP, even if it is introduced significantly later than what is customary. As such, it will be taken into consideration during all following stages of system testing of that particular SOP. Field quality requests, on the other hand, are not always assigned to a particular SOP, due to the priority and urgency of solving customer-related matters. Consequently, changes stemming from such requests are not always tested in the steps outlined by the release process, and their verification and validation are potentially more ad-hoc.

CC_3: On top of the SOP-based electronic system release process at Scania exist a master product development process (the PD process). This study has knowingly omitted the PD process from detailed investigation, since it was considered to be outside the scope of the study. However, in some cases, the PD process inevitably affect the SESAMM release process in unfortunate ways. An example of such a situation is the *approval for serial production*, which must be granted a certain amount of time before the SOP date. For certain types of ECU systems, particularly subcontracted ECU systems, this approval deadline is quite problematic, since it requires a finalized version of the system before the last full system integration test period (P3). On an abstract level, this is related to challenge C_6.

CC_4: There are significant differences in the perceived value

of field testing among the interviewees, particularly in terms of defect detection effectiveness. Some interviwees consider field testing highly valuable in terms of effectiveness (i.e., field testing provides a significant contribution in finding important defects), whereas others have difficulties recollecting ever finding a bug in field testing.

CC_5: The fact that a significant percentage of change requests de facto are introduced and implemented later than intended in the release process, i.e., after P1 (see CC_1), has led a number of interviewees to question the value of keeping all three system test periods (P1-P3). Some suggest a more adaptive process, where P1 or P1 *and* P2 are omitted from the process entirely, and compensating this with having more frequent releases of SESAMM (i.e., if a change request does not reach a sufficient level of quality in one SOP, there is no need to wait for the next release). Other interviewees strongly question the removal of any system test period, stating that the early test periods contribute significantly to quality assurance. One interviewee mention the necessity of visualizing and acknowledging the actual way of working in the organization (i.e., that many change requests are not delivered before P3, rather than the ideal view that all change requests of a SOP are implemented at P1).

CC_6: Several interviewees mention the challenge of clearly and uniformly defining the boundaries of responsibility of each test level and test group. Clear responsibility boundaries help preventing overlap and gaps between different levels of testing. Failing to properly define such boundaries could lead to situations where what is erroneously thought to be tested at an above level might be omitted from testing at the current level, and conversely, what is erroneously thought to be omitted from testing at other levels might also be tested at the current level. This observation also confirms the view reported by related work (see C_6 in Section II).

CC_7: Possibilities of a potential improvement by redistribution and refocusing of testing efforts have been raised during the interviews. Specifically, refinement of the balance between efforts spent on broad user function coverage (which is heavily emphasized in the current situation), and the efforts spent on core functionality is mentioned as a topic for future improvement work.

CC_8: As each SOP is tested three times at the topmost level of integration (P1-P3), and each testing period requires a major reconfiguration of lab and field test vehicles, a significant portion of time is spent on other tasks than testing. This reconfiguration effort is further emphasized by the multiplicity of configuration tools required for configuration of different ECU systems. If the integration labs and vehicles in fact are bottleneck resources in the release process, which could be revealed by a more detailed analysis, there might be a potential for process improvement in a more efficient use of these resources. This issue partly confirms the challenge of lack of hardware availability for system testing, reported in related work (see C_2 in Section II).

CC_9: Two test groups report on legacy subsystems with a significant codebase lacking a sufficient amount of (automated) regression tests. The percieved result is a lack of verification efficiency in testing of these subsystems. Moreover, it raises a concern for the possibility of defects related to code changes on existing functionality being detected unnecessarily late in the release process.

CC_{10}: Some interviewees point out that the fast-growing size and complexity of SESAMM, including the increased number of distributed user functions, will pose challenges in maintaining the high level of quality of the electrical system. This challenge is also reported by related work (see C_5 in Section II).

CC_{11}: The primary reflection relates to systematic measurement, particularly since Scania as a company has a strong tradition in continuous improvement based on lean principles. As an example, uniform requirement and defect tracking systems do not span over the entire development process. Rather, each part of the development organization uses their own system to keep track of issues. Moreover, the systems are mainly used to keep track of the status of current issues - not for process analysis and improvement purposes. Hence, information considered important for systematic improvement is often lacking. Also, there are differences in the terminology used as well as the type of information recorded.

CC_{12}: The mass customization philosophy adopted by Scania enables a high degree of flexibility and configurability of the electrical system, but at the same time, results in a huge configuration space that needs to be verified and validated. This observation also confirms challenge C_4 reported by related work.

V. Discussion

A. Implications

In this section we discuss potential implications of our findings for research and practice. Research implications are primarily additional research needed to strengthen or refute the findings. From a practical point of view it is of high interest to find out whether these findings hold in a larger context, and to find possible solutions to the problems arising from them.

Based on CC_1 and CC_2, it could be hypothesized that a long, fixed-time release process will lead to violations of the process for small and important requirement changes and for corrections of field quality issues. This could be further researched through additional case studies in automotive industries, and also in other domains. Longitudinal post-mortem studies of companies that have drastically reduced release cycles would be of special interest. Focus in such a study would be to investigate if there is correlation between reduced violations of defined processes when the release cycles are shortened. One aspect that we suspect can be domain specific is how to define what a long release process is. This aspect should be research through investigations of different application domains. A practical implication if the above hypothesis would hold is that if the defined process is not followed, the alterations of the system based on requirements changes will not be tested as defined in the release process. Hence, a change in the process might be necessary. Either the organization need

to have shorter release cycles, or change the way that small important requirements and corrections are handled. With the current concept of SOP, it would be difficult to shorten the release cycle. One way to overcome this would be to allow intermediate smaller releases for specific types of changes. The alternative is to define a more suitable parallel process for small, important changes. The current situation, where the process is violated, leads to inconsistent performance and quality of the resulting releases. By defining how these small, important changes should be handled, a common way of working would be used. This would lead to higher quality as the verification and validation processes would be the same.

Based on CC_{10}, and C_5, it could be hypothesized that an increased distribution of functionality over subsystems will push testing later in the process. Many industrial systems also outside the automotive industry see an increasing distribution of functions over subsystems [7], [21]. A possible consequence is that a more elaborate integration process is needed, and that the testing of the functionality will be pushed late. To further investigate this, research need to be made on systems that have evolved over time with increased distribution of functionality. The focus of this research should be the time used for the integration process, and the quality level of the system over time, including after delivery. The purpose would be to discover any correlations between these factors and the level of functional distribution. The practical implication if this second hypothesis would hold is that the system problems are discovered later in the development process, and means to overcome this is needed. A proper integration strategy would be a partial solution [1], [10]. In this case it means ensuring that the development of the different subsystem focus on the same user functionality in parallel, and deliver to integration synchronized with other involved subsystems. However, as different functions may influence each other, the system testing will be pushed later in the process.

B. Additional Future Directions for Research

In addition to the research described in the section above based on our findings, we propose that the identified challenges should be further investigated. The reason is that improvements with respect to these challenges are likely to be beneficial for a wider range of companies in the automotive and embedded systems domain. A good illustrative example is challenge C_3, which has also been identified in this case study (CC_{12}). The fact that the automotive industry deals with variant rich applications is well known since the early 2000s (cf. [23]), hence solutions in supporting the handling of variants are beneficial for the overall industry. However, at the same time solutions for achieving test coverage with a reduced set of test cases are scarce. There are solution proposals (e.g. Reis et al. [19]), but these were not evaluated or used in practice. Hence, empirical studies and applications of approaches to test variant rich applications would largely benefit practitioners. With respect to the other general challenges identified we recommend to synthesize evidence with respect to approaches addressing to challenges to identify potential

solutions and research gaps, e.g. through systematic reviews used in the evidence-based software engineering paradigm [9], or through mapping studies [14]. It is important to mention that challenges identified in this study that were not mentioned in the related work might also be general. One potential reason why they were not identified in other studies is that these studies did not explicitly focus on identifying challenges. Hence, we suggest replication of this study in other automotive companies.

VI. CONCLUSIONS

This paper reported findings from an exploratory case study of system testing in the release process of the Scania automotive electrical system. We presented a detailed description of how system testing activities are integrated in the release process, and what challenges can be observed in the studied setting. Furthermore, based on the observed results, we outline some directions for future research within system testing in the automotive domain. Primarily, we believe that results relating to metrics and ways of quantifying the quality assurance value of testing activities, as well as methods and techniques for handling testing of large configuration spaces would be of high value to this type of industries.

The exploratory study resulting in the process description provided in this paper is the first step of a software process improvement effort in this area. Future work includes a more detailed identification and prioritization of areas with improvement potential, as well as identification of means to attain the potential improvement in the best possible way.

REFERENCES

[1] CMMI for Development, Version 1.3. Technical Report CMU/SEI-2010-TR-033, SEI, Pittsburgh, PA, USA, November 2010.

[2] K. Athanasas and I. Dear. Validation of Complex Vehicle Systems of Prototype Vehicles. *Vehicular Technology, IEEE Transactions on*, 53(6):1835 – 1846, nov. 2004.

[3] V. Bachmann and R. Messnarz. Improving the software development for multiple projects by applying a platform strategy for mechatronic systems. *Journal of Software Maintenance and Evolution: Research and Practice*, June 2010.

[4] B. Beizer. Software Testing Techniques (2nd ed.). New York, NY, USA, 1990. Van Nostrand Reinhold Co.

[5] J.-L. Boulanger and V. Q. Dao. Experiences from a model-based methodology for embedded electronic software in automobile. pages 1 –6, apr. 2008.

[6] E. Bringmann and A. Kramer. Model-based testing of automotive systems. pages 485 –493, apr. 2008.

[7] I. Foster, C. Kesselman, J. M. Nick, and S. Tuecke. Grid Services for Distributed System Integration. *Computer*, 35:37–46, June 2002.

[8] K. Grimm. Software Technology in an Automotive Company: Major Challenges. In *ICSE '03: Proceedings of the 25th International Conference on Software Engineering*, pages 498–503, Washington, DC, USA, 2003. IEEE Computer Society.

[9] B. A. Kitchenham, T. Dybå, and M. Jørgensen. Evidence-based software engineering. In *Proceedings of the 26th International Conference on Software Engineering (ICSE 2004)*, pages 273–281, 2004.

[10] S. Larsson, P. Myllyperkiö, F. Ekdahl, and I. Crnkovic. Software product integration: A case study-based synthesis of reference models. *Inf. Softw. Technol.*, 51:1066–1080, June 2009.

[11] A. Marrero Prez and S. Kaiser. Top-Down Reuse for Multi-level Testing. pages 150 –159, mar. 2010.

[12] A. Michailidis, U. Spieth, T. Ringler, B. Hedenetz, and S. Kowalewski. Test front loading in early stages of automotive software development based on AUTOSAR. pages 435 –440, mar. 2010.

[13] G. Park, D. Ku, S. Lee, W.-J. Won, and W. Jung. Test methods of the autosar application software components. pages 2601 –2606, aug. 2009.

[14] K. Petersen, R. Feldt, S. Mujtaba, and M. Mattsson. Systematic mapping studies in software engineering. In *Proceedings of the 12th International Conference on Evaluation and assessment in Software Engineering (EASE 2008)*, pages 1–10, 2008.

[15] K. Petersen and C. Wohlin. Context in industrial software engineering research. In *Proceedings of the Third International Symposium on Empirical Software Engineering and Measurement (ESEM 2009)*, pages 401–404, 2009.

[16] N. F. Porta. Towards a Model for Cost-Benefit-Analysis of Quality Assurance in the Automotive E/E Development. In *BiPi '08: Proceedings of the 1st international workshop on Business impact of process improvements*, pages 33–38, New York, NY, USA, 2008. ACM.

[17] A. Pretschner, M. Broy, I. H. Kruger, and T. Stauner. Software Engineering for Automotive Systems: A Roadmap. In *FOSE '07: 2007 Future of Software Engineering*, pages 55–71, Washington, DC, USA, 2007. IEEE Computer Society.

[18] A. Ray, I. Morschhaeuser, C. Ackermann, R. Cleaveland, C. Shelton, and C. Martin. Validating Automotive Control Software Using Instrumentation-Based Verification. pages 15 –25, nov. 2009.

[19] S. Reis, A. Metzger, and K. Pohl. Integration testing in software product line engineering: A model-based technique. In *Proceedings of the 10th Conference on Fundamental Approaches to Software Engineering (FASE 2007)*, pages 321–335, 2007.

[20] H. Sassenburg. A multi-disciplinary view on software release decisions. In *WISER '06: Proceedings of the 2006 international workshop on Workshop on interdisciplinary software engineering research*, pages 45–52, New York, NY, USA, 2006. ACM.

[21] R. Schantz and D. Schmidt. *Encyclopedia of Software Engineering*, chapter Middleware for distributed systems: Evolving the common structure for network-centric applications. John Wiley & Sons, Inc., New York, 2001.

[22] M. Short and M. J. Pont. Assessment of high-integrity embedded automotive control systems using hardware in the loop simulation. *J. Syst. Softw.*, 81(7):1163–1183, 2008.

[23] S. Thiel and A. Hein. Modeling and using product line variability in automotive systems. *IEEE Software*, 19(4):66–72, 2002.

[24] C. Wohlin, P. Runesson, M. Höst, M. C. Ohlsson, B. Regnell, and A. Wesslén. *Experimentation in Software Engineering – An Introduction*. Kluwer Academic Publishers, 2000.

[25] R. K. Yin. *Case Study Research Design and Methods*. Sage Publications, Beverly Hills, California, 1994.