

Evaluating costs and benefits of model-based development of embedded software systems in the car industry—Results of a qualitative Case Study

Sascha Kirstan¹ , Dr. Jens Zimmermann

ALTRAN GmbH & Co. KG, Bernhard-Wiki-Str.3,
80636 Munich, Germany
{Sascha.Kirstan, Jens.Zimmermann}@altran.com

Abstract. Model-based development becomes more and more popular in the development of embedded software systems in the car industry. On the websites of tool vendors many success stories can be found, which report efficiency gains from up to 50% in the development, high error reductions and a more rapid increase of the maturity level of developed functions [1] [2] just because of model-based development. The question arises, why many companies in the industry still develop functions hand-coded, when there seems to be such a high efficiency potential with a model-based development? This article describes the results of a case study which examines the costs and benefits of model-based development of embedded systems in the car industry. Results of the case study are qualitative statements on changes in costs, time and quality in real projects, because of model-based development. Furthermore, a common procedure in the industry is described for the process redesign from hand-coded to model-based development, used tools in the development and best practices for an effective model-based development derived from the case study results.

Keywords: Model-based embedded software development, automotive, cost and benefit analysis, qualitative case study

1 Introduction

In the last 20 years the value chain in the car industry has changed drastically. All car producers and suppliers worldwide have worked on improvements in the area of mechanics, the improvement of quality requirements, and improvements in the logistic area. A lot of the potential in these areas is already exploited. A main differentiation factor turns out to be the electronics area, where a change from hardware to software development is carried out. The meaning electronics will have in the next years has been analyzed by a study of Mercer Management Consulting [3].

¹ Engineering Consultant at Altran Technologies and PhD-Candidate at the University of Technology in Munich. The thesis is supervised by Prof. Dr. Dr. h.c. Manfred Broy (Software and Systems Engineering Chair) and Prof. Dr. Helmuth Krcmar (Information Management Chair)

The study focuses mainly on the question how the cost factors in the development of a car will change until the year 2015 in comparison to the year 2002. In 2015 the costs for the development of electronics will have a value of 35% of the total car production costs. Whereas areas as power train and body have small increases, the costs for the development of electronic systems will be almost tripled. The predicted increases result from a variety of innovations which are being expected in this area. The majority of innovations are realized with embedded systems and especially with software. „90 percent of the future innovations in the car will be based on electronics and from that 80 percent will be realized by software" [4]. However, today's software development has big challenges to master like the growing complexity because of the increasing interaction between the functions, shortened development times for the cars in total versus longer development times for the software and high safety requirements. To master these challenges car producers and suppliers conduct a paradigm change in the software development from hand-coded to model-based development. On the one hand model-based development promises considerable productivity increases, improvements in quality and cost savings. On the other hand, it brings challenges since the use of model-based design results in a major process redesign. The introduction of model-based development influences established development processes, required resources and thereby also the organizational structure. In addition, high investment costs for tools and for training of the employees are necessary.

Although model-based development is used by several car producers and suppliers, no major empirical investigation of the costs and benefits of model-based development has been conducted until today. The decision to use model-based development is a decision with highest strategic importance and therefore a task of the top management of each company. To make their decision they actually need a well-founded cost and benefit analysis. Our aim is to analyse the costs and benefits of model-based development of embedded software systems in the car industry in detail.

Outline: Section 2 focuses on related work. In section 3 the approach for analysing the costs and benefits is described. Section 4 summarises the main results of the case study. In section 5 some hypotheses about the use of model-based development, which are currently validated with a global study, are listed. Section 6 concludes the paper with a summary.

2 Related Work

Statements about the benefits of model-based development in the car industry are quite rare. Most of the statements come from tool vendors, who report about successful projects their costumers have conducted [1] [2]. But a neutral investigation of the costs and benefits of model-based development in the car industry has not been conducted yet. Fieber at al. [5] have conducted an empirical study about the benefits of model-based development, but this study focuses on many different sectors and is only conducted within one single company. Mohagheghi et al. [6] examined the state of the art of quality assurance (QA) methods, the pros and cons of each method and the needed effort to use them. As a result of the analysis they report that model-based

development significantly improves the quality of the automotive embedded software development process. For each investigated quality assurance method like for example model reviews or automated model checks they give statements about the automation degree, the effort and the benefit. They come to the conclusion that due to the relatively high effort required to safeguard the model-based development process, it is still desirable to reduce the effort and increase the effectiveness of the applied QA methods.

Fey et al. [7] have listed best practices for verification, validation and test in model-based design. The paper concludes that model-based design improves a team's ability to deploy a high-quality embedded system on time compared to traditional methods, which rely on verification, validation and testing at the end of the process. Best practices for establishing a model-based design culture can be found in [8]. These best practices shall help companies in adopting model-based design and achieve gains in efficiency in the development process.

Another interesting work is from Mohagheghi and Dehlen [9] who reviewed 25 empirical studies by evaluating reasons for and effects on applying the model driven development paradigm in industrial projects. One result was that more empirical studies have to be conducted to analyse the costs and benefits of model-based development.

3 Analyzing the Costs and Benefits of model-based software development (MBSD)

As presented in Figure 1 our approach to analyse the costs and benefits of model-based development consists of five steps. First of all a theory, which summarizes assumed changes in costs, time and quality because of model-based development was developed. The idea behind the theory is to analyze major differences between a hand-coded and a model-based development process. Therefore a couple of hand-coded and model-based development processes, which are used by car manufactures and suppliers, were analyzed. In the following major differences between a hand-coded and a model-based development process were identified. The identified differences are in our opinion responsible for the changes in costs, time and quality. After identifying the differences in the development process and analyzing their influence on cost, time and quality changes, a case study was conducted to evaluate the theory (step 2). After the case study, different cost models based on the theory were developed (step 3). These cost models have been validated with experts from industry and research. Right now a global study (step 4) is being conducted to get quantitative data on cost, time and quality changes in dependency of the conducted steps in the development process like the degree of modelling and code generation or the use of test activities on models. Last but not least a validation of the cost models with the data of the global study at a car producer (step 5) will be conducted.

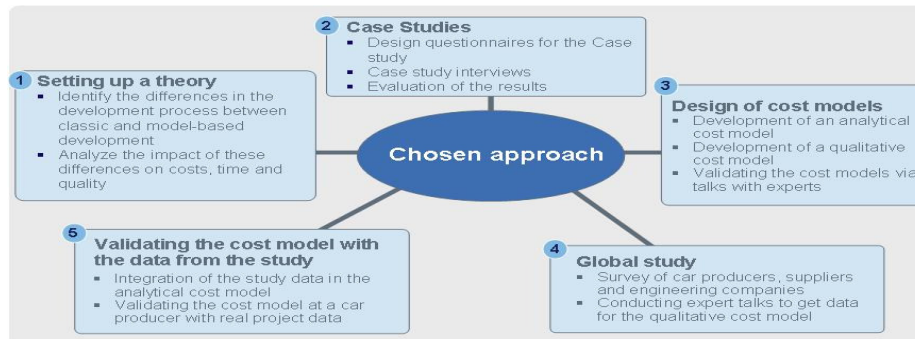


Figure 1: Approach to analyze the costs and benefits of MBSD

In this paper we focus on the results of the case study (Step 2), the performance of the global study (step 4) and the validation of the cost model (step 5) are in progress.

4 The Case Study about Costs and Benefits of model-based development

Between May 2009 and August 2009 twelve² case study interviews with six interviews with car producers, four interviews with suppliers and two interviews with consulting companies have been conducted to bring some light into the costs and benefits of model-based development in the car industry.

4.1 Setup of the Case Study

In each interview we tried to involve a team leader, an engineer and an expert from the procurement, who has insight into the costs. Consequently, almost 30 experts were involved in the case study interviews. The participants were big automotive companies with many years experience in hand-coded as well as in model-based development. Some of them still use both procedures in the development. The know-how of the experts ensures the validity of the case study. Because of the three different groups involved in the case study different viewpoints on model-based development could be worked out. Four different domains have been analyzed in the case study: power train, transmission, energy management and comfort functions.

The main purpose of the case study was to validate the theory (Step 1) about the difference between hand-coded and model-based design which has been developed. Another aspect was to find out which questions on costs and benefits the companies can answer and which questions are rather hard to answer. Therefore generic questions about model-based development and the influence on costs, time and

² Eleven companies have changed to model-based design and one company still develops hand-coded only

quality changes were discussed. This information was needed for the design of the questionnaire for the global study (Step 4).

Figure 2 covers the major questions examined in the case study interviews:

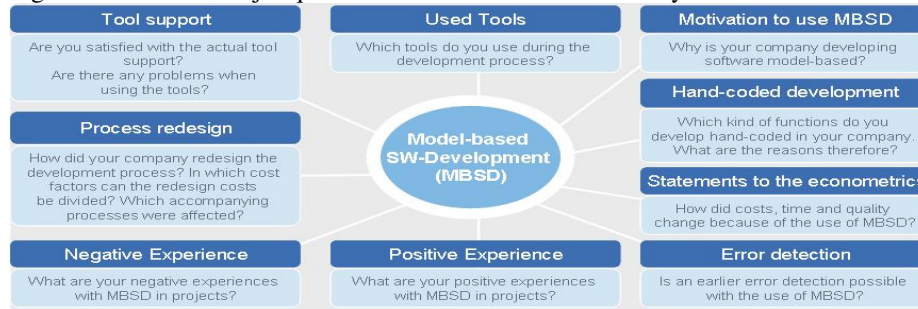


Figure 2: Research questions overview

4.2 Summary of the results

This section describes the results³ of the case study with a structure which is closely related to the questions from Figure 2.

Motivation for the use of model-based development: The motivation to use model-based development can be structured into three points: Company internal motivations, advantages in the development and competitive pressure between suppliers.

Company internal motivations are cost savings in the development, the fear to miss a next generation development technology and the experience that many innovative functions can only be developed via model-based technologies.

The companies hope to achieve the following advantages because of model-based development: Higher automation degree, possibility to detect errors earlier, better communication with colleagues and reduced effort for reusing functions in other car lines.

For suppliers the competitive pressure is a key reason to develop model-based as car producers demand a model-based development.

Positive experiences with model-based development: The experts report on the simplified communication because of the use of function models for the software design. The models provide great support in the communication with other colleagues because of the graphical design. Even colleagues from other departments or domains, who are not familiar with software development, can be involved in the software development because of the use of models. This helps to include extra know-how in the software development.

Another benefit is the possibility to structure the model. Consequently one gets an overview of the whole system while clicking through the function model.

³ This article summarizes the results in a comprehensive manner.

Many experts have made the experience that complex functions can be developed easier with models than with hand-coded design. One main reason for this is the possibility to perform simulations. Especially when using Rapid Control Prototyping (RCP) the experts are able to develop functions where the requirements are unclear. Via trial and error on a real car the requirements of the system are captured. This is especially a benefit, in the development of an innovation with only few clues how to develop it. The RCP-model then serves as a specification.

Function models can be reused over different car lines if a reuse concept is available (see negative aspect reuse). This ensures high cost savings. The possibility to reuse function models is a huge benefit, because in hand-coded software development in a real series project the software is often HW-dependent due to the need for resource optimizations. This makes reuse very difficult. Consequently a lot of hand-coded software has to be developed new for each car line.

The possibility to perform simulations is one of the most important benefits of model-based design. The experts agreed that because of the use of simulations earlier error detection is possible. This ensures cost savings in the development.

Model-based design has a higher automation degree than hand-coded design. With the use of a code generator the code can be generated from the function model. This automation ensures that no programming errors occur. Even the documentation can be generated from the function model. This ensures that the documentation and the generated code are consistent.

Another positive aspect and probably one major reason for the success of model-based development in the car industry is that the majority of the engineers have studied Electrical Engineering or other fields of study, where modelling languages like Simulink [10] or Ascet [11] are more appropriate since they can better express their domain knowledge than C-code (procedural development paradigms).

Negative experiences with model-based development: The experts have made negative experiences with the tools. They report high dependencies from tool vendors. The reason for this is that the tools do not offer interfaces to exchange models among different tools unless inside tools from the same vendor. Consequently if a company has decided for a modelling tool it is advisable to buy the rest of the tool chain from the same tool vendor, to maximise the seamlessness in the development process. Even if standardized interfaces would exist, the experts are still pessimistic about the dependency on tool vendors. The reason for this is that it would probably still be difficult to change for example the modelling tool, because engineers are used to certain modelling blocks. A result of the case study interviews is that it is very hard to convince the developers to apply a new development methodology. Another problem with the tools is that some of them still have a lot of bugs, which are mostly found by users. For the users it is often difficult to figure out if an error comes from wrong modelling or from an error of the tool. Another negative experience is that the tools do not support multi-user functionality. This hampers parallel modelling in large groups.

As described in the positive experiences with model-based development simulation has a high potential to find errors early in the development process. But it is important to keep in mind that there are errors which can't be found via simulation. A well-founded test process after implementation is still needed. Some experts had the hope

that the testing activities after the implementation phase would decrease dramatically in comparison to hand-coded design.

One case study participant was referring to a functionality which was developed model-based but could not be reused. The reasons were lacking planning in the development, no release management, incremental disordered development and too large and unstructured function models. As a consequence all future models had to be developed with a reuse concept. With the reuse concept the company is now able to reuse all their developed functions. The majority of the experts agreed that the design of a reuse concept is vital for reusing function models over different car lines.

Process redesign: Before a company is able to develop model-based a process redesign from hand-coded to model-based development has to be conducted. The experts report high efforts which are needed for the process redesign. They indicate that the main costs for the process redesign are not just costs for tools (although tool costs are a major cost factor) but also costs for defining a new development process, training costs for the employees and the regeneration of hand-coded projects.

In the following an approach for redesigning the development process is described, which the majority of the companies used. The process redesign starts with the evaluation of tools, which are available on the market. After having chosen the tools the design of the new development process can begin. Depending on the chosen tool chain the development process has to be adapted to the company's needs. After designing the development process pilot studies are conducted to evaluate the new development process. With these pilot studies potential for improvement can be identified, for example changes in interfaces or in role allocation. The identified improvements are included in the development process and new pilot studies are conducted to see whether the development process has improved. If the pilot studies have been conducted successfully the companies start to develop a first series product. Depending on the success of this project the new development process is being rolled out in the whole company.

Functions that are still developed hand-coded: Functions that some of the case study participants still develop hand-coded are safety-critical functions and functions with extreme high speed requirements. The reason for this is that the trust in the code generator to generate safety-critical code is too low and hand-coded software is still more efficient than generated software. It is interesting to mention that some participants did not agree and reported that they develop also safety-critical and functions with high speed requirements model-based. Their experience shows that generated code can be as efficient as hand code. The only requirement is to know, which blocks to take to be able to generate efficient code. However, regarding low level software all participants develop basic software and functions for diagnostic hand-coded.

There are some company specific reasons why companies still develop software hand-coded. One reason is that the model-based development process is not established in every department, because the costs for the process redesign (especially the tool costs) are tremendous. Companies simply can not effort to change the development process to model-based development in a whole company even if they want to.

On the supplier side the necessity to develop model-based is sometimes not seen, especially when the functions have already been developed hand-coded for another car line. The reason is that suppliers already have reuse concepts in hand-coded software development and this ensures that they can reuse their already developed system in other car lines.

Used Tools: The used tools in the development process are one indication on how intensive the companies use model-based design so far.

During the analysis of the system requirements and the system architecture no models are being used. For analysing the system requirements Word and Doors are used. In the development of the system architecture the experts use Powerpoint, Visio and Excel. Modelling starts in the software requirements analysis with the use of RCP. Therefore modelling tools like Matlab/Simulink/Stateflow [10] or Ascet [11] are being used. Some companies use the RCP-model as a specification. The majority also uses Word and Doors in addition to RCP for the design of the specification. In the software architecture design a lot of companies still use informal tools like Powerpoint, Visio or Excel. Instead of the informal tools some companies use a modelling tool like Matlab/Simulink/Stateflow or Ascet to design the software architecture. In the SW-Design all companies, which develop model-based, use modelling tools like Matlab/Simulink/Stateflow and Ascet. In the Implementation the code generators TargetLink [12] and Real Time Workshop [13] are used. Used tools for testing are TPT [14], MTest [15], Polyspace [16] and tools developed company-internally. It was interesting to see that no case study participant uses any open source tools.

Frontloading of test activities and changes in error detection: One motivation for developing model-based is earlier error detection, because of the frontloading of test activities via model-based development. 73% of all experts start their test activities with Rapid Control Prototyping (RCP) in the software requirements analysis. 63% of them report they use RCP intensive. All experts use Model in the Loop (MiL)-Tests and 81% use them intensive. This shows that the companies use the frontloading of testing via simulation.

The results of the case study interviews show that the frontloading of test activities influences the time, when errors are detected. The majority of the participants reports less errors which are found during implementation and integration test phase. The reasons for this are the frontloading test activities with RCP, MiL, SiL (Software in the Loop) and PiL (Processor in the Loop) and the higher automation degree with code generation. The experts even report improvements for the maintenance phase, due to the test activities in model-based development as fewer errors in series projects are found. The experts assume that this also has an impact on the maintenance costs because less software updates are needed.

Changes in costs, time and quality: The experts report about small cost and time savings when developing complex innovative systems model-based. The cost savings stem from the possibility to perform simulations (RCP, MiL, SiL, PiL) and from the higher automation with the use of a code generator. When developing functions with small or medium complexity higher costs in the initial development are reported. Cost

savings only result at this complexity level when the functions are reused in other car lines. The statements for the cost and time savings assume that the company has experience in the use of model-based development. For an initial model-based project in a company a considerable cost increase in comparison to an established hand-coded development will occur. Reasons for this are that the engineers have to become familiar with the new development process and still have to learn how to use the tools properly.

A result of the case study is that the change to model-based development alone does not lead to cost and time savings in the development. To use the potential of model-based development a lot of detailed knowledge is needed. One example is the reuse concept that is needed to efficiently reuse functions over different car lines. Many companies fail in model-based design, because they miss a set of best practices.

The companies have made the experience that the highest cost saving potential is the reuse of function models in different car lines. Another cost saving potential is seen in the maintenance. The product has a higher product quality resulting in fewer updates. Furthermore the model can easily be expanded with additional functionality for the facelift of a car for example. Often maintenance activities are conducted by other engineers than the ones who developed it. Model-based development helps the maintenance team to understand the modelled functionality easier than code.

The experts are not agreed when it comes to the question about changes in the quality. The majority of experts reports quality improvements whereas some do not. Their argument is that the product quality of hand-coded software is already very high and difficult to improve. Experts who report quality improvements say that criteria like reliability, maintainability and portability have increased because of model-based design.

The case study interviews validated the theory from step 1 in figure 1 as they confirmed the assumptions from the theory. Thus hypotheses about changes because of model-based development could be derived, some of them are described in chapter 5.

4.3 Best Practices

From the conducted case study interviews Best Practices in model-based development were derived, which can help companies to improve their model-based development.

The majority of the case study participants referred to the added value of model-based development when it comes to reusing the functions in other car lines. One result of the case study is that model-based development alone does not improve the reuse in other car lines unless a reuse concept is applied.

Companies invest a lot of money in making their development process more seamless. This really makes sense, if your development process is quite new. The results of the case study show that at some point (>5 years of MBSE use) the costs for making the process more seamless are extremely high. Instead of spending too much money in developing own tools, defined changeovers in the development process are

more helpful. Research about improving the seamlessness of the development process should be left to research and tool vendors.

The earlier error detection with model-based development has been confirmed by the experts. The results of the case study show that companies that use RCP intensive, find less errors in the testing phase than companies without the use of RCP. If a new system is being developed, we recommend an intensive use of RCP and MiL-Tests, especially for the development of innovative functions.

Another experience many experts made is that it is important that the employees have basic knowledge in software engineering. Most of the function model developers are hardly familiar with software development. If the function models are developed with respect to best practices of software engineering (like modularisation) better products from quality and reuse point of view were developed.

The size of a function model can become a problem. When it becomes too big it takes much time to simulate it and the reuse becomes more difficult. Consequently a best practice approach is to use several smaller function models instead of one big function model.

Another best practice approach is the assortment of the tools. When deciding for a specific tool chain, the companies should keep in mind that it is very difficult to change to another tool chain, after having decided for one tool chain. The reason for this is that the tools do not offer interfaces to exchange models out of one tool into another tool unless the tools are from the same vendor. Consequently the companies become dependent on one tool vendor. Especially for a supplier the assortment of tools is very important. The more common the tools are, which are being used, the easier it is to develop projects for various car producers.

4.4 Threads to validity

The results of the case study show benefits of model-based development. Cost and time savings, the reuse of function models in other car lines and earlier error detection because of model-based development have been confirmed. However threads have to be mentioned which may have affected the validity of the results. These are listed in the following:

1. The number of conducted interviews: To generalize the results of this case study more interviews would have been necessary. But we want to keep in mind that the idea of the case study was to validate the developed theory. For the detailed analysis of the costs and benefits a global study is conducted, which will also give quantitative results. For the purpose of validating the theory twelve interviews with almost 30 interview partners is sufficient from our point of view.
2. The given statements by the experts to cost and benefits are only estimates, because the companies do not have precise data about the changes in costs, time and quality, due to model-based development. The given statements can be influenced by the personal belief of the interviewee about model-based development and / or by pressure within the company that model-based development has to be cheaper (to justify the decision to the top management that the adoption of model-based development was a step in the right direction).

3. Some of the functions were innovative functions which have only been developed model-based. Consequently a real comparison was not possible. But the experts justified their statements with benefits of model-based development like the earlier detection of errors with the use of simulation and the higher automation degree.

5 Hypotheses derived from the results of the Case study and the developed theory

After developing the theory (step 1 in Figure 1) and validating it via case studies more than 20 hypotheses about model-based development were derived from the theory and the results of the case study. These hypotheses will be validated with the data collected via the global study. In the following the five most important of the derived hypotheses are listed:

- *The more intensive model-based development is used, the more savings are expected in cost and time:* We assume that more intensive model-based development will lead to a higher automation in the development.
- *Model-based development is cheaper than a hand coded process even without the reuse of function models in other car lines:* We assume that earlier error detection and the higher degree of automation will result in cost and time savings. Cost savings because of reuse in other car lines is an additional benefit, which ensures further cost savings.
- *The reuse of model-based development in other car lines will lead to high cost savings. 80% of the costs in comparison to a new development can be saved:* We assume that function models can be reused in other car lines much easier compared to hand-written code.
- *If the software is developed model-based fewer errors will be found in the component-, integration- and system test. The errors will be found on model-level (frontloading effect):* The results of the case study have already shown that many errors are found on model level and that the number of errors found in the test phase decreased.
- *Model-based development only pays off in specific domains:* Model-based development is for example hardly used in the area of Infotainment. The reason can be that model-based development does not pay off in this domain.

6 Summary and outlook

The case studies helped to validate and refine the theory (step 1 in Figure 1) on how the costs, time and quality change because of the use of model-based development. Main aspects of the theory like earlier error detection, a higher automation degree, easier reuse in other car lines, and cost savings in the initial development have been confirmed by practitioners. Consequently the case studies helped to bring light into the area of costs and benefits of model-based development in the car industry. Nevertheless only qualitative statements could be made with the case studies. With

the global study (step 4 in Figure 1) we will be able to give also quantitative statements. Since December 2009 we are conducting the global study with car producers, suppliers and engineering companies worldwide. In May 2010 the data of the study will be evaluated. This study gives quantitative information on how intensive the companies in the car industry use model-based development and which changes in costs, time and quality have occurred because of model-based development. In addition the study will also analyze further potential of model-based development to improve its cost-effectiveness in the car industry. With the data collected in the global study the hypotheses from section 5 will be validated and in addition the data will be integrated in one of the developed cost models. With all the steps described in Figure 1, we are quite confident to bring some light into the influence of model-based development on changes in cost, time and quality in the development of embedded software systems.

Acknowledgements:

Special thanks goes to Daniel Ratiu from the chair of Software and Systems Engineering from the technical university in Munich for reviewing this paper and giving valuable input.

References

1. The Mathworks: "User Stories", <http://www.mathworks.com/control-systems/userstories.html>
2. TargetLink: "User Stories", <http://www.dspace.com>
3. Mercer Management Consulting: Study "Future Automotive Industry Structure 2015" (2004)
4. Dr. Lederer, D: "The key to success: Seamless Systems Engineering Process- an urgent task" (2002), <http://www.vectorconsulting.de/portal/medien/cmc/press>
5. Fieber, F., Regnat, N., Rumpe, B.: "Assessing usability of model based development in industrial projects." In CTIT Workshop Proceedings Series WP09-07 ISSN 0929-0672 (2009)
6. Mohagheghi, P., Dehlen, V.: "Where Is the Proof? - A Review of Experiences from Applying MDE in Industry." In: I. Schieferdecker and A. Hartman (Eds.): ECMDA-FA 2008, LNCS 5095, pp. 432-443 (2008)
7. Fey, I.; Stuermer, I.: "Quality Assurance Methods for Model-Based Development: A survey and assessment." In SAE 2007 Proceedings
8. Murphy, B., Wakefield, A., Friedman, J.: "Best Practices for Verification, Validation and Test in Model-based Design" (2008), www.mathworks.com/mason/tag/proxy.html?dataid=11031
9. Smith, P., Prabhu, S., Friedman, J.: "Best Practises for Establishing a Model-Based Design culture" (2007), www.sae.org/technical/papers/2007-01-0777
10. The Mathworks: <http://www.mathworks.com/products/simulink/>
11. Etas: http://www.etas.com/de/products/ascet_software_products.php
12. DSpace <http://dspace.de/ww/de/gmb/home/products/sw/pcgs/targetli.cfm?nv=lb>
13. The Mathworks: <http://www.mathworks.com/products/rtw/>
14. Piketech: <http://www.piketec.com/products/tpt.php?lang=en>
15. DSpace: <http://www.dspace.de>
16. The Mathworks: <http://www.mathworks.com/products/polyspace/>