Assessing the State-of-Practice of Model-Based Engineering in the Embedded Systems Domain

Grischa Liebel[1], Nadja Marko[2], Matthias Tichy[1], Andrea Leitner[2], and Jörgen Hansson[3]

[1] Software Engineering Division, Chalmers | University of Gothenburg, Sweden
grischa@chalmers.se|matthias.tichy@cse.gu.se
[2] Virtual Vehicle Research Center, Graz, Austria
nadja.marko@v2c2.at|andrea.leitner@v2c2.at
[3] School of Informatics, University of Skövde, Sweden
jorgen.hansson@his.se

Abstract. Model-Based Engineering (MBE) aims at increasing the effectiveness of engineering by using models as key artifacts in the development process. While empirical studies on the use and the effects of MBE in industry exist, there is only little work targeting the embedded systems domain. We contribute to the body of knowledge with a study on the use and the assessment of MBE in that particular domain. We collected quantitative data from 112 subjects, mostly professionals working with MBE, with the goal to assess the current State of Practice and the challenges the embedded systems domain is facing. Our main findings are that MBE is used by a majority of all participants in the embedded systems domain, mainly for simulation, code generation, and documentation. Reported positive effects of MBE are higher quality and improved reusability. Main shortcomings are interoperability difficulties between MBE tools, high training effort for developers and usability issues.

Keywords: Model-Based Engineering, Model-Driven Engineering, Embedded Systems, Industry, Modeling, Empirical Study, State-of-Practice.

1 Introduction

Model-Based Engineering (MBE)[1] has a long history in the embedded systems domain. For example, the first version of Matlab/Simulink has been released exactly 30 years ago and by now, it is one of the standard development tools in the automotive domain. MBE aims to increase the effectiveness and efficiency of Software Development [4]. However, empirical evaluation of MBE in industry is scarce [12]. The few existing empirical studies in this field suggest that MBE can have positive effects such as reduction of defects and productivity improvements [3, 12], or increased understandability [10]. Nevertheless, they also report

[1] We use the terms Model-Based Engineering and Model-Driven Engineering interchangeably for a process in which models are used as the primary artifacts.

J. Dingel et al. (Eds.): MODELS 2014, LNCS 8767, pp. 166–182, 2014.

challenges such as insufficient tool support [3, 12, 13], need for additional training [10] or the use of MBE with legacy software [10,12]. However, existing studies are not explicitly targeted at the embedded systems domain [3, 9, 10, 12–14], target only UML [2,5,8], limit themselves to the Brazilian embedded industry [1], or collect only qualitative data from the automotive domain [11]. We contribute to the body of knowledge with a survey on the use of MBE in the embedded systems domain. The goal of the survey was to get an overview about the SoP and challenges the industry is faced with in order to understand industrial needs. More precisely, with the study we want to answer the following research questions:

– **RQ1:** What is the current state of practice and the assessment of Model-Based Engineering in the embedded systems domain?
– **RQ2:** How does the use and the assessment of Model-Based Engineering differ between different demographic subgroups in the embedded systems domain?

RQ1 aims to capture the SoP of MBE in the embedded systems domain, which includes the used modeling environments, modeling languages, types of notations, purposes models are used for and how much activities concern MBE compared to non-MBE. Moreover, we are interested in the introduction reasons and the effects, both positive and negative, after introduction of MBE as well as current shortcomings of this method. With **RQ2**, we want to find out whether there are substantial differences in the SoP between different groups in the embedded systems domain, e.g., differences in the automotive domain and the avionics domain or between new MBE users and highly experienced users.

In order to answer the research questions, we developed a web survey consisting of 24 questions. The survey was distributed to partners taking part in five industrially driven European research projects (between 22 and 100 project partners) as well as to personal contacts of which most are professionals working with MBE. Finally, we have got 121 completed surveys from which 112 are used for the data analysis.

In this paper, we focus on the presentation of the reported positive and negative effects of MBE, shortcomings of MBE, reasons for introducing MBE and purposes models are used for in the development process. Overall, the survey answers show that many survey participants think that the positive effects predominate the negative effects of MBE. Nevertheless, they mention also that interoperability challenges between tools exist and that it causes high efforts to train the developers. More detailed results will be discussed in Section 4. The complete data sample together with the questionnaire is published at `www.cse.chalmers.se/~tichy/models14_LMTLH_dataset.zip`[2].

The remainder of this paper is structured as follows. In the following section, we discuss related work. Section 3 contains the research methodology. This includes the process of study design, data collection, threats of validity. In Section 4, the key results of the survey are discussed. Finally, conclusions and future work are discussed in Section 5.

[2] Password: mbe_usage14

2 Related Work

While industrial evaluation of MBE in research is limited [10], there are a number of recent publications addressing this topic. With respect to the embedded systems domain, we are only aware of two reported studies, [1] and [11], presenting the SoP of MBE in this particular domain. Other publications, such as [3, 12] and [9], also include cases from the embedded systems domain, but do not explicitly address this domain as their target.

In [1], Agner et al. present the results of a survey on the use of UML and model-driven approaches in the Brazilian embedded software development industry. The participants come from a variety of different sub-domains, with industrial automation, information technology, telecommunications and electronic industry being the biggest groups. Key findings are that 45% of the 209 participants use UML. Of these 45%, the majority are experienced developers working at medium-sized companies. The subjects report increases in productivity and improvements in quality, maintenance and portability as key advantages of model-driven practices. According to the participants, the use of UML is mostly hindered by short lead times, lack of knowledge regarding UML and a limited number of employees with expert UML knowledge. Additionally, it is stated that models are mainly used for documentation with only little use of code generation or model-centric approaches in general. In contrast to [1], we do not limit ourselves to a region but include a wide range of subjects from global companies based in Europe.

Kirstan and Zimmermann report a case study within the automotive domain [11]. Their interviewees report positive effects of MBE like an earlier detection of errors, a higher degree of automation and cost savings during the initial phases of development. On the negative side, they state that large function models can become too complex and that interoperability between tools is difficult. The study is limited to qualitative data from a single sub-domain of the embedded systems domain, namely automotive.

Baker et al. present experiences with MBE at Motorola over a time span of almost 20 years in [3]. On the positive side, they report a defect reduction and an improvement in productivity. However, a number of challenges regarding MBE are named as well, such as lack of common tools, poor tool and generated code performance, lack of integrated tools, and lack of scalability.

Mohagheghi and Dehlen published a literature review on the industrial application of MBE [12]. The evidence collected during the review suggests that the use of MBE can lead to improvements in software quality and productivity. However, studies which report productivity losses are also quoted in the review. Insufficient tool chains, modeling complexity, and the use of MBE with legacy systems are reported as challenges. Additionally, the maturity of tool environments is stated to be unsatisfactory for a large-scale adoption of MBE. Generally, the authors conclude that there is too little evidence in order to generalize their results.

In a later publication by Mohagheghi et al., experiences from three companies in a European project "with the objective of developing techniques and tools for applying MDE" are reported [13]. According to the experiences at the studied companies, advantages of using MBE include the possibility to provide abstractions of complex systems, simulation and testing, and performance-related decision support. However, the authors also state that the development of reusable solutions using MBE requires additional effort and might decrease performance. Moreover, transformations required for tool integration can increase the complexity and the implementation effort according to the authors. Furthermore, the user-friendliness of MBE tools and means for managing models of complex systems is described as challenging.

Hutchinson et al. report industrial experiences from the adoption of MBE at a printer company, a car company and a telecommunications company in [9]. The authors conclude that a successful adoption of MBE seems to require, among others, an iterative and progressive approach, organizational commitment, and motivated users. The study is focused mainly on organizational challenges of MBE.

A further assessment of MBE in industry by Hutchinson et al. based on over 250 survey responses, 22 interviews, and observational studies from multiple domains is presented in [10]. From their survey, the authors report that significant additional training is needed for the use of MBE, but that MBE in turn can speed up the implementation of new requirements. Furthermore, the survey indicates that code generation is an important aspect of MBE productivity gains, but integrating the code into existing projects can be problematic. The majority of survey participants states that MBE increases understandability. From their interviews, the authors conclude that people's ability to think abstractly can have a huge impact on their ability to model. Hence, this ability influences the success of MBE.

According to a survey of 113 software practitioners reported by Forward and Lethbridge, common problems with model-centric development approaches are, among others, inconsistency of models over time, model interchange between tools and heavyweight modeling tools [7]. Code-centric development approaches, on the other hand, make it difficult to see the overall design and hard to understand the system behavior.

Torchiano et al. present findings from a survey on the State of Practice in model-driven approaches in the Italian software industry [14]. From the 155 subjects, 68% report to always or sometimes use models. The subjects who do not use models commonly state that modeling requires too much effort (50%) or is not useful enough (46%). Further findings are that models are used mainly in larger companies and that a majority of all the subjects using models (76%) apply UML.

Further empirical evaluations on the application of UML in particular can be found in [2, 5, 8]. These publications are related to our survey with respect to some aspects, such as UML notation types. However, they do not address MBE, or any approach where models are the primary artifact, in particular. Therefore, they are not discussed here in detail.

In conclusion, commonly reported problems in industry are insufficient tool support or tool chains, using MBE together with legacy systems, and the complexity of MBE and modeling in general. On the positive side, productivity gains, defect reductions and increased understandability are reported. However, there is a lack of empirical evidence and reported industry evaluations on the use of MBE within the embedded systems domain. Existing work is either not targeted at the embedded systems domain in particular [3, 9, 10, 12–14], is limited to the Brazilian market [1], or lacks quantitative data [11].

3 Research Methodology

This section outlines the research methodology, consisting of the study design, an outline of the data collection and threats to validity.

3.1 Study Design

The study was designed by three researchers from two different institutions and three practitioners from two different companies as part of the CRYSTAL project.

We decided to perform a survey in order to reach a larger sample size compared to other empirical strategies and, thus, get an overview of the embedded systems domain.

The survey questionnaire consisted of 24 closed-ended and open-ended questions. The first part of the questionnaire contained 13 questions gathering demographic data. Hereby, we asked for company size, position in the value chain, domain, experience with MBE, product size, working tasks, and the attitude towards MBE. The second part, consisting of the remaining eleven questions, addressed **RQ1**. Due to space limitations, we only use questions for the data analysis in this paper regarding the positive and negative effects of MBE, shortcomings of MBE, reasons for introducing MBE and purposes models are used for. The answers for all four questions were scored on a 5-level likert scale. Both parts of the questionnaire were considered together for answering **RQ2**.

The survey was piloted by eleven colleagues in academia and industry. Given their feedback and the time they needed to fill out the survey, the questionnaire was refined. The revised survey was reviewed a second time by one colleague not included in the pilot survey.

Furthermore, we derived a list of 24 hypotheses from the related work discussed in Section 2 (see Table 1) in order to guide the data analysis for **RQ1**. These were then evaluated based on our collected data. The descriptions of hypotheses **H1.1** through **H1.9** are summaries of the actual statements in the related work, based on our understanding. This is due to the fact that similar statements are present in multiple sources. For instance, Hypothesis **H1.5** describes tool quality in general, while Baker et al. talk about poor tool performance [3], Mohagheghi and Dehlen report lack of maturity of third-party tool environments [12], Mohagheghi et al. report challenges with the user-friendliness of tools [13], and Forward et al. report that heavyweight modeling tools are problematic [7]. While we lose the exact statements for **H1.1** through **H1.9** from related work, we argue that this summary is helpful for getting an overview over the findings in the area of MBE. We do not claim that this list of hypotheses is complete. However, we believe that it can guide future research in this area. Additionally, we derived a list of eight hypotheses in order to answer **RQ2**. We

Table 1. Hypotheses from related work

Hypoth.	**Description**	**Reported by**
H1.1	MBE leads to a reduction of defects/improvements in quality.	[1,3,12]
H1.2	MBE leads to improvements in productivity.	[1,3,12]
H1.3	MBE increases understandability.	[10], partly [13]
H1.4	Using MBE with legacy systems is challenging.	[10,12]
H1.5	Current MBE tools are insufficient.	[3,7,11–13]
H1.6	Significant additional training is needed for using MBE.	[1,10]
H1.7	UML is the preferred modeling language employed in MBE.	[1,14]
H1.8	Managing models of complex systems is challenging.	[11,13]
H1.9	Tool integration is challenging.	[7,11,13]
H1.10	Code generated from models has poor performance.	[3]
H1.11	MBE lacks scalability.	[3]
H1.12	The complexity of modeling is challenging.	[12]
H1.13	Advantages of MBE are simulation and testing, and performance-related decision support.	[13]
H1.14	MBE leads to an earlier detection of errors.	[11]
H1.15	MBE can speed up the implementation of new requirements.	[10]
H1.16	Code generation is an important aspect of MBE productivity gains.	[10]
H1.17	Companies which consider software development their main business seem to find the adoption of MBE more challenging than other companies.	[10]
H1.18	Modeling requires too much effort.	[14]
H1.19	Handling the consistency of models over time is challenging.	[7]
H1.20	Modeling is not useful enough.	[14]
H1.21	Models are used mainly in larger companies.	[14]
H1.22	UML is mostly used by experienced developers working at medium-sized companies.	[1]
H1.23	There is little use of code generation or model-centric approaches.	[1]
H1.24	MBE leads to a higher degree of automation.	[11]

derived these hypotheses after designing our questionnaire from our own view on MBE. That is, we elicited the hypotheses based demographic subgroups which we were able to distinguish in our survey. The alternative hypotheses that there are significant differences between the subgroups are listed in Table 2. The corresponding null hypotheses are that there are no significant differences between the subgroups.

Table 2. Hypotheses defined for RQ2

Hypoth.	Description
H2.1	Users of in-house tools report more positive and less negative effects of MBE than users who do not use in-house tools.
H2.2	Supporters of MBE report more positive effects than subjects opposed to or neutral towards MBE.
H2.3	Subjects who are still using MBE report more positive and less negative effects than subjects who stopped using MBE.
H2.4	Subjects who only use models for means of information/documentation report less positive than negative effects.
H2.5	Subjects who do not see many usability issues with MBE tools report fewer negative effects.
H2.6	Highly experienced users of MBE report less problems with MBE tools than users with less experience.
H2.7	Large companies have more tool integration problems than small or medium-sized enterprises.
H2.8	MBE promoters use more MBE tools in comparison to subjects neutral or opposed to MBE.

3.2 Data Collection

The theoretical target population of the survey are all people involved with systems engineering from the embedded systems domain, e.g. software architects, software developers, project managers, system engineers. We distributed the survey to partners taking part at the Artemis projects Crystal (70 partners), VeTeSS (22 partners), MBAT (38 partners), nSafeCer (29 partners), and EMC^2(100 partners), as well as to personal contacts of which most are professionals working with MBE. This can be described as a convenience sample. However, we also encouraged recipients to distribute the survey to colleagues or partners. We used an online survey[3] in order to keep administration costs low and facilitate the distribution.

The final version of the survey was published on 18th October 2013 for a time period of six weeks. Out of 196 started surveys, 121 were completed corresponding to a completion rate of 61.73%. The survey data was automatically coded and enhanced with additional quality data by the survey tool, such as completed answers and time to fill out the survey. We cleaned the remaining 121 surveys based on degradation points computed from missing answers and the time to fill out each survey page. As we did not use compulsory questions, it could happen that subjects lost interest but still navigated through the entire survey until the end or simply looked at the survey without filling in data. Therefore, we argue that this data cleaning process is necessary in order to ensure data validity as discussed in [15]. We excluded nine surveys based on a threshold of 200 degradation points proposed by the survey tool for a light data filtering. This left us with 112 answered surveys for data analysis. We made adaptations to the

[3] Through `www.soscisurvey.de`

demographic data in cases where free-text answers clearly corresponded to one of the given answering options.

3.3 Validity Threats

In the following, we discuss the four different aspects of validity as discussed in Wohlin et al. [15].

Construct Validity. Construct validity reflects whether the studied measures are generalizeable to the concept underlying the study. We collected data from different sources in order to avoid mono-operation bias. Hypothesis guessing, the participants guessing what the researchers are aiming for and answering accordingly, can not be ruled out completely. We tried, however, to formulate the questions in a neutral way and improved the questionnaire based on obtained feedback from the pilot study in order to address this threat. Finally, answers were treated completely anonymous in order to avoid biased answers due to evaluation apprehension.

Internal Validity. Internal validity reflects whether all causal relations are studied or if unknown factors affect the results. Instrumentation was improved by using a pilot study. The survey took approximately 15 minutes to fill out and was intended to be filled out once by every participant. This reduces the likelihood for learning effects and, hence, maturation effects. Additionally, the completion rate of 61.73% indicates that the majority of participants was interested in finishing the survey. Selection threats can not be ruled out as participants volunteered to fill out the survey.

External Validity. External validity is concerned with the generalizeability of the findings. The CRYSTAL project and other projects, to which the survey was distributed, consist of partners from all major sub-domains of the embedded systems domain. Additionally, demographic data was collected in order to confirm this aspect. Therefore, we are confident that we have reached subjects with a variety of different backgrounds representative for the embedded systems domain. While CRYSTAL is a project on European level, many of the involved partners are global companies. Hence, we argue that this does not limit the validity of our results and that it is possibile to generalize them to other cases on non-EU level.

Conclusion Validity. Conclusion validity is concerned with the ability to draw correct conclusions from the studied measures. We involved three researchers and three practitioners with different background into the study design. Therefore, the survey was designed by multiple people with different aims and backgrounds, which should reduce the risk for "fishing" for results. A standard introduction e-mail was designed to be distributed with the link to the online survey.

Hence, reliability of treatment implementation is given. Reliability of measures was increased through a survey pilot filled out by eleven people and then, after improvements, reviewed by one more researcher. The detailed questionnaire is furthermore published in order to enable replications and an assessment of the validity of our study. Significance tests were only performed based on our hypotheses. That is, we did only perform a fixed number of statistical tests and did not randomly search for significant results.

4 Results

This chapter summarizes the results of the survey. First, demographic data about the subjects participating in the survey is illustrated in order to get information about their company and experiences. Then, **RQ1** is addressed and, where possible, compared to hypotheses **H1.1** to **H1.24** in order to show our survey results and compare it to related work. Finally, we discuss **RQ2** based on hypotheses **H2.1** to **H2.8** and evaluate validity of the hypotheses.

4.1 Demographic Data

The first part of the survey contained context questions providing demographic data. Mainly, two kinds of background information have been asked; first, some context questions concerning the company and secondly, questions about the personal MBE experiences of the participants. With the company related questions we wanted to get an idea of the work environment such as domain, company size or company position. Questions about the personal experiences such as daily working tasks, usage of MBE or whether the participant is a supporter for MBE or not should help to better understand answers and opinions of the surveyed subjects.

Company context. From the 112 surveys a bit more than the half stated the company they worked for; consequently, at least 30 different companies could have been identified that participated in the online survey. About three-fourths of all respondents (87) work in large companies with more than 250 employees, 14 persons are employed in small and medium enterprises (SME) and 11 at university. Hence, the main percentage of answers represent opinions of large companies. 50 of the companies are first-tier supplier, 40 OEMs, 25 second-tier supplier and 18 have other positions in the value chain such as research institutes, consultants or technology/software provider. More than a half of the respondents (60) work in the automotive industry, 31 in avionics, 25 in health care, 15 in defense, 11 in rail and 4 in telecommunications. 16 companies work domain independently and 9 operate in other domains such as semiconductor or industrial automation industry. Asking the participants the point in time their company introduced MBE, 37 say that their company started 10 or more years ago, 56 state 1-10 years ago and 4 started in the last 12 months. 8 companies still do not apply MBE, the rest (10 participants) does not know the introduction time. Accordingly, most companies have experiences with MBE for quite some time. 73 companies use MBE for developing a commercial product, 46 therefrom for large scale series production (more than 1000 pieces), 19 for medium scale production and 8 for small scale production (less than 10 pieces). 23 use MBE for research demonstration, 9 use it for non-commercial products and 7 for other purposes such as teaching or developing methods and tools.

Personal experiences. In order to understand for which activities the participants use MBE, we asked for their main working tasks. The answers, multiple answers were possible, are: 60 of the participants implement software, 56 are responsible for architecture definition, 55 for testing, 53 for design definition, 49 specify requirements, 39 are project managers, 24 are safety managers, 16 are quality managers, 14 are responsible for customer support and 12 work in general management. 14 participants execute other activities than the mentioned, such as process improvement, consulting or tool engineering. Hence, we cover a diversity of subjects working in different functions. Concerning the MBE experience, many participants (46) are well experienced with more than 3 years of usage. 40 persons state that they have moderate experience and only 26 are new in the field of MBE. 72 of the participants are still using MBE, 15 have used MBE the last time 1 month to 1 year ago and 16 have used MBE the last time more than 1 year ago. Only 9 people state that they have never used MBE; thus, a large percentage of the survey participants are experienced. 86 of subjects are promoters for MBE, 25 have a neutral attitude for MBE and 0 are opponents.

4.2 RQ1: State of Practice

The key results of the survey should offer valuable clues to industrial needs concerning MBE. Mainly, reasons for applying MBE, effects of using it, shortcomings of MBE and model purposes represent interesting outcomes of the survey.

Modeling tools and languages. Even though we do not focus on presenting the answers about modeling tools and languages in this paper, we present a summary about the most used tools and languages as context for the following results. Regarding the technical aspects of MBE, we asked the participants which languages and notation types they use for modeling and which functional aspects of their system they describe using models. Most survey participants use Matlab/Simulink (50 answers) or Eclipse-based (34 answers) MBE tools. As for notations, Finite State Machines are used by 74 participants, followed by sequence-based models (64 participants) and block diagrams (61 participants). Finally, we asked for which functional aspects of a system participants already use models. Here, structure (68 participants), discrete state specifications (48 participants) and static interfaces (47 participants) are most common.

Needs for introducing MBE. One interesting issue is the motivation why companies decide to use models for developing their systems. Reasons for introducing MBE will give information about companies' opinions regarding the advantages of MBE as well as challenges they are faced with. Therefore, the survey contains one question asking about the needs for introducing MBE. The results are summarized in Figure 1.

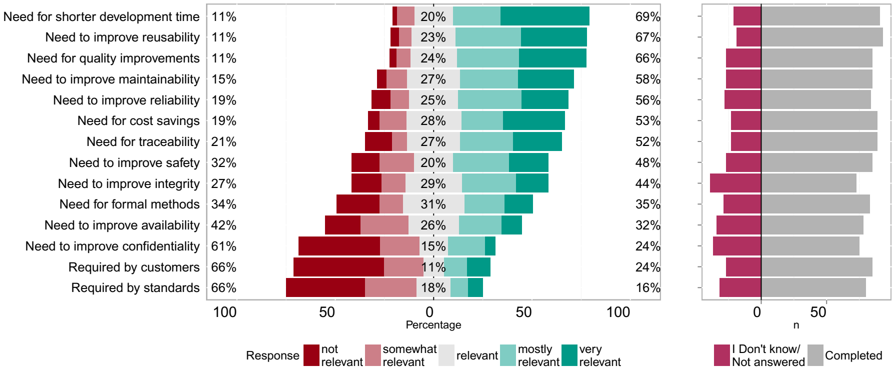

Fig. 1. Reasons for introducing MBE

On the left side of the figure the needs, which have been stated in the questionnaire, and the responses concerning the needs are listed. The three percentage declarations in the figure show on the left side the percentage of the answers with 'not relevant' and 'somewhat relevant', in the middle the percentage of the neutral 'relevant' answers and on the right side the percentage of answers with 'mostly relevant' and 'very relevant'. The second part of the figure located on the right side gives information about the amount of participants who filled in the grade (completed) and the number of participants who did not fill in a grade or do not know it (Not answered/I don't know). The figures in the following sections can be read equally but with adapted questions, responses and response types. As the figure shows, most participants (69%) think that their company adopted MBE because they had a need for shorter development time. Further, more than 50% say that needs for reusability, quality, maintainability and reliability improvements as well as cost savings and traceability are reasons for applying MBE. Least important for the respondents are needs to improve availability and confidentiality and that MBE is required by customers or standards.

Purpose of models. Further, we wanted to know for which purposes models are used for. The results for this question are illustrated in Figure 2. According to the responses, models are mainly used for simulation, code generation, opposing **H1.23**, and for information/documentation; hence, the automation of activities in the development process seems to be an important function. In contrast, timing analysis, safety compliance checks, reliability analysis and formal verification have not as much application as the other mentioned purposes.

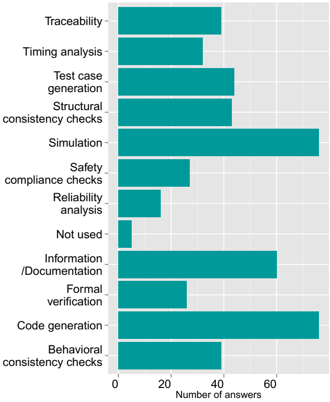

Fig. 2. Model purpose

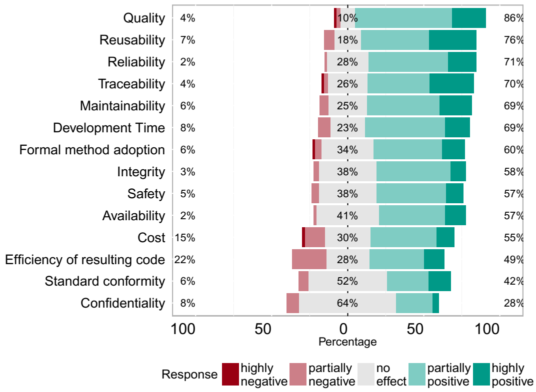

Fig. 3. Positive and negative effects of MBE

Positive and negative effects of MBE. In addition to the needs for introduction, the effects of the actual use of MBE are interesting. There are positive and negative effects when applying MBE; hence, we asked 'What were the effects of introducing MBE in your division/department?'. Figure 3 shows the answers for this question. For this question, between 5 and 9 people did not answer each item and between 30 and 53 did not know the effects. Accordingly, quality, reusability, reliability, traceability, maintainability, development time, formal method adoption, integrity, safety, availability, cost and efficiency of resulting code are rated highly or partially positive by most participants. Standard conformity and confidentiality have no effect according to more than 50% of the surveyed subjects. Thus, most survey participants think that MBE has more positive effects than negatives. From related work, **H1.1** (quality improvements) is supported by the data. **H1.2** (productivity improvements) and **H1.15** (increased development speed of new requirements) are supported with respect to development times. Other aspects of these hypotheses, such as productivity improvements due to increased efficiency, are not captured by our questionnaire. Finally, **H1.3** is supported with respect to maintainability.

Shortcomings of MBE. In order to identify potential improvements, subjects were asked about current shortcomings of MBE. Figure 4 shows the answers for this question which range from does not apply at all to fully applies. Many survey participants think that difficulties with interfaces to inter-operate with other tools is a shortcoming that fully or mostly applies. This is in line with survey results in [7], supporting **H1.9**. Moreover, more than one third of the people thinks that MBE requires a high effort to train developers (supporting **H1.6**), that there are usability issues with tools (supporting **H1.5** with respect to usability) and that benefits require high efforts (supporting **H1.18** and supporting **H1.20** with respect to the required effort). Even though **H1.18** is supported by

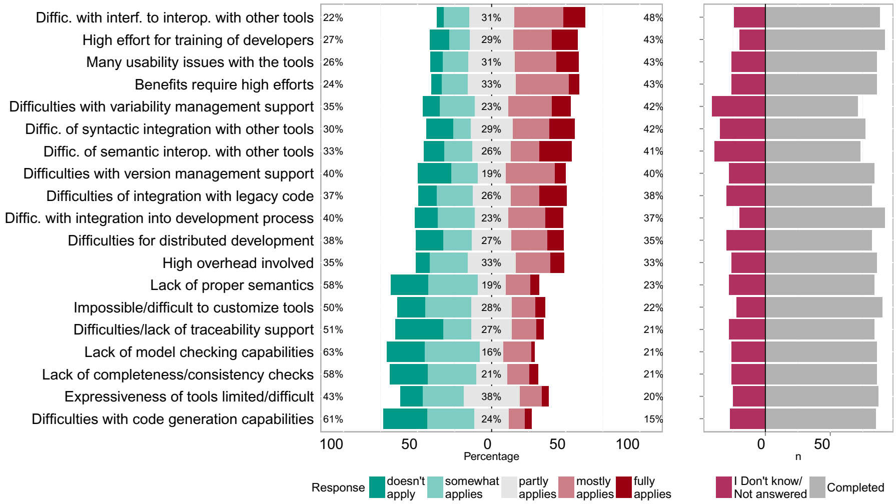

Fig. 4. Shortcomings of MBE

"benefits require high efforts", opinions about whether high overhead is involved with the usage of MBE vary. No shortcomings according to the responses are difficulties to customize tools and limitations on what can be expressed within tools what is in opposite to **H1.5** with respect to customization aspects and **H1.5** with respect to expressiveness. Hence, although the interoperability between tools seems to be a main shortcoming, capabilities of single methods and tools are satisfactory for many surveyed subjects.

MBE tool usage. In order to judge how familiar subjects are with MBE tooling, we asked how much they use MBE tools in comparison to non-MBE tools. Here, 5 subjects stated to not use any MBE tools, 26 answered that they use less MBE tools than non-MBE tools, 46 use more MBE tools than non-MBE tools and 11 use only MBE tools. Finally, 8 answered that they do not perform any engineering activities. Hence, the majority of all participants use mainly MBE tools during their work.

All in all it can be said that many survey participants think that the positive effects predominate the negative effects of MBE. However, the interoperability between tools and the usability of them, the effort to train developers as well as that the benefits require high efforts are considered as the main shortcomings of MBE.

4.3 RQ2: Differences by Subgroups

In the following, we discuss the results on research question 2 with respect to our hypotheses about differences in answers of subgroups of survey participants (cf.

Table 2). As shown in the previous section, the answers of the survey participants are ordinal scaled, e.g., a likert scale in the question about positive and negative effects of MBE. Thus, we have to use a statistical test which supports ordinal scaled data to assess whether the differences are significant. We use Fisher's exact test [6] (two-tailed) with a level of significance $\alpha \leq 0.05$. This test is a non-parametric statistical test for contingency tables. In our case, the contingency table consists of the answers of the participants in the columns and the different subgroups in the rows.

The hypotheses **H2.1**, **H2.2**, **H2.3**, **H2.4**, and **H2.5** address the full list of positive and negative effects as presented in Section 4.2. We check and report significance for each effect (e.g., cost, quality) individually.

For hypothesis **H2.4**, we do not have enough data for each subgroup in order to compare the groups. Hypotheses **H2.1**, **H2.6** and **H2.7** did not show any significant differences (i.e. $p \geq 0.05$) between the subgroups. Hence, here we can not reject the null hypotheses.

It is common that supporters of a paradigm or a methodology perceive its advantages much more positively than subjects who do not support it. Therefore, we tested this hypothesis for the case of MBE supporters and MBE opponents or neutral participants (**H2.2**). Traceability ($p = 0.00017$), safety ($p = 0.018$), and reusability ($p = 0.019$) yielded significant differences. That is, supporters of MBE perceive the effects of MBE on these three aspects significantly more positive than subjects opposed to or neutral towards MBE (See Fig. 3 for the complete sample). On traceability, 80% of MBE supporters report partially or highly positive effects, in contrast to only 27% for the opponents and neutral participants. Note that in our sample there are no opponents of MBE.

Similarly, it could be expected that participants who still use MBE also see more positive effects of MBE than participants who stopped using MBE (**H2.3**). However, significant differences exist only for cost ($p = 0.016$) and traceability ($p = 0.006$). That means that participants who are still using MBE report in total more positive effects on cost and traceability than participants who stopped using MBE. For instance, 79% of the participants still using MBE report partially or highly positive effects on traceability, while participants who stopped using MBE report only 48%. A possible explanation for the few significant differences might be that participants who stopped using MBE did so because they moved to a different position, e.g. in management, and not because they did not see the benefits of MBE.

Tooling in MBE is often reported to be insufficient. We would expect that usability issues with tools also influence other aspects such as productivity or quality negatively. Therefore, we investigated whether subjects who see many usability issues with MBE tools also report more negative effects than other subjects (**H2.5**). However, there is only a significant difference with respect to quality ($p = 0.011$). Participants who reported that many usability issues with tools mostly or fully applies rated the effects on quality slightly less positive (10% highly or partially negative, 13% no effect, and 77% partially or highly positive) than participants who reported that usability issues apply at most partly (0%, 7% and 93%).

Supporters of MBE also use more MBE tools in comparison to subjects who are opposed to or neutral towards MBE (**H2.8**) ($p = 0.00046$, less-than Fisher test). Here, 51 supporters of MBE reported to use MBE tools more than non-MBE tools or only MBE tools, and 18 reported to use less MBE tools than non-MBE tools or no MBE tools at all. This contrasts with a score of 5 and 13 answers on the opponent/neutral side.

In total we performed 72 significance checks resulting in seven significant differences. While the number of found significances is low for this amount of significance checks, we believe that our results could be used as indicators for future studies.

5 Conclusions and Future Work

The presented results strongly confirm that indeed Model-Based Engineering is widespread in the embedded domain. Models are clearly not only used for informative and documentation purposes; they are key artifacts of the development processes, and they are used for, e.g., simulation and code generation. Other widespread uses of significant importance are behavioral and structural consistency checking, as well as test case generation, traceability and timing analysis. While survey respondents reported mostly positive effects of Model-Based Engineering, the data also suggests some common and major challenges for MBE that need further attention. These include effective adoption among developers to reduce effort-intensive activities currently needed to realize benefits of MBE. Furthermore, some challenges concern the specific tools adopted and their interoperation.

In the future, we plan on following-up the results of this study by replicating the survey with a different target group in the embedded domain to validate the identified results. Furthermore, a validation of some effects of the introduction of Model-Based Engineering can be performed by collecting quantitative data in a company which introduces a MBE approach. Tool interoperability was mentioned as one of the key shortcomings, which fits well with the goals of the research project CRYSTAL where we focus on interoperability.

Acknowledgments. The research leading to these results has received partial funding from the European Union's Seventh Framework Program (FP7/2007-2013) for CRYSTAL-Critical System Engineering Acceleration Joint Undertaking under grant agreement No 332830 and from Vinnova under DIARIENR 2012-04304. Further, the authors gratefully acknowledge financial support from FFG Austria for the project in which the above presented research results were achieved. We would also like to thank Aleksander Lodwich, Jos Langen and Leon Bouwmeester for participating in the survey design and contributing to the final study.

References

1. Agner, L.T.W., Soares, I.W., Stadzisz, P.C., Simão, J.M.: A brazilian survey on UML and model-driven practices for embedded software development. Journal of Systems and Software 86(4), 997–1005 (2013), http://www.sciencedirect.com/science/article/pii/S0164121212003160; SI : Software Engineering in Brazil: Retrospective and Prospective Views
2. Anda, B., Hansen, K., Gullesen, I., Thorsen, H.: Experiences from introducing uml-based development in a large safety-critical project. Empirical Software Engineering 11(4), 555–581 (2006), http://dx.doi.org/10.1007/s10664-006-9020-6
3. Baker, P., Loh, S.C., Weil, F.: Model-driven engineering in a large industrial context - motorola case study. In: Briand, L.C., Williams, C. (eds.) MoDELS 2005. LNCS, vol. 3713, pp. 476–491. Springer, Heidelberg (2005)
4. Brambilla, M., Cabot, J., Wimmer, M.: Model-Driven Software Engineering in Practice. Synthesis Lectures on Software Engineering, Morgan & Claypool Publishers (2012)
5. Dobing, B., Parsons, J.: Dimensions of uml diagram use: A survey of practitioners. Journal of Database Management 19(1), 1–18 (2008), http://search.proquest.com/docview/199606166?accountid=10041
6. Fisher, R.A.: On the interpretation of χ^2 from contingency tables, and the calculation of p. Journal of the Royal Statistical Society 85(1), 87–94 (1922), http://www.jstor.org/stable/2340521
7. Forward, A., Lethbridge, T.C.: Problems and opportunities for model-centric versus code-centric software development: A survey of software professionals. In: Proceedings of the 2008 International Workshop on Models in Software Engineering, MiSE 2008, pp. 27–32. ACM, New York (2008), http://doi.acm.org/10.1145/1370731.1370738
8. Grossman, M., Aronson, J.E., McCarthy, R.V.: Does UML make the grade? insights from the software development community. Information and Software Technology 47(6), 383–397 (2005)
9. Hutchinson, J., Rouncefield, M., Whittle, J.: Model-driven engineering practices in industry. In: 2011 33rd International Conference on Software Engineering (ICSE), pp. 633–642 (2011)
10. Hutchinson, J., Whittle, J., Rouncefield, M., Kristoffersen, S.: Empirical assessment of mde in industry. In: 2011 33rd International Conference on Software Engineering (ICSE), pp. 471–480 (2011)
11. Kirstan, S., Zimmermann, J.: Evaluating costs and benefits of model-based development of embedded software systems in the car industry–results of a qualitative case study. In: Proceedings Workshop C2M: EEMDD “From Code Centric to Model Centric: Evaluating the Effectiveness of MDD” ECMFA (2010)
12. Mohagheghi, P., Dehlen, V.: Where is the proof? - a review of experiences from applying mde in industry. In: Schieferdecker, I., Hartman, A. (eds.) ECMDA-FA 2008. LNCS, vol. 5095, pp. 432–443. Springer, Heidelberg (2008), http://dx.doi.org/10.1007/978-3-540-69100-6_31
13. Mohagheghi, P., Gilani, W., Stefanescu, A., Fernandez, M., Nordmoen, B., Fritzsche, M.: Where does model-driven engineering help? experiences from three industrial cases. Software & Systems Modeling 12(3), 619–639 (2013), http://dx.doi.org/10.1007/s10270-011-0219-7

14. Torchiano, M., Tomassetti, F., Ricca, F., Tiso, A., Reggio, G.: Preliminary findings from a survey on the md* state of the practice. In: 2011 International Symposium on Empirical Software Engineering and Measurement (ESEM), pp. 372–375 (September 2011)
15. Wohlin, C., Runeson, P., Höst, M., Ohlsson, M.C., Regnell, B., Wesslén, A.: Experimentation in Software Engineering: An Introduction. Kluwer Academic Publishers, Norwell (2000)