Thesis for The Degree of Licentiate of Engineering

Model-Based Requirements Engineering in the Automotive Industry: Challenges and Opportunities

Grischa Liebel

Division of Software Engineering
Department of Computer Science & Engineering
Chalmers University of Technology and Göteborg University
Göteborg, Sweden, 2016

Model-Based Requirements Engineering in the Automotive Industry: Challenges and Opportunities

Grischa Liebel

Technical Report No 146L
ISSN 1652-876X
Department of Computer Science & Engineering
Division of Software Engineering
Chalmers University of Technology and Göteborg University
Göteborg, Sweden

This thesis has been prepared using LaTeX.
Printed by Chalmers Reproservice,
Göteborg, Sweden 2016.

“Abandon all hope, you who enter here.”
- Dante Alighieri

Abstract

Context: The automotive industry is faced with rapid increases in size and complexity of their software engineering efforts, which makes successful Requirements Engineering essential. Model-Based Engineering has been suggested as a method to handle increasing complexity on a higher level of abstraction. Using models already during Requirements Engineering could offer several benefits, as changes are quick and cheap to implement. However, due to the high level of uncertainty and abstraction from implementation, it is unclear whether models can be used in the same way during Requirements Engineering as during later project stages.

Objective: The overall aim of this PhD project is to simplify the introduction of Model-Based Requirements Engineering in an automotive environment, based on objective guidelines. These guidelines should enable engineers and decision makers to decide on important factors such as the point of time or appropriate abstraction levels for requirement models. As a first step in this direction, the contribution of this thesis is an overview of the current industrial practice of Model-Based Engineering and Requirements Engineering in the automotive industry and initial results on how automotive requirements models can be created and exploited for testing purposes.

Method: Results of this thesis are obtained using the three empirical strategies case study, controlled experiment and survey. Additionally, improvements are suggested using one study following the engineering paradigm, proposing and evaluating improvements to existing solutions.

Results and Conclusions: The thesis outlines the general feasibility of models during automotive Requirements Engineering. Findings are that Model-Based Engineering is widespread in the automotive domain and used for Requirements Engineering by some practitioners. However, several problems exist in the Requirements Engineering practices of automotive companies. As a part of these, we report problems with respect to communication and organisation structure. We show that behaviour requirements from an emission standard draft can be formalised as models and used as test oracles. Furthermore, we compare two notations for formalising behaviour of an automotive requirements specification. The results indicate that languages can be chosen based on other factors than the notation, such as tool support or experience.

Future Work: There are several directions for future work. For example, high-level requirements can be re-used as test oracles on different abstraction and testing levels. Additionally, communication in Requirements Engineering could be improved by using existing model-based requirements specifications and ownership relations between requirements and stakeholders.

Keywords

Software Engineering, Empirical Research, Requirements Engineering, Modelling, MBE, Automotive

Acknowledgment

While this PhD project has only been ongoing for a bit more than 2 years, I already managed to build up quite a list of people I would like to thank. These are:

My main supervisor Matthias Tichy, for giving me the possibility to conduct research and the freedom to throw out all plans of and expectations towards this PhD project that might have existed beforehand. My co-supervisor Jörgen Hansson, for providing me with strategic guidance whenever needed. Eric Knauss, my soon-to-be main supervisor, for taking over the supervision of a project slightly out of his scope. As a part of my academic committee, my examiner Aarne Ranta for leaving enough room and giving me the freedom to independently conduct my research. Let's continue the good work!

To all my colleagues at the Software Engineering Division, including of course all the administrative staff: thank you for providing such an excellent work environment. In particular, thank you Lucas for the occasional madness. Abdullah, for sharing the office during my first two years, including lots of interesting discussions. The 'MDSD Team': Rogardt, for enduring and welcoming my never-ending arguments about how to change his course, and Håkan, for advice on how to stay sane.

My research would not have been possible without the support from industry. In particular, I would like to express my gratitude towards Gerald Stieglbauer and Oscar Ljungkrantz for welcoming my ideas and providing me with resources and feedback.

To every professional life, there is a private counterpart, without which the professional one would be impossible. Thank you Anna, любимая, for becoming part of my life and sharing the good and the bad times. Also, for accepting my regular not-so-social personality side. Furthermore, I owe much of my success to my parents, who always encouraged me and let me make my own choices in life (e.g., to not study math).

Finally, I would like to thank the Swedish weather for tempting me so rarely to leave my office!

The research that lead to the findings presented in this thesis received partial funding from the European Union's Seventh Framework Program (FP7/2007-2013) under grant agreement No 332830 and from Vinnova under DIARIENR 2012-04304.

List of Publications

Appended publications

This thesis is based on the following publications[1]:

[A] G. Liebel, N. Marko, M. Tichy, A. Leitner, J. Hansson "Assessing the State-of-Practice of Model-Based Engineering in the Embedded Systems Domain"
ACM/IEEE 17th International Conference on Model Driven Engineering Language and Systems (MODELS 2014), Valencia, Spain, September 28 - October 3, 2014. **Best Paper Award MODELS 2014**.

[B] G. Liebel, M. Tichy, E. Knauss, O. Ljungkrantz, G. Stieglbauer "Organisation and Communication Problems in Automotive Requirements Engineering"
In submission to Requirements Engineering Journal.

[C] C. Brenner, J. Greenyer, J. Holtmann, G. Liebel, M. Tichy, G. Stieglbauer "ScenarioTools Real-Time Play-Out for Test Sequence Validation in an Automotive Case Study"
13th International Workshop on Graph Transformation and Visual Modeling Techniques (GT-VMT 2014), Grenoble, France, April 5 - 6, 2014.

[D] G. Liebel, M. Tichy "Comparing Comprehensibility of Modelling Languages for Specifying Behavioural Requirements"
First International Workshop on Human Factors in Modeling (HuFaMo 2015), Ottawa, Canada, September 28, 2015.

[1] For consistency reasons, Papers A and C have been converted to British English.

Other publications

The following publications were published during my PhD studies, or are currently in submission/under revision. However, they are not appended to this thesis, due to contents overlapping that of appended publications or contents not related to the thesis.

[a] M. Tichy, C. Krause, G. Liebel "Detecting performance bad smells for Henshin model transformations"
2nd Workshop on the Analysis of Model Transformations (AMT 2013), Miami, USA, September 29, 2013.

[b] N. Marko, G. Liebel, D. Sauter, A. Lodwich, M. Tichy, A. Leitner, J. Hansson "Model-Based Engineering for Embedded Systems in Practice"
In Research reports in software engineering and management 2014:01

[c] G. Liebel, M. Tichy, N. Marko, J. Hansson "Industrielle Praxis modellbasierter Entwicklung im Bereich eingebetteter Systeme"
Multikonferenz Software Engineering & Management (SE 2015), Dresden, Germany, March 17 - 20, 2015

[d] G. Liebel, N. Marko, M. Tichy, A. Leitner and J. Hansson "Model-Based Engineering in the Embedded Systems Domain - An Industrial Survey on the State-of-Practice"
Under revision at International Journal on Software & Systems Modeling

[e] G. Liebel, R. Heldal, J.-P. Steghöfer, M.R.V. Chaudron "Ready for Prime Time - Yes, Industrial-Grade Modelling Tools can be Used in Education"
In Research reports in software engineering and management 2015:01

[f] G. Liebel, A. Olaru, S. Rajendran, U. Ingelsson, H. Kaijser, H. Lönn, R. B. Svensson "Addressing Complexity in Automotive Embedded Systems - Selection of System Model Elements for Allocation of Requirements"
4th International Conference on Model-Driven Engineering and Software Development (MODELSWARD 2016), Rome, Italy, February 19 - 21, 2016

[g] G. Liebel, H. Burden, R. Heldal "For Free: Continuity and Change by Team Teaching"
Under review at Journal of Studies in Higher Education

[h] G. Liebel, R. Heldal, J.-P. Steghöfer "Impact of the Use of Industrial Modelling Tools on Modelling Education"
In submission to the 29th IEEE Conference on Software Engineering Education and Training (CSEE&T 2016)

[i] G. Stieglbauer, M. Tichy, G. Liebel "Verfahren zur Überprüfung der Einhaltung der Anforderungen eines standardisierten Fahrzyklus "
In registration, Patent IPC G01M 017/00

Research Contribution

I joined the work on Paper A during the early planning phases of the survey. I consequently contributed by consolidating the existing survey and research questions and introducing the final study design. The hypotheses which are used for data analysis were elicited by me. Furthermore, I took the lead of executing the survey, analysing the data and writing the final publication.

My contributions to Papers B and D are the study design, data collection, data analysis and the majority of writing. In Paper B, the remaining co-authors contributed with reviews, interview organisation and improvement suggestions.

Paper C was written in collaboration with multiple co-authors. My contribution to this publication lies in the demonstration of the applicability of the presented approach in terms of the gear shift example used in the publication. Similarly, I wrote the majority of the publication that refers to this part.

Contents

Chapter 1

Introduction

The automotive industry is faced with rapid increases in size and complexity of the software included in vehicles [1]. This increase includes not only areas such as infotainment or driver comfort, but also safety-critical vehicle functions. As a consequence, managing the scale and complexity becomes increasingly difficult, especially since demands on the quality of software increase.

In particular, the automotive industry is struggling with Requirements Engineering (RE) [2], which has a significant influence on the success of software projects [3, 4]. Improving RE has a positive influence on the overall outcome in software projects, as shown in several studies, e.g., in [5, 6]. Therefore, improving RE practices is essential in order to succeed in future development efforts in the automotive domain.

Model-Based Engineering (MBE) is an engineering approach using models to handle complexity by means of abstraction [7]. As changes are quick and cheap to implement, using models during early RE holds many potential benefits. However, due to the high level of uncertainty and abstraction from implementation, it is unclear whether modelling of requirements can be performed in the same way as modelling at later project stages. Therefore, the overall aim of this PhD project is to investigate the potential use of Model-Based Requirements Engineering (MBRE) in an automotive context. The outcome of the PhD project should be objective guidelines which enable engineers and decision makers to decide on important factors such as the point of time or appropriate abstraction levels for using models in automotive RE. As a first step in this direction, the goal of this thesis is as follows:

G1: To investigate the current use of MBE in automotive systems engineering

G2: To elicit problems that exist in automotive RE

G3: To demonstrate initial solutions to existing problems in automotive RE using models

The first two goals aim to improve the general understanding of the problem and the solution domain, namely MBE and automotive RE. These goals are needed to understand where models could bring potential benefits and where they are already used. The third goal complements the first two goals and aims to provide initial contributions in a constructive manner.

The remainder of this chapter introduces the theoretical background of the problem domain, automotive RE, and of the solution domain, MBE, in Section 1.1. Based on the background, MBRE is discussed in Section 1.2 and the scope of this thesis is presented in more detail in Section 1.3. Related work to this thesis is outlined in Section 1.4, followed by the research methodology in Section 1.5. The contribution of each individual paper is discussed in Section 1.6. A discussion of the findings is presented in Section 1.7, followed by a discussion of validity threats in Section 1.8. The introduction chapter is concluded by a summary and a discussion of future work.

1.1 Background

This thesis addresses the problem domain of RE as a part of automotive systems engineering. As a solution domain, MBE is considered. This section provides an overview over both the problem and solution domain.

1.1.1 Automotive Requirements Engineering

According to Maurer and Winner [8], automotive systems engineering is "(A) methodology for developing systems for a vehicle, or a vehicle as a system.". This development includes several disciplines, such as mechanical engineering, electrical engineering and software engineering. Vehicles are usually developed by an Original Equipment Manufacturer (OEM), in cooperation with several first-tier and second-tier suppliers.

Automotive systems engineering has specific characteristics, which distinguishes it from other areas in systems engineering [8]. First, vehicles are used under greatly varying conditions, e.g., imposed by different laws in different countries, different skill levels and behaviour of drivers or variations within different cars of the same model. Secondly, demands on compatibility of subsystems are high, as components are re-used across vehicle models. In particular, this means that vehicle projects rarely start from scratch but rather evolve existing specifications. Thirdly, the high degree of safety critical functions and the large production volume greatly influence the costs of errors made during development.

Overall, the distribution among disciplines and organisations, as well as the specific characteristics of automotive systems engineering makes developing vehicles highly challenging.

The automotive industry faced rapid increases in size and complexity of the software included in vehicles in the past three decades [1]. For instance, in year 2002, the Volvo XC90 automobile contained 38 Electronic Control Units (ECUs) [9]. The 2015 model already contains 108 ECUs [9], almost three times as many as in the 2002 model. Today, software is used for purposes such as engine control, infotainment and for safety-critical aspects of vehicles, such as braking or steering the vehicle. As a consequence, demands on the quality of software increase even more [1].

As a part of the overall development, the automotive industry is also struggling with RE [2]. From software projects on a general level, it is known that RE influences project outcome significantly [3]. Therefore, by improving RE, the chances of succeeding in a project can be increased. For instance, improving particular RE practices leads to fewer defects in the end product [5] or to increased productivity and improved communication during development [6].

RE is "a systematic and disciplined approach to the specification and management of requirements [..]" [10]. A requirement is "(1) A need perceived by a stakeholder. (2) A capability or property that a system shall have. (3) A documented representation of a need, capability or property." [10]. The process of RE is typically broken down into requirements elicitation, requirements specification and requirements management. In requirements elicitation, requirements are sought, captured and consolidated [10]. This includes to address sufficiently the goals of multiple stakeholders, which can be conflicting. In requirements specification, the requirements are specified in a systematic fashion [10]. The resulting document is also called a requirements specification and separate documents can exist on the customer side (Customer Requirements Specification) and on the supplier side (System Requirements Specification) [10]. Finally, requirements management is the process of managing existing requirements specifications. In particular, this includes making changes to requirements and tracing later development artefacts, such as code and tests, to requirements. In this thesis, we mainly consider requirements specification and management.

The overall requirements specification should meet several quality criteria to be considered a 'good' specification, according to IEEE Std. 830-1998 [11]. These contain (definitions according to [12]), e.g., correctness, to meet a customer's need or expectation; consistency, to not contain any conflicting requirements; or verifiability, that there exists a feasible way to check that the product meets the requirements. These criteria are, in practice, difficult to achieve.

Overall, the specific characteristics and rapidly increasing complexity of software in automotive systems engineering, together with the importance of RE are the reason for selecting automotive RE as a problem domain in this thesis. In particular, goal G2 aims at understanding specific problems in automotive RE and goal G3 aims at suggesting initial solutions to existing problems.

1.1.2 Model-Based Engineering

Models are central elements in many engineering disciplines, as well as in science. They can help to explain complex concepts in a simplified way, by excluding information that is not relevant for the explanation [13].

We use the following definition for a model, based on Stachowiak's features of a model [14]: *A model is a representation of entities and relationships in the real world with a certain correspondence for a certain purpose.* Therefore, when talking about a model, it is essential to answer what the model represents (the *model object*) and why it exists (the *model purpose*).

MBE and Model-Driven Engineering (MDE) are engineering approaches that have been devised to handle complexity and increase efficiency in the development of software or engineering in general [7]. Both employ models to handle complexity by means of abstraction. Additionally, both approaches are used in industry [15–17] and several empirical studies show benefits of MBE, e.g., increased productivty [16] or improved quality [18]. Therefore, we consider them as a candidate solution to address the challenges in automotive RE.

We use the definitions of Brambilla et al. [13], in which MBE comprises approaches were models play an important role but not necessarily the primary role. MDE has a narrower scope and is an approach in which models are used as the primary artefacts throughout the entire engineering process [13]. In the literature, many similar abbreviations are used to describe approaches that differ only slightly from MBE and MDE. For example, Model-Driven Development (MDD) can be seen as a subset of MDE, only focusing on development. Finally, Model-Driven Architecture (MDA) is a specific version of MDD devised by the Object Management Group [19] and therefore a subset of MDD. The overlap between the different approaches is visualised in Figure 1.1.

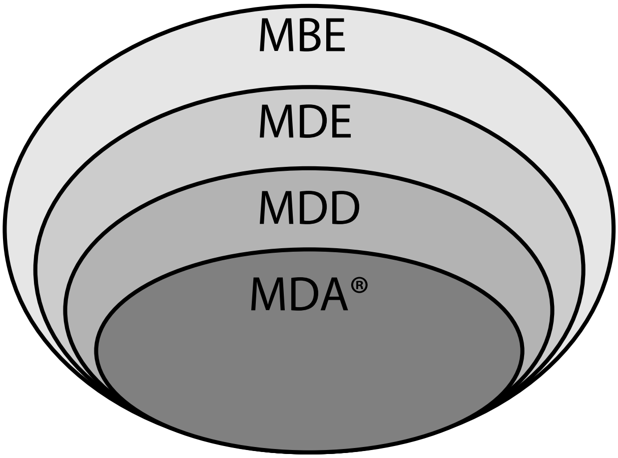

Figure 1.1: Classification of Model-Driven and Model-Based Approaches, Adapted from [13]

Apart from models, transformations play a key role in MBE and MDE. For example, model transformations are considered to be “a key part of MDA” [20] and to be “among the most important operations applied to models” [21]. Their task is to transform models into different artefacts, e.g., into other models or into software code. These transformation steps could be automated by using special transformation languages, e.g., following the Query/View/Transformation standard [22].

Ideally, transformations are automated in order to require as little as possible manual work in between the transformation steps. If this is the case, tracing between the models exist and complexity is reduced, as less manual work is needed to create and maintain artefacts throughout the entire engineering process.

As to which extent model transformations are used in industry and whether it is in practice possible and feasible to have an automated chain all the way to source code, or any other low-level artefact, is not answered by related academic work. This vision is however not very realistic given the abstraction gaps in between artefacts in the development process. For example, high-level requirements do typically not contain any information on how the architecture of a software system should be structured. Therefore, domain knowledge is needed to add this information while constructing a model of the software architecture.

The terminology regarding model-based or model-driven approaches is not used consistently in academic literature. For example, the definition itself does not clearly state what it means that models are primary artefacts. Furthermore, some researchers regard only approaches which contain model transformations to be model-driven. Therefore, we position ourselves broadly in this thesis and use the most general term, MBE, as a possible solution domain to challenges in automotive RE. In particular, goal G1 aims at understanding how MBE is already used in the automotive industry and goal G3 aims at addressing existing problems in automotive RE using MBE. We do not regard model transformations as mandatory in MBE.

1.2 Model-Based Requirements Engineering

While the overall demands for developing automotive systems increase, requirements in particular are a major cost driver in all types of embedded systems [1]. Therefore, special attention towards the process of RE is warranted.

The documented benefits of using MBE in industry, e.g. in [15, 16, 18], indicate that MBE can be used to address these increased demands in industry. As a major cost driver, MBE should not be restricted to software design and development only, but be already used during RE in the automotive domain. In this thesis, we refer to this approach as automotive MBRE. Figure 1.2 visualises this relationship between the problem domain of automotive RE as a subdomain of automotive systems engineering and the solution domain of MBE.

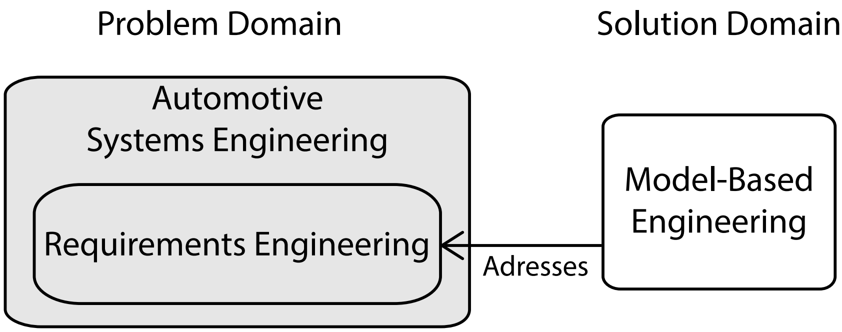

Figure 1.2: Model-Based Requirements Engineering

Based on the definitions of RE and MBE and the different aspects of automotive systems engineering, the 'Addresses' arrow in Figure 1.2 could be refined in several different ways. In particular, modelling in RE has a long history and, therefore, some kind of classification scheme or taxonomy is needed to discuss our contributions in comparison to existing work. For this purpose, we use the classification model depicted in Figure 1.3, using the feature model notation by Kang et al. [23]. In this figure, a feature can be seen as one aspect of model usage. We call one instance of this model a *configuration*. Some of the model's aspects can only have pre-defined values, i.e., *grade of* *formality* and *completeness*. Other aspects can have arbitrary values, or at least a non-exhaustive list of them, e.g., the *purpose*.

This classification is based on our current understanding. As of now, it has not been validated and serves therefore only as a means to describe the variety of how MBE can be used to address automotive RE (or other problem domains) and the contribution of this thesis. Therefore, we do not discuss here if all aspects are required (mandatory) and what their multiplicities are.

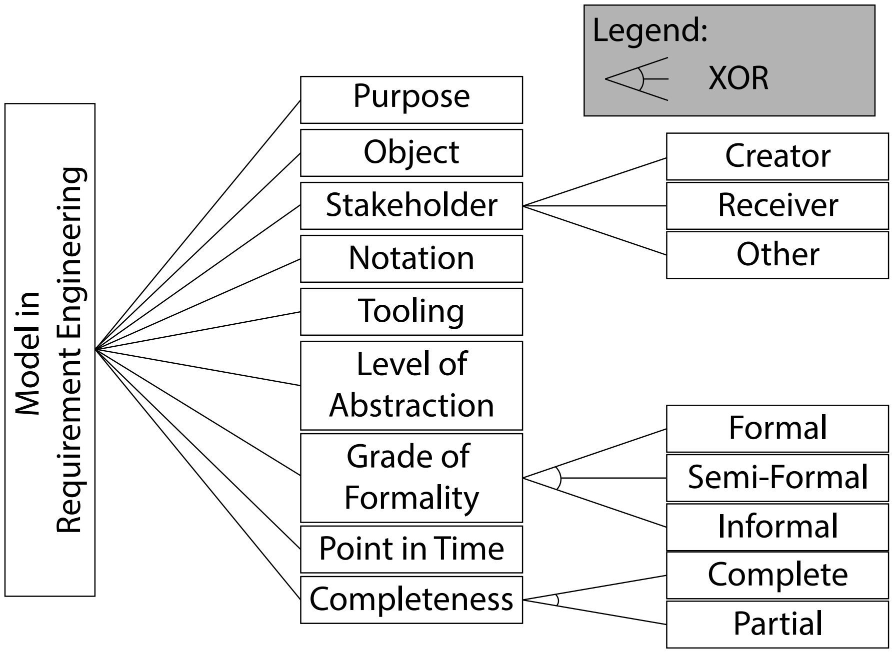

Figure 1.3: Different Aspects of using MBE for a Given Problem Domain

According to the definition, every model has a *purpose*. In MBRE, the model purpose could be, e.g., to increase the domain understanding, to document requirements in form of a graphical model, to aid elicitation or to enable requirements validation.

Furthermore, each model represents a part of the real-world, which we here call the model *object*. For example, a model could represent a functional requirement, an entire requirements specification, the relation between multiple stakeholders used for elicitation or the structure of the requirements specification and traces between single requirements. Whether functional requirements or non-functional requirements (quality requirements) are described is also dictated by the model *object*. Due to the rather important distinction of these two terms, they could however also be seen as a separate aspect in this classification.

Similarly to requirements and requirements specifications, each model in MBRE has a number of *stakeholders*. These can be broken down further into the *creators* of the model, the *receivers* of the model and *others*. Creators need to have the technical and domain knowledge to create the models in a way that is semantically and syntactically correct and that represents the domain in a correct way. Receivers need to be able to understand the model. In RE, receivers are often no modelling experts. Especially when models are intended for documentation and comprehension purposes, this needs to be taken into account when devising modelling approaches for RE. Further stakeholders (*others*) can be considered in the RE sense, i.e., any person who has a stake in the model.

Models can have different *notations*, which can in turn be both graphical or textual. Candidates are common modelling languages, such as UML [24] or i* [25]. The notation could further be divided into standard and non-standard notations. However, this aspect is not relevant for the contents of this thesis and therefore omitted.

The *tooling* used for creating, editing and viewing models plays an important, and in practice often deciding, role. Tools influence which notations can be used and for what purposes the models can be employed.

The *level of abstraction* on which the models are created is especially relevant in RE. For example, a model of requirements can follow common requirements abstraction levels, such as the Requirements Abstraction Model [26] or Lauesen's classification into goal-level, domain-level, product-level and design-level requirements [12]. Additionally, a model could be intentionally simplified, e.g., to be more understandable to non-experts.

The *grade of formality* varies largely between different models. A model that is aimed at exploring the domain during requirements elicitation and that should be understandable for several types of stakeholders will most likely be informal, whereas a model used for simulation or code generation will have a higher grade of formality. A model's grade of formality is typically broken down into *formal*, for defined formal syntax and semantics, *semi-formal*, when either semantics or syntax are not completely formally defined or *informal*, when both are at least partially missing.

During RE, there is a high level of uncertainty initially, with only few requirements and goals being known. During requirements elicitation, this situation changes as uncertainty is slowly decreasing. Similarly, the amount of requirements, their abstraction level and the level of detail changes. Therefore, the *point in time* at which models are created plays an important role in MBRE. For example, goal models might be commonly created in early stages of RE, whereas detailed UML models for simulation are more realistic later on.

The term *completeness* can be defined in two different ways in MBRE. On the one hand, completeness is used in RE as quality criterion to describe whether a requirements specification covers all non-trivial stakeholder expectations and needs. On the other hand, the notion of completeness is often used for models that contain all necessary information and do not conflict with constraints imposed by the meta model. Incompleteness in models can be cause by uncertainties or multiple stakeholder opinions [27].

We do not see uncertainty as a separate aspect in this classification. While it could be considered as such, we consider it a part of the model *notation*, i.e., notations that contain elements to explicitly encode uncertainty, and part of the *completeness*, i.e., by leaving out uncertain information.

The aspects depicted in Figure 1.3 are related to each other. For instance, the most obvious relationship is that of *notation* to *tooling*, as a modelling tool supports only a defined set of modelling notations (not considering portability of file formats). Similarly, *notation* and *tooling* will in practice affect the *purpose*, as both can have different restrictions on what purposes are possible. Additionally, the order in which the aspects are defined is often unclear. For example, organisations might decide to introduce a special model *notation* for a specific *purpose*, e.g., executable state machines in order to simulate system behaviour early on. However, this decision could be taken the other way around, e.g., if executable state machines are already used in the organisation, the decision to use them for simulation could be taken later. These relations between aspects are non-trivial and require further empirical investigation.

1.3 Goals and Scope

The classification of MBRE in Section 1.2 now allows us to revisit the goals of this thesis and the overall PhD project and illustrate the contribution in terms of this classification. As stated in the beginning of this chapter, the goal of the overall PhD project is to provide guidelines on using models in automotive RE. That is, the outcome of the PhD should be a set of configurations of the model in Figure 1.3 which are beneficial to the overall development. Their benefit should be supported or at least indicated by empirical data or by constructive means, i.e., by proposing improvements to existing solutions or methods.

This thesis contributes an initial part to this overall goal and addresses the following goals.

G1: To investigate the current use of MBE in automotive systems engineering

G2: To elicit problems that exist in automotive RE

G3: To demonstrate initial solutions to existing problems in automotive RE using models

In terms of Figure 1.3, this means that **G1** aims at establishing a picture of the different configurations of the model which are already used in the automotive domain, but not restricted to RE only. **G2** aims at increasing the understanding of which kind of configurations of the model in Figure 1.3 could be beneficial for automotive MBRE. In particular, by understanding existing problems in RE, the *purpose*, the *object* and the *point in time* of using models in future approaches can be restricted to problem areas. **G3** aims at presenting sample configurations of the model in Figure 1.3 for automotive MBRE.

To reach the goals, we formulate the following four research questions.

RQ1: To what extent is MBE used in the automotive domain and how is it perceived?

RQ2: Which problems exist in automotive RE?

RQ3: How can models of behavioural requirements be used for testing purposes?

RQ4: How do modelling languages for expressing behavioural requirements compare with respect to comprehensibility?

Research questions **RQ1** and **RQ2** aim at establishing a comprehensive picture of the current state of practice in the problem domain, which can guide future research, not only as a part of this PhD project. Specifically, by answering **RQ1**, shortcomings in the solution domain can be identified and future efforts of using MBRE directed accordingly. This is connected to the aim of establishing a picture of the different configurations of MBE which are already used in the automotive domain (**G1**). **RQ2** is instead aiming at identifying specific issues in the problem domain that can be addressed in the future and is connected to the aim of identifying suitable configurations of the model in Figure 1.3, i.e., **G2**. Research questions **RQ3** and **RQ4** aim at presenting first results in using models during RE and are connected to **G3**.

1.4 Related Work

In the scope of this thesis, each of the three goals has its own related work. The related work to goals G1 and G2 is discussed in Papers A and B. Here, only a short summary is given.

Goal G1 targets the use of MBE in the embedded industry. Empirical studies on this topic are limited [16] and we are only aware of two studies focusing explicitly on MBE in embedded systems, i.e., [18, 28]. Agner et al. survey the Brazilian embedded industry [18], reporting that MBE increases productivity and improves quality, maintenance and portability. Additionally, the authors report that MBE is mainly used for documentation, with only little use of code generation or model-centric approaches. From a case study within the automotive domain, Kirstan and Zimmermann report positive effects of MBE, such as an earlier detection of errors or cost savings during initial development phases [28]. As shortcomings, they report tool interoperability.

Goal G2 targets problems in automotive RE. Existing publications on this topic often lack empirical support, e.g., [2, 29]. We are aware of two studies reporting specific problems in automotive RE [30, 31]. Almefelt et al. report that requirements are often incomplete or conflicting and that it is difficult to overview specifications due to their size [31]. Pernstål et al. report that requirements are often unclear in early phases of a project and that it is difficult to communicate requirements to suppliers.

Goal G3 is broader and targets modelling in automotive RE in general. Therefore, the related work to G3 is not exhaustively covered in Papers C and D.

Models play an important role in the requirements engineering community. For example, many of the accepted papers at the Requirements Engineering (RE) conference series, the premier academic conference in RE, discuss models [32]. As the classification in Section 1.2 suggests, the variety of using models for or during RE is vast. In particular, common models named in these papers are goal models or meta models [32].

Restricting the scope to automotive RE only, models are often seen as a hope to cope with future challenges, e.g., in [29, 33]. A common approach is to use structural models, such as EAST-ADL [34] or the SysML requirements diagram extensions [35], to enforce the structure of a requirements specification. For example, Boulanger and Văn describe a methodology to develop embedded automotive systems, using EAST-ADL and SysML for requirements modelling [36]. Similarly, Piques and Andrianarison report industrial experiences with using SysML in the automotive domain [37]. Albinet et al. introduce a similar approach, but also use the UML profile MARTE for real-time systems [38]. All three approaches have in common that the requirements themselves are expressed in natural language, whereas the model *object* is the specification structure.

With respect to industrial practice, the extent to which models are used in RE is not as clear. Lubars et al. report that Entity-Relationship diagrams and object-oriented models are common in RE during the early 90s [39]. However, the authors do not report how these models are used later on. The popularity of models could partly stem from the widespread use of the Rational Unified Process and Structured Analysis at that time, which both include the use of models during RE. However, a more recent study by Sikora et al. reports that practitioners in the embedded industry advocate a more intensive use of models during RE [40]. The authors attribute this to the automation possibilities that RE models could offer. Given the amount of suggested modelling approaches and notations, it is surprising that the embedded industry has not adopted these sufficiently.

With respect to the goal of the overall PhD, to provide guidelines for modelling during automotive RE, we are not aware of directly related work. However, there are several publications that provide guidelines for single modelling notations or domains, e.g., for UML Use Cases [41] or for business process modelling [42]. Additionally, guidelines are often proposed as a part of a methodology or process. For example, suggestions on which models should be used can be found in the Rational Unified Process [43], in literature on systems engineering, e.g., [44, 45] or in several books introducing UML, e.g., [46]. While they cover a wide spectrum of modelling languages and typically the entire software development process, these guides are often based on the authors' subjective opinion and experience and lack empirical backing. The aim of this PhD is to provide guidelines based on empirical studies in automotive RE. While the outcome will be of much narrower scope than existing processes, the focus is instead on the empirical foundation of the results.

Finally, several tool vendors offer training in methodologies which are specifically adapted to their tools, e.g., PTC [47] or Vector [48]. As these methodologies are tool-specific, they are out of the scope of this thesis and future work towards the PhD.

1.5 Research Methodology

Software engineering involves human activities as a part of development process. In particular, the development of software is a creative process and we are unable to 'manufacture' software [49]. Therefore, it is difficult to evaluate many aspects of software engineering without human involvement. To acknowledge this, the research methodology used in this thesis mainly follows the empirical paradigm, in which new models are proposed and then evaluated using empirical studies [50]. We see the need for empirical studies in order to increase our understanding of the current state in automotive RE and to test hypotheses about the use of models within this area. To reach the overall goal of this PhD, to investigate the use of MBRE in an automotive context, we also see the need to propose and develop improvements to existing solutions. This corresponds to the engineering paradigm [50]. However, in order to be able to propose relevant improvements in the automotive domain, it is important to first understand the current state sufficiently well.

We use different research strategies for the three empirical studies described in Papers A, B and D. In Paper A, the data was collected using a survey. Surveys can be helpful to obtain a snapshot of the current situation and can give a broad overview of the surveyed area [49]. They are used to obtain a representative picture of a larger population [51]. This fits the aim of the study, as we want to create a broad picture of the state-of-practice in the embedded systems domain. In Paper B, we used a mixed-methods approach, collecting qualitative data in terms of multiple case study and testing the outcomes of the data analysis with a survey. A case study is appropriate when the boundaries between the studied concept and the context are not clear [49], which is the case in Paper B. In Paper D, we use a controlled experiment in order to compare two different approaches. Controlled experiments are suitable when we have a high level of control in execution in the study and in measurement [51].

Towards the aim of the PhD degree, it is also intended to demonstrate literacy in using different research methods in an appropriate manner. Therefore, additionally to reaching the goals and answering the research questions of this thesis, the variety of research strategies used in Papers A to D is also aimed at demonstrating this literacy.

A detailed discussion of the research strategies is found in the appended papers (Chapters 2 to 5).

1.6 Contribution

In the following, the four papers on which this thesis builds are outlined shortly. The entire papers can be found in Chapters 2 to 5.

1.6.1 Paper A, State-of-Practice in MBE in embedded systems

MBE aims at increasing effectiveness of engineering and handling complexity by using models as important artefacts throughout the entire engineering process. A substantial amount of empirical studies investigate the application, benefits and drawbacks of MBE in industry. However, there is a lack of comparable studies focusing on the area of embedded systems.

Modelling standards such as MARTE [52] and the widespread use of modelling tools such as Matlab/Simulink suggest that MBE is used widely in this domain. Personal experience from cooperations with industry corroborates this view. Nevertheless, empirical data is missing.

The contribution of Paper A is to fill this gap by providing empirical data on the state of practice of MBE in the embedded systems domain. In particular, the paper lays the foundation for this thesis by providing empirical support that models are already widely used in the automotive domain, to which a large part of the survey participants (60) belong to. In terms of the classification introduced previously, the paper can be described as depicted in Figure 1.4. The

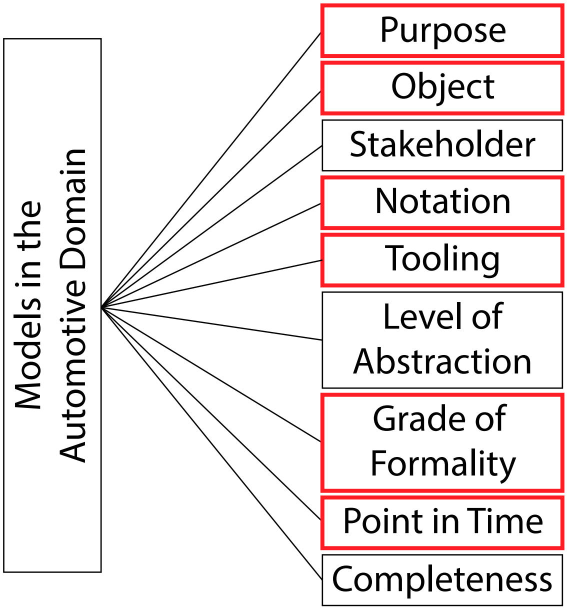

Figure 1.4: Scope of Paper A

paper does not present a defined configuration of MBRE, but rather presents data on which configurations exist in the automotive industry. In particular, it provides answers to the *purposes* for which models are created, *objects* which are described in models, used *notations*, *tooling*, *grade of formality* and the *point in time* at which the models are created. These aspects are marked red in Figure 1.4. It does so with respect to modelling in general, not restricted to RE only.

We collected data from 113 individuals, mainly professionals working in the embedded systems domain. Our findings are as follows, answering RQ1. MBE is widely used in embedded systems and leads to a reduction of defects and improvements in quality. Sequence-based models, as used in Paper C, are widely used by the participants. Models are used for several different purposes, such as simulation or test-case generation. Furthermore, 49 participants use MBE to specify requirements.

MBE was introduced at the participants' companies for a number of purposes, e.g., for increased safety, traceability, quality or to shorten development time. However, less than half state a need for formal methods.

The participants report mainly positive effects of applying MBE, e.g., on quality or reusability, but they do report increase costs. However, several shortcomings are reported as well, e.g., high efforts associated with MBE training and in order to receive benefits from MBE.

Overall, Paper A encourages us that MBE provides the necessary benefits to address current problems in automotive RE and is at the same time widely accepted in the embedded domain. To be able to focus future efforts even better, we plan to extend the work on this paper by synthesising the body of work on MBE in the embedded domain and in software engineering in general. Furthermore, we are currently analysing interview data on the use of models in automotive RE in order to obtain an in-depth picture of the domain.

The study's results are further described in a technical report [53] and were also presented in [54] as an invited paper.

1.6.2 Paper B, Organisation and communication problems in automotive RE

While Paper A explored the solution domain, i.e., the use of MBE in industry, Paper B explores the problem domain of automotive RE. Specifically, we explore which problems exist in automotive RE. These need to be understood sufficiently well before any improvements can be made.

Existing literature on automotive RE lacks empirical support, e.g., [2,29], or aims to explain how automotive RE functions in general, e.g., [30,31]. Therefore, we aim to extract a list of problems in automotive RE. In particular, the two aspects of communication and the organisation structure play a major role in the complex and distributed automotive development process. Based on this aim, the paper answers the following two research questions:

- **RQ1**: What are current problems or challenges in automotive RE with respect to organisation structure and communication?

- **RQ2**: How can these problems or challenges be addressed in the future?

We refer to the organisation structure as the logical relations or the "decision rule connections" between people in an organisation [55]. Communication refers to the exchange of information between individuals in an organisation or between organisations, not necessarily following the organisation structure.

In terms of the classification introduced previously, the paper can be described as depicted in Figure 1.5. Similarly to Paper A, the paper does not present a defined configuration of MBRE, but outlines problems in automotive RE. Therefore, it sketches possible *purposes*, *objects*, *stakeholders* and *points in time* for which models could be used in order to address the found problems. These aspects are marked red in Figure 1.5.

We performed an exploratory case study, collecting data from 14 interviews at two automotive companies, an OEM and a supplier, in order to answer RQ2. From the interviews, we extracted seven key problems related to organisation structure and communication, which we tested through a questionnaire with 31 practitioners from the automotive industry. The seven problems are

- P1: Lack of Product Knowledge: the lack of sufficient knowledge about the product in early stages;

- P2: Lack of Context Knowledge: the lack of context information regarding requirements on low levels of abstraction;

- P3: Unconnected Abstraction Levels: a mismatch between requirements on different abstraction levels;

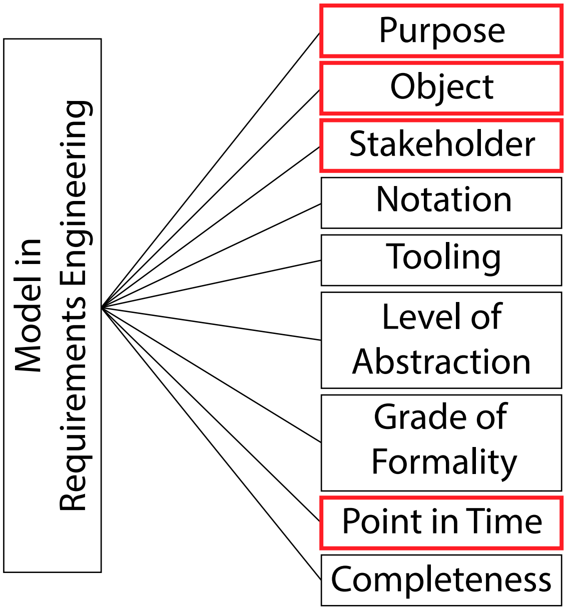

Figure 1.5: Scope of Paper B

- P4: Insufficient Communication and Feedback Channels: lacking communication with other people within or across the organisation;
- P5: Lack of Common Interdisciplinary Understanding: the lack of common understanding across multiple disciplines;
- P6: Unclear Responsibilities and Borders: the lack of clear and communicated responsibilities between different parts of the organisation; and
- P7: Insufficient Resources for Understanding and Maintaining Requirements: to lack enough resources in early phases to get an understanding of the needs and to maintain requirements later on.

Based on these problems, we see the following needs for future research. First, there is a *need for a process that allows for sufficient levels of uncertainty during early RE*. Uncertainty itself is not a new concept in RE and in project management, but it is becoming more and more important due to the increasing speed of technological change. Models could be used to clearly encode which parts of a specification are uncertain. Secondly, the *need for an organisation structure that effectively supports interdisciplinary RE, taking into account the central role of software* can be seen. Here, models could take the role of a common specification language across disciplines. Additionally, they could serve as an information source to enable coordination and communication, e.g., by visualising organisation structure. Thirdly, there is a *need for concepts and an organisation structure that allow for and support managing 'requirements debt'*. In automotive systems engineering, projects build on parts which are developed in previous projects, e.g., an existing requirements base or the electrical vehicle architecture. As long as there is no budget for maintaining and improving existing requirements, technical debt [56] for requirements, i.e., 'requirements debt', will be accumulated and never paid off. Models could serve as a means to manage this debt, e.g., by enriching specifications with meta information regarding the quality of requirements.

1.6.3 Paper C, Test sequence validation using requirements models

When requirements are modelled in a notation that allows execution, they can be used to provide stakeholders with early feedback regarding the system functionality. Furthermore, the models can be used for validation purposes throughout later development stages. In Paper C, we propose a scenario-based approach for requirements modelling using MSDs [57], a variant of Live Sequence Charts [58]. MSDs allow the specification of liveliness behaviour (something good must happen) and safety behaviour (something bad must not happen). Furthermore, they can be executed using the play-out algorithm [59].

The paper's contribution is twofold. First, we present the first tool environment able to model and execute MSD models with real-time properties. Secondly, we show the applicability of the approach by modelling requirements on gear shift behaviour from an upcoming emission standard in the automotive domain [60]. We then show how to validate gear shift sequences from test cycles generated by a third-party tool that implements this standard. The modelled requirements together with the evaluation answer RQ3.

In terms of the previously introduced classification scheme, the second contribution of this paper can be expressed as depicted in Figure 1.6. The *purpose* of the presented model is to perform verification activities by using the model as a test oracle. The model *object* are properties on gear shift behaviour, as described in the WLTP standard draft [60]. In Paper C, the authors serve as the *creators* of the model. However, if applied in industry, the models would be created by requirements engineers or domain experts. Therefore, it is not highlighted in the figure. *Receivers* of the requirement model are verification engineers who would use the model during testing. The *notation* we used in this paper are MSDs, with ScenarioTools [61] serving as a *tooling* environment. The *formality* of MSDs is formal, as they possess a defined syntax and executable semantics. The *level of abstraction* of the modelled requirements is on function level, following [26], as it describes what actions are possible or not possible to perform in the final system. The *point in time* is not defined for this approach. If used in industry, these kind of models could be created at various points in time, starting from early requirements elicitation to the evolution of the product, as new standards arise. Therefore, this aspect depends on the concrete implementation of the approach in practice. Finally, the models are *complete* with respect to the given abstraction level.

Paper C shows the applicability of the introduced approach to model behaviour requirements and use them as test oracles. In order to increase validity and to enable industry transfer, we plan to demonstrate the applicability on further real-life specifications from the automotive domain. Furthermore, we are actively working on several language extensions to support a wider range of

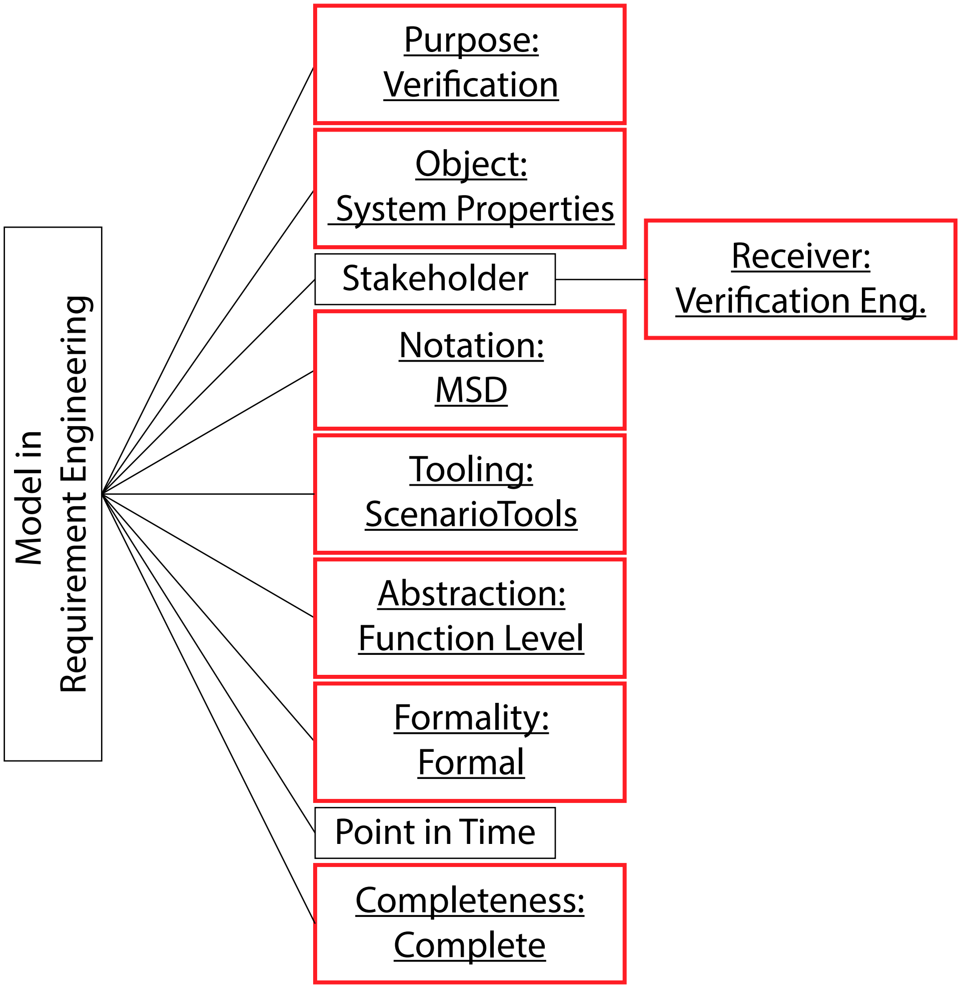

Figure 1.6: Scope of Paper C

behaviour requirements. Finally, we are investigating means to automatically transform requirements in a restricted natural language format into MSD models. This step would enable us to create models with a much lower effort and enable engineers without detailed modelling knowledge to create them.

1.6.4 Paper D, Comprehension of requirements expressed in two notations

Modelling notations are in practice often chosen ad hoc, e.g., due to existing tool licenses or experience with a notation. The actual implications of this choice are unclear. In particular, the comprehensibility of requirements modelled in one notation could be worse compared to the same requirements expressed in a different notation.

Paper D aims to increase the knowledge regarding comprehensibility of requirements expressed in different modelling notations. We conducted a controlled experiment with 22 students from an undergraduate course on software modelling to compare requirements from a real-life automotive specification. The students were divided into two groups and provided with a requirements specification modelled either in a sequence-based modelling notation, MSDs, or a state-based notation, Timed Automata (TA). Based on a comprehension questionnaire, we compare the comprehensibility of the two notations.

We cannot reject the null hypothesis, that there is no significant difference between the two notations, with respect to the score achieved on the comprehension questionnaires. However, subjects who received the MSD specification managed to answer significantly more questions than subjects who received the TA specification. This indicates that if the speed or the efficiency plays an important role, scenario-based models should be considered instead of the state-based models. This answers research question RQ4.

According to the previously introduced classification scheme, the contribution of this paper can be described as depicted in Figure 1.7. The *purpose* of

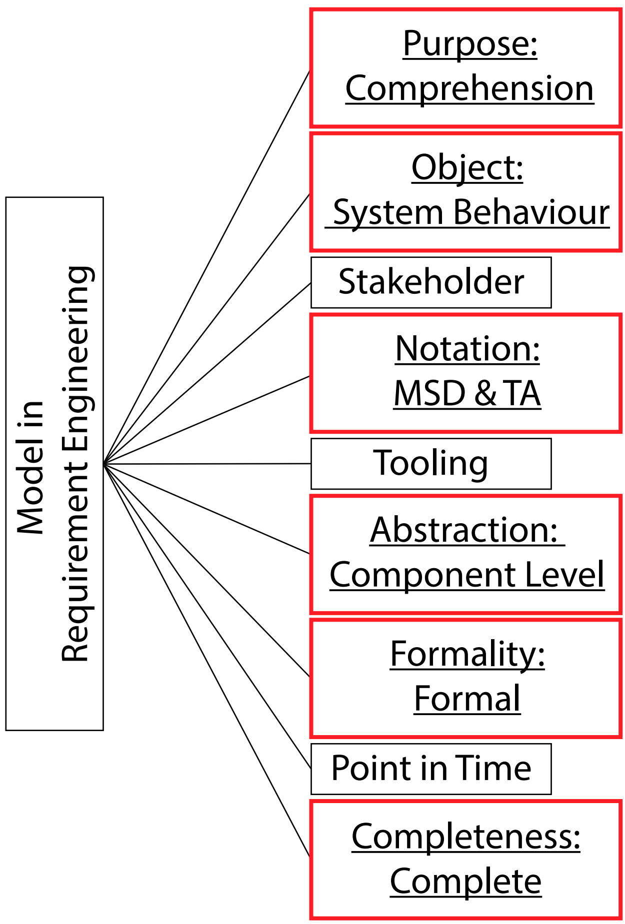

Figure 1.7: Scope of Paper D

the models used in the experiment is comprehension of the requirements, i.e., the classical purpose of a requirement. The *object* described by the models is the specification of the behaviour of a car wiper system. Similarly to Paper C, we serve as the *creators* of the model, but in industry the models would be created by requirements engineers or domain experts. Potential *receivers* of the requirement models are all receivers of a requirements specification, e.g., software engineers or verification engineers. The *notations* we used in the experiment are MSDs and TAs. In particular, this aspect is the independent variable of the controlled experiment. As the models were handed out in paper form, no *tooling* was used. The *formality* of both MSDs and TA is formal, as they possess a defined syntax and executable semantics. The *level of abstraction* of the modelled requirements is on component level, following [26], as a concrete implementation is described. This abstraction follows the original specification. The *point in time* is not defined for this approach, as the modelled requirements were already existing and, therefore, the models were reverse engineered. If used in industry, these kinds of models could most likely be created together with requirements of the same abstraction level, i.e., component-level requirements. The models are *complete* with respect to the asked questions, i.e., all necessary information is captured in them.

Paper D leaves much room for future work on studying how models are created, read and understood by practitioners. In particular, we are planning replications of the experiment using further real-life specifications from the automotive domain. Additionally, similar experiments could be conducted where practitioners create models instead of receiving finished ones. Finally, in order to understand how modelled specifications are read, we are planning studies using eye-tracking equipment.

1.7 Discussion

In summary, the research questions can be answered as follows.

MBE is widespread in the automotive domain and used for numerous purposes, such as simulation or test-case generation (RQ1). Also, the benefits of MBE are clearly seen by practitioners. It can therefore be assumed that MBRE is feasible in the automotive industry and in embedded systems generally.

However, shortcomings with MBE remain and should be considered in the future. Several of these are related to the effort required to introduce or use MBE.

This means for MBE in general, and for MBRE in the automotive domain in particular, that there are two ways of improving the situation. First, the effort to introduce and use MBE could be lowered, e.g., for training and for creating models. If successful, this could cause a much broader acceptance of MBE in industry, as the benefits and the potential of MBE seem to be recognised by practitioners. Secondly, the benefits received from MBE could be increased, e.g., by automating a larger proportion of transformations between models or enabling the models to be used for a wider range of purposes. In order to do so, a higher level of formality would have to be introduced.

These two ways are in general contradicting each other. If a higher degree of formality is required, the effort for creating these models and maintaining them is likely to increase, as well as the required training.

As a part of overall problems in automotive RE, we studied those related to communication and organisation structure (RQ2). For example, the lack of sufficient knowledge about the product in early stages or the lack of common understanding across multiple disciplines in automotive RE. The found problems raise several needs for future research, e.g., the need for a process that allows for sufficient levels of uncertainty during early RE. Models could be used as a means to address these needs.

Within the proposed solutions, the two contradicting ways of using more formality versus lowering the overall effort for RE are also visible. While some practitioners proposed to impose stricter rules or a more formal process, others proposed to relax the existing processes and lower the effort spent for RE-related activities.

A middle way between these two ways could be to use partial formal models, as demonstrated for RQ3. Here, the system behaviour does not have to be described completely, but the models are still formal and therefore enable simulation or testing. This would also lower the threat that the presented models are not suitable for all kinds of behaviour requirements. The effort to create formal models would still be required, but could be focused on smaller parts of the requirements specification, i.e., safety-critical or costly requirements. On a more general level, instead of focusing on generating running code, requirements models could be reused later on, e.g., as test oracles.

Another way to benefit from MBE during RE without a large effort would be to exploit already existing requirements specifications. One common approach in research to do so is to transform existing natural language requirements into models [62]. Similarly, the relationship between existing requirements and other systems engineering elements could be exploited. In this case, the model *object* would be the structure of the requirements specification instead of the requirements themselves. This is the approach used by modelling languages such as EAST-ADL [34], the AUTOSAR meta model [63] or requirements diagrams in SysML [35], languages which are all adopted in the automotive domain. Even if they are not necessarily visible to the user, many requirements management tools employ such a modelling language to structure requirements and relate them to each other or support export to standardised formats such as ReqIF [64]. These structural models could be used to address one or several of the problems found in RQ2.

To answer RQ4, we compared two modelling languages, MSDs and Timed Automata (TA), with respect to comprehensibility when modelling behaviour requirements. There are no significant differences between them when it comes to comprehensibility, but MSDs are quicker to read. This is an initial step towards providing guidelines for choosing modelling *notations*, but leaves a lot of room for interpretation and future work. We only compared two modelling notations and only one real-life specification. In particular, we do not investigate differences between different people creating a model of the same specification, differences between the same requirements being expressed in a single diagram and in multiple diagrams, and differences between different levels of abstraction. Studying these aspects is clearly relevant, but too much of an overhead for a single study. Therefore, this topic could be covered as a part of a larger research project in the future.

1.8 Validity Threats

In this section, we give an overview of the threats to the validity of the results of this thesis, i.e., the answers to research questions RQ1 to RQ4. Detailed threats to validity of the included publications are discussed in their respective chapters, i.e., Chapters 2 to 5.

RQ1 is intended to give an overview of MBE in the automotive domain. Therefore, we aim for a high external validity in order to be able to generalise the findings from our sample to the larger population. This validity could be restricted through selection effects. We distributed the survey within research projects we participate in and within our own industrial contacts. In particular, it can be expected that most people involved in research projects with the topic of modelling will also be favourable towards MBE. Therefore, we make no claims as of how representative our sample is of the overall population when it comes to extent of MBE in the automotive domain. However, our sample contains a relevant set of global companies working in the embedded domain. Furthermore, the results provide us with a valuable picture of MBE in industry and we receive indications on which particular shortcomings can be addressed in the future and which benefits are already present.

RQ2 aims at exploring problems in automotive RE, but not to provide an exhaustive list of problems. The results should again be generaliseable to the automotive domain. Due to choosing case study as a research strategy, external validity is low by design. The findings stem from two cases and 15 individuals. As these cannot possibly represent the entire automotive domain, we use the additional survey in order to test the findings and increase external validity. The survey confirms the case study findings, namely that the problems exist in a broader sample and that they are relevant.

RQ3 demonstrates how models of behavioural requirements can be used for testing purposes. The main threat for this result is that the used modelled notation and the presented extensions to it are too restricted to model a larger population of behaviour requirements. While this might especially be the case for already existing specifications, we expect that newly created requirements could be expressed in the proposed notation. However, smaller notation extensions are still required, e.g., global variables, and will be addressed in the future. Furthermore, we plan to investigate what kind of properties a specification has to have in order to be representable as MSDs. These properties will then influence the guidelines for choosing a modelling notation for expressing behaviour requirements.

For answering RQ4, the control introduced as a part of the experimental setup has threats to external validity. While the controlled environment allows for systematic analysis of the dependent variables, it does not necessarily scale to a real-life environment. This is a drawback of the controlled environment, which was chosen in favour of the extended amount of control. In order to address this threat, the experiment needs to be complemented by qualitative studies that study the use of modelling notations in a real-life context. Additionally, we believe that the way the models were created plays an important role and might introduce bias. This is an interesting point, as the way models are created is a process of abstraction, which does not follow explicit and established rules. Therefore, if someone else would create the models, the results could differ.

1.9 Conclusions and Future Work

With this thesis, we contribute to the body of knowledge in automotive MBRE by providing an overview of the current state-of-practice in automotive MBE (RQ1) and specific problems in automotive RE (RQ2). Regarding RQ1, there is already a large body of knowledge of using formal models for different tasks. Therefore, an option for future work would be to investigate ways to use formal or semi-formal models during RE, as mentioned in Section 1.7 as a middle way between using formal models and lowering the effort for MBE. Possible ways could integrate formal models with other artefacts, such as source code or informal models. Simulink [65] is a prominent example of such an approach, by integrating C code with block diagrams. While this prevents formal analysis, e.g., model checking, simulation is still possible.

In the context of RQ2, the question of organisation structure is raised. Many of the problems occur because there is no or only little communication across organisation borders. Here, future work should investigate whether requirements can be structured in a way that fits the organisation structure better. Vice versa, different organisation structures could be devised to fit the functional structure of software systems, which take an ever larger portion of the overall effort in automotive systems engineering.

Additionally, we report two concrete studies in MBRE (RQ3 and RQ4) as initial results of using models in automotive RE. For RQ3, we demonstrate how models of behaviour requirements can be used as test oracles. While the approach works for the provided example, gear shift requirements from a real-life specification, it needs to be shown in a broader context. Furthermore, the effort of creating the models is currently high.

For RQ4, our study shows no significant differences between the understandability of behaviour requirements expressed in MSDs and in TAs. However, the study needs to be replicated in order to make general claims regarding the understandability of modelling notations.

To reach the overall PhD goal, to provide guidelines for using models in automotive RE, several steps will be taken in the future. The answers to RQ1 and RQ2 allow us to make certain statements as to which shortcomings in MBE need to be considered and for which problems in automotive RE MBE could be used. We plan to extend this body of knowledge in order to be able to provide more detailed guidance. First, we will narrow the scope of RQ1 and investigate how models are used in automotive RE. In order to obtain in-depth knowledge, we will do so in terms of a case study. Secondly, we are currently studying the use of requirements specification expressed in structural modelling notations, such as EAST-ADL [34], to tackle several of the problems reported for RQ2. To do so, we exploit ownership and change information of requirements and their relations to construct social networks of automotive organisations. We hope that this will help practitioners to establish effective communication across organisation boundaries.

Furthermore, we aim to increase the knowledge of how modelling notations are used and understood in practice. Therefore, we plan to replicate or complement the study presented in Paper D. In particular, we need to understand how different modelling notations are read and whether this affects comprehension of the modelled content (the model *object*). Apart from using comprehension questionnaires, eye-tracking, as used in, e.g., [66], could be a suitable way to understand how humans read diagrams. In particular, it would be beneficial to make recommendations on which modelling *notations* should be used depending on the *purpose*, *point in time*, *abstraction level* and model *receivers*. The outcomes of RQ1 and RQ2 serve as a base for selecting a subset of these aspects for further investigation, e.g., by studying configurations that address the problems reported for RQ2. For *tooling*, recommendations could be made on how models in RE should be visualised and used in conjunction with other artefacts in RE. Finally, we plan to collect qualitative data, e.g., through interviews, in order to understand how requirements models are created and read by engineers.