Mälardalen University Press Licentiate Theses
No. 92

ECONOMICAL VALUATION OF ARCHITECTURAL DECISIONS WITHIN AUTOMOTIVE ELECTRONICS

Håkan Gustavsson

2008

MÄLARDALEN UNIVERSITY
SWEDEN

School of Innovation, Design and Engineering

ISSN 1651-9256
ISBN 978-91-85485-98-7
Printed by Arkitektkopia, Västerås, Sweden

Abstract

Today most innovations made within the automotive domain are driven by electronics. The automotive customers demand new functionality with every new product release and the time-to-market is constantly shortened. Automotive embedded systems are often resource constrained and trade-offs between the system behavior and the resources required is of great importance. The cost sensitive automotive industry has to optimize the use of the system's limited resources, but in the meantime also be flexible.

The system needs to support a large number of vehicle configurations over many years of production. The design decisions are usually based on many factors that pull in different directions such as maintenance, portability, usability etc. The growing complexity of the product and the many uncertain factors create a need for support in the design process.

To better understand this problem decision methods used within an R&D department of an international vehicle manufacturer has been investigated through interviews and surveys. The survey reveals that a majority of the respondents use unstructured methods for resolving decision issues. When respondents were asked about their preferences there was an expressed need for more structured methods.

In this research several existing methods have been surveyed and the methods most relevant to this issue are further described in this thesis. The main contribution of this thesis is an evaluation method using Real Options. The method provides the opportunity to analyze the cost of designing for flexibility to cope with a future growth of the product, based on the estimated value of the future functionality. To improve the usability an evaluation process is defined to aid engineers. This process provides a way of valuing system designs and enables the engineer to think about the future in a systematic manor. To analyze the resource usage within an embedded system a method is proposed on how to evaluate the resource efficiency of functions implemented within an automotive embedded system. The challenge of this work has been to develop methods that are found helpful to the industry and are easy enough to use so that designers are willing to try them again.

Acknowledgements

I would like to express my gratitude to my supervisor Jakob Axelsson for introducing me to the incredible world of research. Thank you for your broad knowledge and ability to structure my thoughts!

I also want to thank my industrial supervisor Jan Sterner for his support and constant flow of references, and my assistant supervisor Christer Norström for his enthusiasm and ideas. I would like to thank Erik Persson for your contribution to this thesis.

For the opportunity and sponsoring I would like to thank the managers of the system development department at Scania for giving me the chance to do research. A special thank to the members of my steering committee who kept me on track. The members of the BESS research group have been great support.

This work has been financially supported by the Knowledge Foundation and the Swedish Agency for Innovation Systems (VINNOVA) and for that I am grateful. I would like to thank my parents for giving me the courage to accept the challenge. Finally, thank you Cissi for being there with me every day.

Stockholm, September 2008

Håkan Gustavsson

List of Included Papers

Paper A

An Industrial Case Study of Design Methodology and Decision Making for Automotive Electronics. Håkan Gustavsson and Jan Sterner, In *Proceedings of the ASME 2008 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference*, ASME, ISBN 0-7918-3831-5, New York, August, 2008.

Paper B

Evaluating Flexibility in Embedded Automotive Product Lines Using Real Options, Håkan Gustavsson and Jakob Axelsson, In *Proceedings of the 12th International Software Product Line Conference*, IEEE, ISBN 978-0-7695-3303-2, Limerick, September, 2008

Paper C

A Framework For The Evaluation Of Resource Efficiency In Automotive Embedded Systems. Håkan Gustavsson and Erik Persson, In *Proceedings of the ASME 2008 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference*, ASME, ISBN 0-7918-3831-5, New York, August, 2008.

Additional publications

Workshop

- Coping with Variability in Automotive Product line Architectures Using Real Options. Håkan Gustavsson and Jakob Axelsson, In *Proceedings of the 11th International Conference of Software Product Line Conference, workshop on Managing Variability for Software Product Lines*, Kyoto, Japan, September, 2007, http://www.softwareproductlines.com/wiki/images/8/88/Gusbavsson_revised.pdf

Conference

- Using Real Options In Embedded Automotive System Design. Håkan Gustavsson and Jakob Axelsson, In *Proceedings of the Conference on Systems Engineering Research*, INCOSE, Redondo Beach, California, April, 2008, http://cser.lboro.ac.uk/CSER08/pdfs/Paper%20142.pdf

Table of Contents

Chapter 1. Introduction

Improvements of the existing product are debated during coffee breaks and in the hallways. Ideas are discussed and eventually a new function is developed. The new function is not part of any project and no budget exists. Instead it is the creation of highly motivated developers and their ambition to improve the product.

To implement this new functionality they need management support which is created through demonstrations. From this point of time everything moves very rapidly. Customers are invited to workshops and it turns out that they are willing to pay for the functionality, but it will probably only be sold in low volumes to a high end segment. The decision is made to introduce the function as fast as possible based on very uncertain information. The function that has been demonstrated is developed using components made for an experimental environment. That does not fit to the current system architecture and is not suitable for production. Management stresses that time-to-market is important and it is assured that the quality of the product will not be affected if implemented as is. Therefore the decision is made to integrate the function rapidly even if the chosen solution does not follow the common design rationale.

This was system development as a short fictive story. It hopefully does not follow the common practice, but it still includes many of the issues that most system developers have experienced in different projects. The solution solves the problem today, but could cause difficulties in the future (a situation referred to as "technical debt" in [7]). The developers did not have the methods available to evaluate and show the economical value of a more long term solution. Such methods would be very useful early in the design process when uncertainty is high. Functions developed in this fashion are likely to be innovative and meeting the demand of the customer. A valuation of the resources early in the design process could reduce the lifecycle cost of the system. If the designs are made in a structured manner, the design decision will be traceable and continuous improvements are more likely to occur. The following sections will introduce the background of the research and explain the research scope.

1.1 Scope: Automotive embedded systems

Today most innovations made within the automotive domain is driven by electronics. A 2001 study made by Mercer Management Consulting and Hypovereinsbank [22] claims that the total value of software will rise from 4% to 13% in 2010. According to a more recent 2006 study made by McKinsey [12] they expect the total value of electronics in automobiles to rise from the current 25% to 40% in 2010. The same study mentions that a large Japanese car manufacture had to recall 160 000 vehicles due to software failure. The automotive customer demands new functionality with every new product release and the time-to-market is constantly shortened. One of the reasons to the high cost of electronics is the large number of Electronic Control Units (ECU) used. The trend is currently changing, but there has been a philosophy within the passenger car industry of “one function – one ECU”. The electrical system must withstand the rapid growth of new functionalities without causing costly redesigns. To manage this task the industry needs methods to choose the best long term design with respect to quality, time-to-market and cost.

Research has shown that decision and evaluation tools are not widely used within the industry [1][10][16][27]. The demands on a method are high usability and the ability to manage the diversity and complexity of an industry.

1.2 Complexity, product and organization

The long life-cycle of automotive products demand that changes to the product can be made with as little impact to the different components as possible. The automotive electronic and electrical (E/E) system is used to implement most new functionalities in vehicles produced today. It is therefore crucial that each of those functions can be implemented without causing large system-wide changes. The building blocks of an automotive E/E system consist of software modules which are embedded into ECUs connected to communication networks. Those networks are usually divided into sub networks and the communications between those are made through gateway ECUs connected to a backbone. Different sensors and actuators are connected to the ECUs depending on the function allocated to the ECU. There are many new functions that are about to be introduced or already introduced that have a large impact on the electrical system of automotive vehicles. To cope with this continuous change the system needs to be designed with the right amount of flexibility.

Examples of functions that require flexibility are Advanced Driver Assistance Systems (ADAS), these systems help the driver in its driver process. Those systems typically use information about the surrounding to increase road safety. This is done by using sensors to identify nearby objects or communication with other vehicles or infrastructure to attain more information. The increased interaction between various components and the wider boundaries of the system increases the complexity and demand flexibility to be easily integrated.

A commercial vehicle must manage to run 300 000 km per year and breakdowns do not just influence the driver, but also the delivery time of the goods it carries. Commercial vehicles have a lot in common with passenger cars, much of the functionality are found in both segments. The passenger car industry has traditionally been adopting new technology earlier. This can be explained by the different needs of the customer. The main purpose of commercial vehicles is transportation of goods, but the transport mission differs from each customer and market.

The architecture of the system must therefore enable a large variation of customer needs without introducing unnecessary changes of common components. Figure 1 shows a simplified view of the architecture supporting all variants of current Scania trucks. The product reaching the customer will only use a subset of components depending on customer choice. A high end long-haulage truck (Figure 2) will therefore be composed by another subset than an all wheel drive construction truck (Figure 3) or a bus (Figure 4).

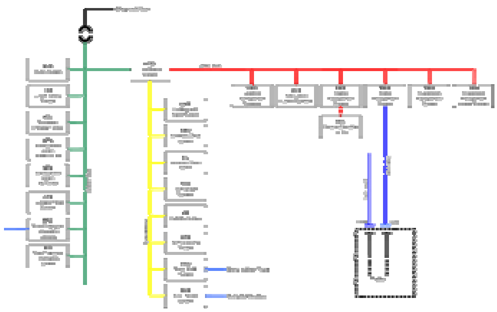

Figure 1. An architecture supporting all variants.

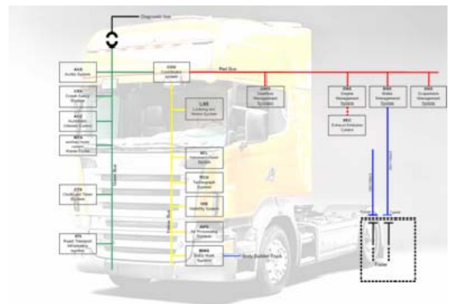

Figure 2. High end long-haulage truck.

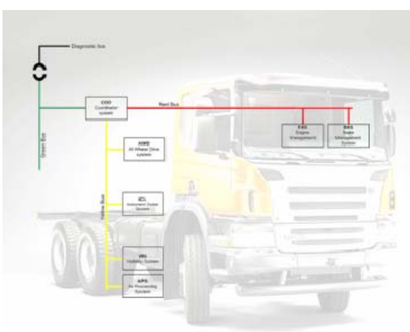

Figure 3. A 6x6 construction truck.

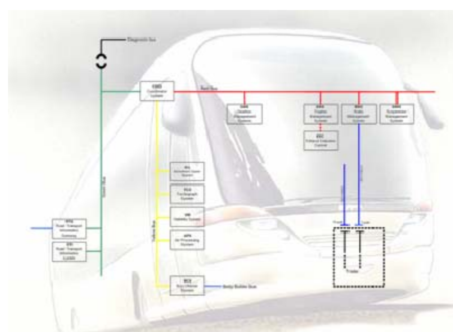

Figure 4. Bus architecture.

The architecture reflects the business goals of a specific commercial vehicle manufacture. The development of the product involves many stakeholders that have an interest of the system during the entire life-cycle. The development process will include stakeholders from departments such as purchase, aftermarket and sales. How those stakeholders are separated within the organization will affect what solutions are chosen and also the system architecture. Changes in one of the concerns business, architecture, process or organization (Figure 5) will therefore have an impact on other concerns [21].

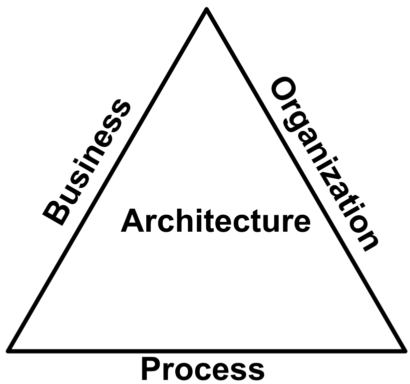

Figure 5. Changes in business, organization or process will have an impact on architecture [21].

1.3 Research scope

Using the definition of the recommended practice stated in IEEE 1471 [14] the studied system is a software-intensive system. The E/E architecture of the system is therefore very important to accomplish the mission of the system.

Software-intensive system: A*ny system where software contributes essential influences to the design, construction, deployment, and evolution of the system as a whole.*

Architecture: *The fundamental organization of a system embodied in its components, their relationships to each other, and to the environment, and the principles guiding its design and evolution.*

The goal of the project is to investigate current methods and develop new improved methods to be used within E/E system development. The developed methods must therefore be found helpful and easy enough to be used again. The usage should lead to continuous improvements trough structured methods which lead to deliberate decisions. The results should be better or equal to what is produced by common sense.

The thesis contributes to the understanding on how design decisions are made and can be improved within automotive E/E system development. The current industry practice is shown using the result from interviews and an extensive survey. A new method and process is developed for improved decision making when making architectural changes in early phases with an emphasis on the balancing of business and technology aspects. The method shows how the value of a flexible design can be estimated and the proposed process shows how it can be accepted by practitioners. To analyze the resource usage within an E/E system a method is proposed for how to evaluate the resource efficiency of functions implemented within an automotive E/E system.

The main contribution is the method for improved decision making when making architectural changes in early phases within the automotive industry. The developed method provides valuable guidance when making system design decisions and more importantly also shows that it can be used and accepted by practitioners.

1.4 Research questions

The overall goal of the project is to develop new or improved methods that can improve decision making when developing embedded systems. To perform this task one needs to investigate how the decisions are made and where in the organization. The first research question therefore states:

Q1 How and where in the organization are architectural decisions made today?

A hypothesis to be tested is whether system design decisions are made by individual developers without any common guidelines. It should also be investigated to what extent formal evaluation methods are used when evaluating design alternatives.

The system is evolving and changes are continuously introduced. To cope with those changes the system needs to be designed with the right amount of flexibility. Research question two aims at developing methods that will aid the architect when making architectural decisions.

Q2 How can one value the flexibility needed to withstand an uncertain future in automotive embedded systems?

The hypothesis is that valuation can be made using Real Options theory.

Common system resources used in an automotive electronic and electrical system have an economical value. The large variation between low and high end applications makes it difficult to optimize the usage of the system resources.

Q3 How can one quantify the resource utilization in automotive embedded systems in the automotive industry?

The hypothesis is that sales figures can be used to analyze how cost-efficiently the resources of the present system architecture are being utilized.

1.5 Thesis outline

The remainder of the thesis is divided into four chapters and an appendix with the published papers. The methods used to study each research question are presented in Chapter 2, where also the validity of the results is discussed. This is followed by Chapter 3 where important related work is presented. The research results and their relation to the appended papers are described in Chapter 4. Finally, in Chapter 5 conclusions are made and future work is proposed.

Chapter 2. Research approach

Various methods have been used to study the stated research questions. The methods used for each of the question are described in this section.

2.1 Method used for research question 1

To study how and where in the organization architectural decisions are made three different methods were utilized, namely document study, semi-structured interviews and a survey.

Decisions are often documented in meeting minutes and similar documents. A document study was therefore carried out to investigate and understand the impact of the formal decision process at levels above the development team.

A semi-structured interview has predetermined questions, but the order can be modified based upon the interviewer's perception of what seems most appropriate. Question wording can be changed and explanations given [25]. A number of semi-structured interviews were conducted to study how decisions were made and with what requirements available. Only decisions made by persons working close to system developers were considered in this part. Before the round of interviews started the questions were tested on a person who just recently held the same position as the respondents. To ensure openness during the interview a second person took notes rather than recording it using audio equipment. The notes were than confirmed by the respondent to avoid misunderstanding. After finishing all interviews the answers were collected and compared side by side.

With the knowledge collected during the document study and interviews a web survey was conducted. This was done to verify the common statements found during the interviews. The result of the collected knowledge is presented in paper A.

2.2 Method used for research question 2 and 3

Research question two and three are similar in the sense that they are answered by exploring the methods used today and investigate if and how they can be improved. The issues are experienced by the author during his years working as system architect. To better understand the problem a comprehensive literature study has been performed. A number of different solutions were identified and compared.

In the case of research question two the development of the method is presented in three papers which have been reviewed and accepted. The first major step was to show why the method is suitable to solve the stated problem, which was published at a workshop (see Additional publications). To verify the usefulness of the method a case study was performed where the method was applied on a real case. "A Case study is a strategy for doing research which involves an empirical investigation of a particular contemporary phenomenon within its real life context using multiple sources of evidence" [28]. The result of the case study was presented at a conference (see Additional publications) and a developed process is presented in paper B.

The methodology to arrive with the answer to research question three has followed the same path, but the case study has not yet been performed on a real case. The case study was instead created using fictitious data, but the relevance of the data was validated by industry experts. The result of the case study is presented in paper C.

2.3 Validity

The largest threats to the validity are found in the method used when answering research questions one. This section is therefore focusing on the method applied during that work.

2.3.1 Construct validity

Construct validity ensures that the studied artifacts can be applied to analyze this exact problem. Threats to construct validity are for instance that the documents available for the document study can be partial, but by triangulation of the information with interviews and web survey construct validity can be ensured. The documentation of the interviews is also reviewed by the informant. The working experience of the author will also help to ensure construct validity.

2.3.2 Internal validity

Internal validity ensures that the conclusions we draw from a case-study are the only possible one and have not been effected by another possible cause. Internal validity is ensured by doing pilot interviews with informants similar to the ones questioned in the study. The questions can thereby be altered to ensure internal validity.

2.3.3 External validity

External validity is the degree to which the conclusions in the study would hold for other organizations and at other times. The major threat to external validity is the degree to which the conclusions would hold for other companies within the automotive industry. The research has been done within one company. It is therefore important to study theory and analyze related work from other areas to prove its validity.

2.3.4 Reliability

Reliability is about minimizing faults and biases in a study and to make the result repeatable. Reliability is ensured by well documented and planned case studies and interviews.

Chapter 3. Related work

"A mathematical model does not have to be exact; it just has to be close enough to provide better results than can be obtained by common sense."

Herbert A. Simon

Nobel prize winner in economics and expert in decision making

To develop new methods which can improve the quality of architectural decisions during the early phases of system development a literature study of existing methods has been performed. The methods found most appropriate for this task are described in this chapter. Examples of where they have been applied or adapted to similar problem are also presented.

3.1 Evaluation methods

In the next section some of the methods available are presented based upon industry surveys [1][27] and a survey of software architecture analysis methods [8].

3.1.1 Pugh

An example of one of the more commonly used formal methods is Pugh's evaluation matrix which was developed by Stuart Pugh in the 1980s [24]. To use the method you first need to decide which criteria to base your decision on and then compare your alternatives to a chosen baseline alternative (Figure 6).The baseline or datum could be any one, but often the current solution or the favorite among the design team will be chosen. The evaluation is then done by comparing every single alternative to the baseline. If the specific criterion is considered better it receives a plus (+), if considered equal it receives a zero (0) and a minus (-) if considered worse than the baseline. The sum of the plusses, minuses and zeros are then calculated. There are also extensions of the method that makes it possible to weight each criterion. Thereby it is possible to rank the alternatives and make a decision.

Alternatives	Baseline	1	2	....
Criteria				
Quality	0	+	0	
Cost	0	-	+	
∑+ ∑- ∑0	0 0 0	1 1 0	1 0 1	

Figure 6. An example of Pugh's evaluation matrix.

3.1.2 Trade-off Analysis

"Trade-off is a decision-making action that selects from various requirements and alternative solutions on the basis of net benefit to the stakeholders" [15]. It is a traditional way to highlight which attributes that should be focused on. Common trade-off pairs could be performance vs. weight or cost vs. quality.

The analytic hierarchy process (AHP) [26] is one way to prioritize design alternatives having multiple criteria. AHP uses a scaled pair-wise comparison between the different criteria The pair-wise comparison has the advantage of being less sensitive to judgmental errors. This theory is used as the basis of the requirement engineering tool Focal Point [17].

A structured method suitable for system development is developed by the Carnegie Mellon Software Engineering Institute [18]. The Architecture Trade-off Analysis Method (ATAM) is a method for evaluating different architectural approaches. The goal of ATAM is to assess the consequences of architectural decisions in light of quality attribute requirements and perform an analysis in a repeatable manner. Each stakeholder has different quality attributes that they consider to be the most important ones. The top level attributes are typically attributes like safety, performance, maintenance and maintainability but the number of attributes can vary from case to case.

A utility tree is created with input from all stakeholders. The utility tree is only constructed by the architects and the project leader and will therefore only show the architects' view of what is important to the system. The next step is to perform a brainstorming of scenarios. The scenarios are made up by all stakeholders. The scenarios are comparable to the leaves of the utility tree.

Each stakeholder is given a number of votes, typically 30% of the total number of scenarios, and then vote for what each stakeholder considers being the most important one. The result from the voting is then compared with the result from the utility tree. If the result is the same, it is quite certain that the most important attributes are considered in the architectural decision. If not, the view of what are the most crucial attributes for a successful architecture differ between system architects and other stakeholders. In this case some kind of reasoning is necessary between the system architects and other stakeholders to conclude the most important parts.

The Cost Benefit Analysis Method (CBAM) is an extension of the ATAM and is also developed by the Carnegie Mellon Software Engineering Institute [19]. It uses the quality attributes from the ATAM but also consider cost when reasoning around the most suitable architecture.

3.1.3 Options theory

Using options theory is one approach to deal with the high level of uncertainty when making design decisions in the early phases. The theory derives from finance where an option is the right but not the obligation to exercise a feature of a contract at a future date [13]. An option has a value because it gives its owner the possibility to decide in the future whether or not to pay the strike price for an asset whose future value is not known today. An option therefore provides a right to make the costly decision after receiving more information. There are two different types of options, American and European. A European option may only be exercised at maturity opposite to an American option that can be exercised any time until the exercise date.

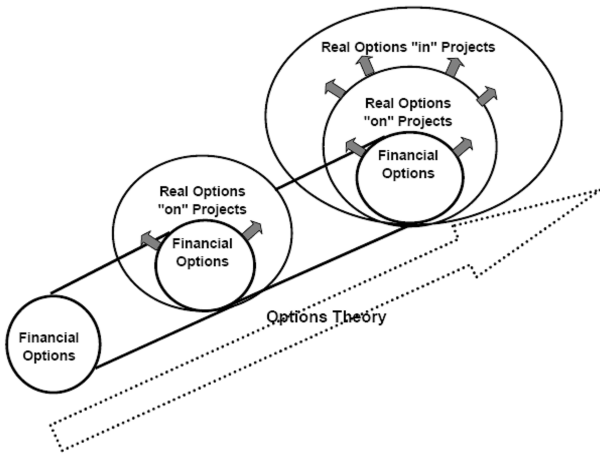

Figure 7. The evolution of option theory. [23]

Real Options could be seen as an extension of financial option theory to options on real (non-financial) assets [13]. Copeland [6] defines a real option as: "the right, but not the obligation, to take an action (e.g. deferring, expanding, contracting, or abandoning) at a predetermined cost called the exercise price, for a predetermined period of time - the life of the option."

Since the 1990s Real Options theory has started to be utilized within the field of engineering to manage the risk of uncertain design decisions. In 2001 de Neufville [23] coined the expressions Real Options "in" and "on" projects (Figure 7). Real Options "on" projects treat the enabling technology as a black box while Real Options "in" projects are options created by changing the actual design of the technical system. Real Options on projects provide a more accurate value of the project and Real Options in projects support the decision on what amount of flexibility to add. "Real Options on projects are mostly concerned with an accurate value to assist sound investment decisions, while Real Options in projects are mostly concerned with go or no go decisions and an exact value is less important" [29].

3.2 Decision support in automotive embedded system design

This section will present the most relevant related work on available decision support within automotive embedded systems.

3.2.1 Decision levels

Architectural decisions are made when selecting components and allocating them to subsystems which then are combined into a system. The decisions can be made on different levels which have various impacts and predictability. In [9] they are grouped into three levels; top-level, high-level and low-level (Figure 8). Top-level decisions concern the quality and function attributes and have the largest impact. Choosing architectural patterns and technologies are found to be high-level decisions. The most predictable decisions are those concerning the hardware architecture and function mapping. The impact of the decision will vary depending on how decoupled software is from hardware. The paper does not discuss on what organizational level the decisions are made, but a hypothesis is that they do not necessary correlate to the organizational hierarchy.

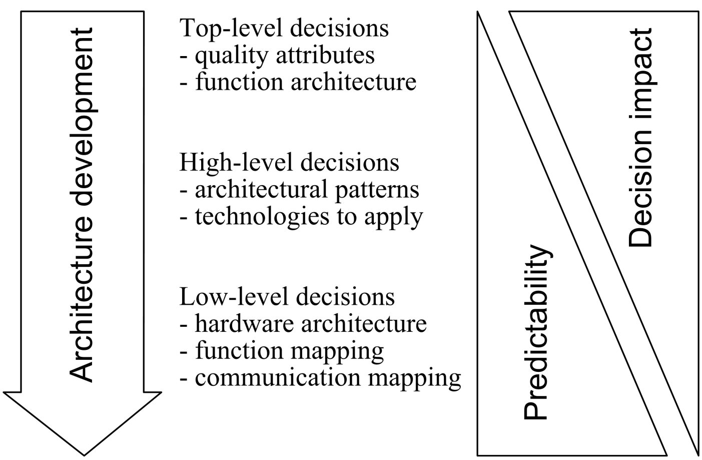

Figure 8. Decisions made during the development of the architecture will have different impact and the outcome will be more or less predictable [9].

3.2.2 Analyzing architectural attributes

When designing an automotive E/E system there are many different attributes to consider. The most important architectural attributes are according to [11]:

- Functional requirement
- E/E Components (sensors, actuators, etc.)
- Energy management (energy storage, control strategy,
- Communication (bus-topology, baud rate etc.)
- Electronic Control Units (housing, operating system, processor, etc.)
- Wiring harness (connectors, cable length, etc.)

All of those attributes will influence the business aspects of the product. Many of the important attributes mentioned in [11] are used in the cost model presented in [2]. The cost model uses probability distributions to handle the uncertainties available when making architectural design decision during early phases.

3.2.3 System as a marketplace

The value of the system resources can also be traded at run time. This technique is utilized for solving large-scale problems using so called computational grids [5]. The marketplace consists of virtual enterprises connected over the internet. It is used for sharing an enormous amount of resources which are geographically distributed. The common resource which is traded is usually computing power.

Similar theory is utilized to optimize vehicle energy management where the common traded resource is power. The pricing depends on the actual driving condition [4]. During braking the vehicle energy management uses the excess energy from the vehicle inertia in a useful way. For instance the generator could be set to a stronger charging of the battery.

This approach shows how economical reasoning can be used to make decisions even at run-time.

3.2.4 Balanced scorecard

Balanced scorecard was developed at Harvard business school in the early 1990's to aid strategic management. It is used to view the organization from four different perspectives (Figure 10). With the right measure for each of them, one should be able to balance the decisions according to the company strategy and vision.

Larses [20] used the original balanced scorecard to balance the important perspectives in system design of the complete E/E system (Figure 9). By using design structure matrix (DSM) he found a way to measure modularity of different solutions. Performance was measured by analysis of resource utilization. Dependability was measured by looking at the number of connection points. Cost was calculated with regard to factors like cable length, component cost etc.

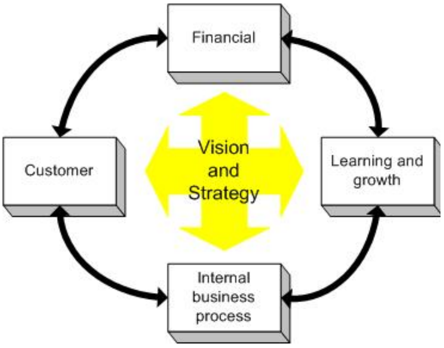

Figure 9.

Original balanced score card.

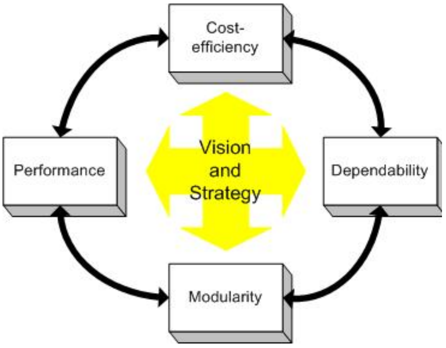

Figure 10.

System design using the balanced score card.

Chapter 4. Research Results

This chapter summarizes the research results and relates the research questions to the individual papers included in this thesis.

4.1 Relation between appended papers

In total three papers are appended, paper A, B and C. Paper A studies how design decisions are made within system development by interviews and an extensive survey. It was found that most decisions made by developers are based on intuition and the usage of structured methods is very low. Three improvements were suggested to respond to the identified problems.

Paper B presents how flexibility can be valued through an evaluation method. The value is calculated using various input parameters such as implementation cost, system lifetime and uncertainty of customer demand. A process is presented to make it possible for practitioners to utilize the method. The paper also presents how the process is applied on a real case.

The cost of a distributed function is often seen as the sum of the hardware resources used. Issues such as software development costs and maintenance costs have historically been neglected. Paper C proposes a method to evaluate the resource efficiency of functions implemented through the embedded system. The resource utilization can thereby be quantified.

Figure 11 shows the relation between the appended papers. Paper A studies how design decisions are made within a large organization and proposes areas of improvements. Paper B presents how flexibility can be valued when making design decisions. Paper C presents a method to quantify the resource utilization to support a cost-efficient development.

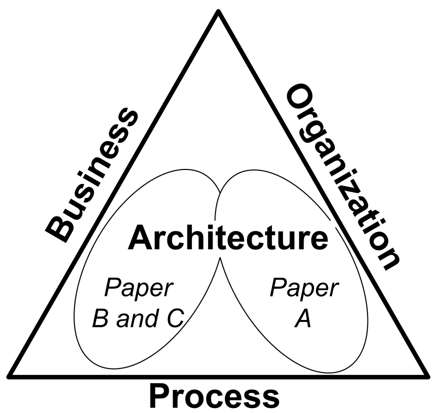

Figure 11. The papers relation to important aspects of an industrial context.

4.2 Paper A: An Industrial Case Study of Design Methodology and Decision Making for Automotive Electronics

The results of this paper are based upon interviews and a survey made at Scania during 2007 together with Jan Sterner. The aim of this paper is to study research question one. The paper was accepted and presented on the International Conference on Design Theory and Methodology (http://www.asmeconferences.org/IDETC08/).

The main objective of this study has been to examine the decision making process within the R&D department of an international vehicle manufacturer. The aim is to gain knowledge of where the organization stands today and how the decision process can be further optimized. One target is to describe decision paths and the decision criteria used today, but also to suggest an improved process. Part of the work is to identify the company best practice and also to develop a checklist to be used at gate meetings.

The complexity of the system is high; each ECU of the E/E system is arranged in families, each containing a number of ECU executions. Further there are many versions of each execution. Three system executions of different system families were studied (engine management system, gearbox management system and the main coordinator gateway).

The research topic has been divided into three issues:

1. Who make decisions that influence the design of the electric/electronic system in Scania vehicles?

2. What are the bases for decision?

3. How reliable is the underlying data?

The study has been focused on decision making in the development team. Decisions made by personnel close to the product are regarded as informal and less documented whereas the formal process is considered as well known. Interviews were carried out within the former category while the latter was investigated through a web survey.

4.3 Paper B: Evaluating Flexibility in Embedded Automotive Product Lines Using Real Options

The results presented in this paper are an evolution of two papers presented at the workshop on Managing Variability for Software Product Lines at SPLC 2007 and at the Conference on Systems Engineering Research 2008.

The aim of this paper is to study research question two and was written together with Jakob Axelsson. The paper was accepted and presented at the Software Product Line Conference 2008 (http://www.lero.ie/SPLC2008).

The automotive customers demand new functionality with every new product release and the time-to-market is constantly shortened. The automotive embedded systems are characterized by being mechatronic system which adds complexity. The systems are often resource constrained and trade-offs between the system behavior and the resources required is of great importance. The decisions are usually based on many factors that pull in different directions such as maintenance, portability, usability etc. The complex system and the many uncertain factors create a need for support in the design process. In this paper the use of Real Options is evaluated on a hypothetic but realistic case taken from the automotive industry. The case shows how real option valuation provides additional guidance when making system design decisions. Real Options provide the opportunity to analyze the cost of designing for future growth of a platform, based on the estimated value of the future functionality. The value of a flexible design can thereby be quantified making the trade-off between short and long term solutions more accurate.

4.4 Paper C: A Framework For The Evaluation Of Resource Efficiency In Automotive Embedded Systems

The aim of this paper is to study research question three and is based on the master's thesis by Erik Person, whose thesis work was supervised by the author. The author contributed as supervisor of the thesis work, the literature study and the analysis and conclusions were made in cooperation with the co-author. The paper was accepted and presented at the International Conference on Design Theory and Methodology (http://www.asmeconferences.org/IDETC08/).

Scania trucks and buses are produced with a common product platform of modular components in order to keep the product cost low, a high level of quality and to offer the customer a wide array of choice. A significant part of current and future functionality will be implemented through the use of electronics, consisting of modular components with the same demands as traditional mechanical components. The cost of electronics has risen significantly over the last years. A reduction of the product cost of the electronics system would thus have a substantial impact on the total cost of the vehicle. This paper discusses the resource utilization of embedded systems in the automotive industry. Traditionally, the major cost driver – or resource input – has been regarded as the hardware cost. Issues such as software development costs and maintenance costs have historically been neglected. In order to address this, the paper embraces the more comprehensive view that a resource can be regarded as anything which could be thought of as a strength or weakness of a given firm. In this paper the major drivers of resource consumption are identified. The work has included several interviews with employees in order to find empirical data of the embedded systems in vehicles. A method is developed to evaluate the resource efficiency of user functions implemented through the embedded system. By the use of Data Envelopment Analysis – which has proven to be a useful method – the resource utilization of six user functions is evaluated.

Chapter 5. Conclusions and Future Work

The goal of this research has been to investigate current methods and develop new improved methods to be used within E/E system development. The conclusions and suggestions of future work are divided and presented together with the corresponding research question.

Q1 How and where in the organization are architectural decisions made today?

The study of current industry practice showed that formal evaluation methods are not often used, but developers are willing to use them. The developers should therefore be given education in the use of structured methods. Decisions are often made by individuals and there exists good guidelines and rules of thumb, but they need to be updated and more widely spread within the organization. This can be achieved if the roles of the expert within the organization would be strengthened.

Developers were very willing to express their thoughts on how development should and should not be done. When listening to their complaints it is notable that the quality of the developed systems is even used as a sales argument and the product as a whole is considered to be state of the art. The success factors found within this work are well motivated engineers working in an open minded climate.

The large scope of this study makes one stumble into interesting topics available for further investigation. The most important one from the study of current industry practice is to investigate the technical career and the role of the expert. This could be done by interviews and surveys. To improve and follow up the use of design know-how and guidelines is also an important topic.

Q2 How can one value the flexibility needed to withstand an uncertain future in automotive embedded systems?

The flexibility of different system designs can be valued using a method based on the theory of Real Options. This method has been tested with positive result on one case study. A process has been developed using the lessons learned from this case study. The economical valuation connects the business goals to the architectural decisions. The real option approach could when fully developed provide not only evaluation but also prediction of future needs. The process provides a way of valuing system designs and enables the engineer to think about the future in a systematic manor.

There is research needed to find ways on how to calculate volatility based on available data. There is also a need to make case studies focusing on the acceptance of the process in the developing organization. Also, it would be interesting to dig deeper into the software aspects of an embedded system, and analyze the value of building more flexible software, e.g. based on frameworks like AutoSAR.

Q3 How can one quantify the resource utilization in automotive embedded systems in the automotive industry?

Different methods that can be used to quantify the resource utilization in automotive embedded systems have been studied. The major drivers of resource consumption were identified through interviews. Finally a resource utilization framework was created using the method of Data Envelopment Analysis. The framework was tested on six fictitious user functions with different sales figures and was found to be helpful when evaluating historic design decisions. The method indicates how cost-efficiently the resources are being utilized.

The most interesting future work for the proposed framework on resource utilization would be to make a case study on existing user functions. An extension would also incorporate more quantitative data in the analysis. For instance the framework of COCOMO [3], may be useful in order to quantify software development costs and software maintenance costs.

References

[1] Araujo, C.S. The utilization of product development methods - A survey of UK industry. In Journal of Engineering Design 7, nr. 3, pp. 265-277, 1996

[2] Axelsson, J. Cost models with explicit uncertainties for electronic architecture trade-off and risk analysis. In Proc. 16th INCOSE Annual International Symposium, 2006

[3] Boehm, B. et.al. Software cost estimation with COCOMO II. Upper Saddle River, New Jersey, USA: Prentice Hall, 2000.

[4] Boucharel, P. and Graf, H.M.; Optimizing vehicle energy management utilizing Power Trading Concept, VDI Berichte Nr. 1907, In 12th International Conference and Exhibition Electronic Systems for Vehicles, Germany, October, pp. 347-358, 2005

[5] Buyya, R et.al. Economic models for resource management and scheduling in Grid computing. Concurrency and computation: practice and experience, No. 14, pp. 1507-1542, 2002

[6] Copeland, T. and Antikarov, V. Real Options: A Practitioner's Guide. Texere Publishing Limited, New York, 2001

[7] Cunningham, W. The WyCash portfolio management system. In: Conference on Object Oriented Programming Systems Languages and Applications, pp. 29-30. ACM Press New York, NY, USA, 1992

[8] Dobrica, L. and Niemela, E. A Survey on Software Architecture Analysis Methods. In: IEEE Transactions on software engineering 28 (2002), pp. 638-653

[9] Florentz, B. and Huhn, M. Architecture Potential Analysis: A Closer Look inside Architecture Evaluation, In Journal of Software, vol. 2, no. 4, pp. 43-56, October 2007

[10] Frost, R. B. Why Does Industry Ignore Design Science? Journal of Engineering Design 10, no. 4, pp. 301-304, 1999

[11] Gemmerich R. et al. A Comprehensive Approach for Generation and Optimization of Automotive Electric and Electronic Distribution Systems. VDI Berichte Nr. 1907 In: 12th International Conference and Exhibition Electronic Systems for Vehicles, Germany, October, pp. 597-608, 2005

[12] Hoch, Detlev et al. The race to master automotive embedded systems development. Germany : McKinsey Company, Automotive and assembly sector business technology office, 2006

[13] Hull, J. C. Options, Futures, and other derivative securities. 2nd Edition. Prentice Hall International Editions, New Jersey, 1993

[14] IEEE Architecture Working Group, IEEE Recommended Practice for Architectural Description of Software-Intensive Systems, IEEE Std 1471-2000, IEEE, 2000

[15] ISO/IEC 15288 International Standardization Organization, Information Technology - Life Cycle Management - System Life Cycle Processes, International Organization for Standardization, Switzerland, Geneva, 2002

[16] Janhager, J. et al. A.: Survey on Product Development Methods, Design Competencies, and Communication in Swedish Industry. In Proc. TCME, Tools and Methods of Competitive Engineering. Wuhan, China, 2002

[17] Karlsson, J. and Ryan, K. A Cost-Value Approach for Prioritizing Requirements, In IEEE Software, vol. 14(5), pp. 67-74, 1997

[18] Kazman, R. et al. ATAM: Method for Architecture Evaluation, CMU SEI Technical Report CMU/SEI-2000-TR-004, Carnegie Mellon Software Engineering Institute, 2000

[19] Kazman, R. et al. Making Architecture Design Decisions: An Economic Approach. Technical Report CMU/SEI-2002-TR-035, Carnegie Mellon Software Engineering Institute, 2002

[20] Larses O. Architecting and Modeling Automotive Embedded Systems. Ph.D. Thesis. Dept. of Machine Design, Royal Institute of Technolog, Stockholm, Nov. 2005

[21] Linden, F. et al. Software Product Family Evaluation, in Proceedings of the 3rd International Conference on Software Product Lines, pp. 110-129, Boston, 2004

[22] Merecer Management Consulting and Hypovereinsbank, Studie-Automobiletechnologie 2010, Munchen, August, 2001

[23] Neufville, R. Real options: dealing with uncertainty in systems planning and design. In Proc. 5th. International Conference on Technology Policy and Innovation, Technical University of Delft, Netherlands, 2001

[24] Pugh, S. Total design: integrated methods for successful product engineering. Wokingham: Addison-Wesley, 1990

[25] Robson, C. Real World Research, Second edition, Blackwell Publishers Ltd., Oxford, 2002

[26] Saaty, T. L. How to make decisions: The analytic Hierarchy Process. In Journal of Operational Research, vol. 48, pp. 9-26, 1990

[27] Salonen, M. and Perttula, M. Utilization of concept selection methods – a survey of Finnish industry. Helsinki University of Technology, Helsinki, Finland, 2005

[28] Yin, R.K. Case Study Research: Design and Methods, Sage Publications Inc, 2003

[29] Wang, T. Real Options in Projects and Systems Design-Identification of Options and Solution for Path Dependency. Ph.D. thesis, Massachusetts Institute of Technology, Cambridge, 2005

Paper A

An Industrial Case Study of Design Methodology and Decision Making for Automotive Electronics

Håkan Gustavsson
Scania CV AB and Mälardalen University
hakan.gustavsson@mdh.se

Jan Sterner
Scania CV AB
Jan.Sterner@scania.se

Abstract

The growth rate of R&D activities in automotive industry brings an increased need for transfer of design knowledge. This, in combination with growing complexity of the product puts new demands on the decision process. In this paper, decision methods used within the R&D department of an international vehicle manufacturer has been investigated through interviews and surveys. The main focus has been to identify and analyze methods used by the individual roles within different development teams. The survey reveals that a majority of the respondents use unstructured methods for resolving decision issues. When respondents were asked about their preferences there was an expressed need for more structured methods.

Among these, two methods are elaborated that are well established within the product development process: expert support and guidelines, but also on methods training in general. A third conclusion is to redirect the current decision process to build on more structured methods through training.

This work has contributed also by identifying the company best practice. The long term goal is to have all development teams adopt one common development process at the team level.

1 INTRODUCTION

To stay competitive in the automotive industry vehicle manufacturers are forced to release new models more often. At the same time the product portfolio must be further diversified in order to satisfy individual customer demands. The shorter development cycle and increased number of concurrent models brings an increased need for transfer of design knowledge.

Today most innovations made within the automotive domain are driven by electronics. A study made by Mercer Management Consulting and Hypovereinsbank in 2001 [15] claims that the total value of software in cars will rise from 4% to 13% by 2010. According to a more recent 2006 study made by McKinsey [7] they expect the total value of electronics in automobiles to rise from the current 25% to 40% by 2010. One of the reasons for the high cost of electronics is the large number of Electronic Control Units (ECU) used. The trend in the car industry is currently changing, but there has been a philosophy of “one function – one ECU”.

1.1 AUTOMOTIVE ELECTRIC AND ELECTRONIC SYSTEM DEVELOPMENT

The embedded software development within the automotive industry is not very different from other embedded systems. The automotive embedded systems are characterized by being a mechatronic system which adds complexity. The systems are often resource constrained and trade-offs between the system behavior and the resources required is of great importance. Cost, time-to-market and quality are the most important factors. The growing number of interconnected sensors, actuators and functions allocated to different ECUs has led to a need for standardization to simplify the development process.

The studied company is an internationally well known vehicle manufacturer of commercial vehicles and should be representative for the rest of the industry. It has managed to achieve sustainable and profitable growth and it is interesting to study which factors that contribute to the success. The current and future growth rate of its R&D organization (Figure 2) brings an increased need for transfer of design knowledge. This, in combination with growing complexity of the product puts new demands on the design decision process. In future development projects it will be necessary to handle an increasing amount of information. It is critical that the decision making process is up to date with the fast changes of the product design. The growing demand for decision support tools is of special interest.

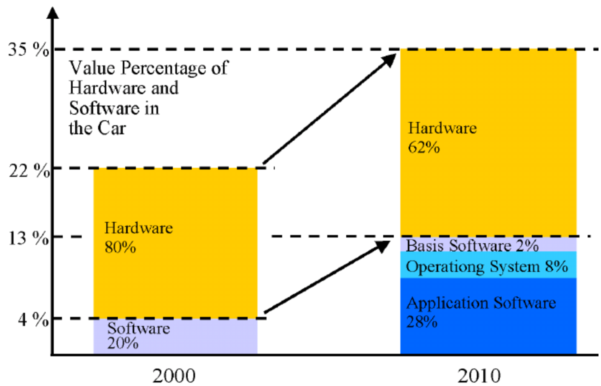

Figure 1 Value of hardware and software in cars [15]

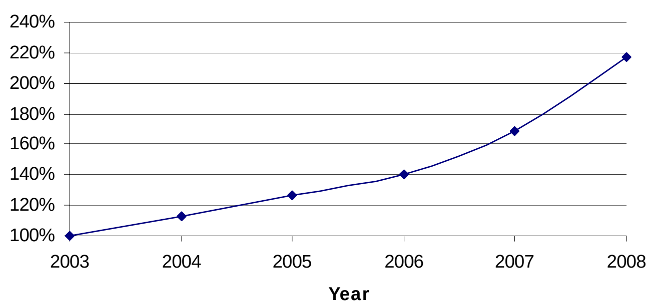

Figure 2 The increase of number of employees working within the system development organization expressed as a percentage compared to 2003.

1.2 OBJECTIVE AND DELIMITATIONS

The main objective of this study is to examine the decision making process within electric/electronic system development. The aim is to gain knowledge on where the company stands today and how the decision process can be further developed.

The studied manufacturer purchases about half of the systems from external suppliers. The study requires detailed information about the design process and relevant results are best achieved by focusing on systems that are internally developed. Three systems were studied; the engine management system, gearbox management system and the main vehicle controller. Those systems were chosen because of their differences in size of the developing organization.

The overall decision process includes roles at every organizational level. However, in this study only decisions made by persons that work as, or close to system developers, are considered. The decision process at levels above the development team is not regarded.

1.3 PAPER OUTLINE

In the next section, the method used to obtain information about the design decision process is described. This is followed by a literature survey on structured methods that are available today. Empirical findings are then analyzed according to a general decision model, in order to identify current weaknesses and potential improvements of the design development process. Finally three conclusions are drawn from the analysis

2 METHODOLOGY

The information needed in the study was gathered in three steps.

1. A document study was carried out to investigate and understand the impact on the development team of the formal decision process at levels above the development team.
2. A number of interviews were conducted to study how decisions are made and with what requirements available. Only decisions made by persons working close to system developers were considered in this part.
3. With the knowledge collected during the document study and interviews a web survey was conducted. This was done to verify the common statements found during the interviews.

These steps are described in more detail in the next subsections.

2.1 DOCUMENT STUDY

Product design specifications are stored in databases and are historic documents. Decisions are often documented in meeting minutes and similar documents. Those documents are therefore a valuable source of information which the content analysis of the document study will gather. All minutes of the monthly technical specifications meeting have been reviewed on decisions with a direct impact on the electrical system. The structure and content of the main engineering change order (MECO) and engineering change order (ECO) for one of the studied systems have been reviewed.

2.2 INTERVIEW

Three different roles within system development where investigated using semi-structured interviews. The system owner is provided with the overall responsibility of the development of the ECU hard and software. The object leader is responsible for planning and follow-up of all development activities within the system. The object leader allocates resources necessary to reach the main deliverables requested at each phase transition. The function owner supervises development of the many part functions allocated at different ECUs. The development engineer is responsible for code in different part functions belonging to one system and, thus, is working closely with several function owners.

Each respondent was interviewed during 60-90 minutes. Answers were recorded in writing without the use of audio equipment. Respondents were assured anonymity and received their answers in writing after the interview. The first part contained process related questions in the form of a case study. The respondent was confronted with four cases, each designed to represent a scenario with a (small / large / expensive / difficult) decision. The four cases were visualized on paper. The second part contained questions related to design methods. The interviews were summarized together with feedback from the respondent.

2.3 SURVEY

The interviews were supplemented by a web survey. The survey, based on a subset of the interview questions, was sent to the corresponding roles (system owner, object leader, function owner, and developer) within all systems. The web survey was sent to 150 engineers at R&D. They were asked to answer the survey within one week. After the first week 36 persons had responded. The final response frequency amounted to 64 persons. Of these 28% worked as system owner, 21% as object leader, 40% as developer, and 11% as function owner.

Results from the web survey were analyzed in Statistica [21]. Correlations were investigated through regression analysis and filtered at the 90% significance level.

2.4 VALIDITY

Construct validity ensures that the studied artifacts can be applied to analyze this exact problem [25]. Threats to construct validity are for instance that the documents available for the document study can be partial. By triangulation of the information with interviews and the web survey construct validity can be ensured. The documentation of the interviews is also reviewed by the informant. The working experience of the authors will also help to ensure construct validity.

Internal validity ensures that the conclusions we draw from the web-survey is the only possible one and have not been affected by another possible cause [25]. Internal validity is ensured by doing pilot interviews with informants similar to the ones questioned in the study. The questions can thereby be altered to ensure internal validity. The respondent should not be biased by how questions are phrased. During the interviews it was important to avoid the use of formal words like “method” or “process”. Respondents were rather asked to describe their process in their own words. Two questions were rephrased after the third interview.

External validity is the degree to which the conclusions in the study would hold for other organizations and at other times [25]. The major threat to external validity is the degree to which the conclusions would hold for other companies. A major part of the research was done within the company. It is therefore important to study theory and analyze related work from other areas to prove its validity.

Reliability is about minimizing faults and biases in a study and to make the result repeatable [25]. Reliability is ensured by well documented and planned interviews.

3 THEORETICAL FRAMEWORK

Decision making under uncertainty is influenced by a number of factors [11], and some of them lead to less rational decisions. One of many social psychology factors is the anchoring effect. The anchoring effect describes how your initial guess or starting point relates to your final answer. The mind creates an imaginary point of reference.

Another problem when making decisions is that one might be influenced by previously made decisions. This might lead to a bias in the review of some alternatives. An important factor in this case is the so-called sunk cost, which describes how our decisions are influenced by previously made investments in such a way that one bad investment decision is often followed by a new bad one in order to justify the first decision.

To make a decision many types of information need to be present. The model presented in Figure 3 shows how the different types of information can be related. An issue states the problem encountered which is defined and limited by different criteria. A criterion limits the design space and the number of feasible alternatives that can address the issue. During the evaluation the alternatives are measured relative to the criteria and leads to a decision. The evaluation of alternatives can be done by using one of the structured methods presented in next section.

Figure 3 A model for the information flow of decision-making [23].

4 RELATED WORK

There are many structured methods available. Some examples mentioned in industry surveys [2] are Pugh evaluation matrix [18] and the analytical hierarchy process (AHP) [19]. A comparison of methods specialized towards software architecture analysis [5] has found Architecture Trade-off Analysis Method (ATAM) [12] to be the most suitable. The Cost Benefit Analysis Method (CBAM) [13] is an extension of the ATAM and uses the quality attributes from ATAM but also consider cost when reasoning around the most suitable architecture.

In a study of 46 companies made 2005 in Finland [20] it was shown that the most commonly used (76%) concept selection method was concept review meetings. About half of the companies used informal methods like checklists, intuitive selection or expert assessment. Less than one out of four companies responded to use one or several formal methods. The explanation to the low usage of formal methods is that they are hard to fit into the industry. The proposed solution is to present success stories and to further investigate the need of the industry. A survey has also been made on the UK industry [2] and showed similar results. An article published in Journal of Engineering Design attempts to answer the question, why does industry ignore design science [6]. The article claims that industry solves problems by using the knowledge of experienced engineers, which is often faster than using a structured method. One of the presented answers is that many structured methods require information which is often not present or very resource consuming to generate.

Ken Hurst [8] presents the following reasons why a structured method should be used:

- Time wasted in pursuing wrong alternatives to the detail design stage is avoided.
- Causing decision-making to be visible helps ensure the process is repeatable.
- The ability to evaluate the thought processes of others is developed.
- The designer can defend decisions made in discussions with managers or clients.
- A designer with no previous experience can carry out a sensible evaluation of alternative concepts.
- The process of concept selection stimulates new concepts or encourages combination of concepts.

Ulrich and Eppinger [24] present a similar list of benefits and emphasize that the use of a structured method provides customer focus and a more competitive design. The way structured methods encourage knowledge transfer is also stressed by Liker [16]. He points out the difficulty to transfer tacit knowledge compared to explicit knowledge. Explicit knowledge such as mathematical equations and historical facts are often more easy to store. Tacit knowledge is often more diffuse similar to what is taught through apprenticeship. Toyota creates their learning network through activities such as technology demonstrations, checklists, know-how databases, mentoring and lessons learned [16].

5 EMPIRICAL FINDINGS

In this section, results from the document study and the investigation on the process in the development teams are presented. Together with the document study the product development process is described.

The current official process for decisions in the company is described in an internal standard. In this document it is stated that "The success of the assignments in the product development process requires among other things a clear decision structure." It is noted that the embedded software development process is slightly different. Here the important principle is expressed as "in each development stage, the software shall be documented in such a way that a new programmer is able to develop the next release further". An earlier development model for embedded software ("checklist for designers") however, is now replaced by a general checklist for designers.

Trade-off curves, guidelines, checklist are used in different departments and projects. Trade-off curves are being used frequently in part of the development process. Much knowledge is stored in guidelines and a new effort has recently started to gather all design know-how into one database.

There are also various training courses offered; decision making and problem analysis, modularization, ECU-system and functional concept design.

General aspects of the decision process of the company is defined and well documented as described. Yet, knowledge about the process is poor. Insufficient knowledge has two interpretations, either confidence in the product development method is poor or simply there is a lack of education. Respondents thought there is a correct answer. Evidently, the interview participants recognize there is formally a correct way of how to perform the selection of alternatives. The expressed need for structured decision methods is contradicted by the poor knowledge. Education can bring better understanding and improve performance of existing decision meetings and on how decisions should be communicated.

5.1 DOCUMENT STUDY

From the investigation on the minutes from project meetings it is noted that only a small fraction of the decisions made concerns system development. Instead, it was evaluated if decisions could be traced through the ECOs. ECOs related to one system were reviewed and was found that only in a few cases, introduced changes are referring directly to a decision point. By tracing the ECO back to the main ECO the project decision is found. Normally the changes made in individual ECOs are not easily derived from project decision point referenced to in the MECO. The small influence of the project meeting on individual decision points in the daily work of system development is confirmed in the interview results.

5.2 INTERVIEWS AND WEB SURVEY

The web survey contains a subset of the interview questions. Results from both studies are reported together. In general the respondents were familiar with the issue of concern. Many of them had prior to the interview thought about shortcomings of the current decision process. Several participants responded by asking whether their answer was correct or not. In other words there is a common belief that in each case, there is a formally correct way of how decisions should be handled.

Several aspects of the decision process are covered in the interviews and the survey. Potential improvements of the decision process extend over all the investigated aspects. The IBIS model [23] was chosen in order to develop well-defined solutions that are easy to implement. The model is mainly used to organize the results and the analysis in a similar way. In the following, findings from the interviews and survey are presented referring to the IBIS model.

5.2.1 ISSUE

A decision issue is a call for action to resolve some question or a problem [23]. This topic was divided into two questions, of which the first handles information flow and the second handles the action performed to resolve the questions.

The result from the survey is illustrated in Figure 4. Here developers generally answered that they learned about changes from other developers or system architects. Object leaders and system owners gave no solid picture about the origin of information. System owners answered that they receive information through a variety of channels, some of them even from all channels. This is consistent with the representative role of the system owner. He should be ambassador of the system but all system owners should receive information through the same set of channels.

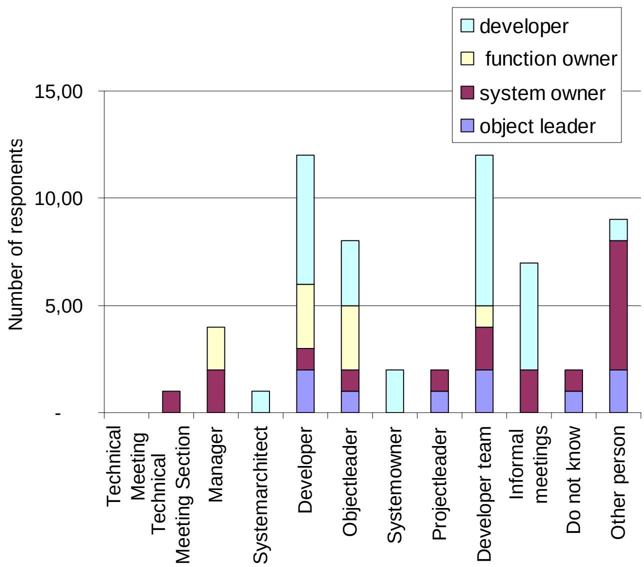

Figure 4 Information channels: the bar graph shows from where developers/function owners/system owner/object leaders receive information about changes on their system.

In several cases respondents indicated that there was much confusion about the information flow. In one answer it was commented that “information is received through all channels and it is a real mess”. In one case it was also mentioned that rumors about decisions made have negative effects and could cause panic.

From the interviews it is noted that decision issues are raised effectively within the engine and gearbox system development teams, much owing to a recently adopted Rational Unified Process (RUP) [10]. RUP is an iterative software development process which has been adapted to suit embedded system development. In their event management system actions are handled in two steps. Development engineers enter new ideas to solve a problem and then prepare a detailed solution which is presented at the architecture decision meeting. The event management system keeps the designers updated on a daily basis and takes the role of primary source of information.

There is some confusion about how development work is related to the project meetings. According to the formal process, there are three conditions, each one individually sufficient, to raise a decision point to the project level.

It concerns the project meeting if there is a change in technical specification, change in driver environment or change that will have impact on customer choice or on the market organization in any other way. During interviews only two respondents were able to identify one necessary requirement for a decision to be raised to the project level.

As a consequence of limited knowledge about conditions for PM, issues inevitably will fall between chairs. Further employees expect information to be distributed from the project meetings. When this does not happen, requirements will be regarded as unclear. The level of detail in the decision points at the project meeting should be harmonized. It must always be evident why a certain issue is raised to the project meeting and also why decisions made have a certain level of detail. Further, training about the project process, and specifically decision paths, must be emphasized.

Our observation that developers receive information primarily from other developers or team of developers and from informal meetings is consistent with the theory of the global village [4]. In the village no resident is independent of other actors. In a vehicle all systems are interfaced physically through a CAN network and logically through distributed functionality. Thus, the design work is directly affected by changes in neighboring systems as well as company external factors.

Potential improvements within the issues stage in the IBIS model are related to the observation of a large number of information channels. To reduce the number of information sources it is important to promote one source as reliable and showing endurance. Such a source will gain trust among those who depend on its information. From this rather limited study it is noted that the way of working differs significantly between individual development teams. The successful use of the event management system and the change control board (CCB) keeps the designers updated on a daily basis and takes the role of primary source of information.

5.2.2 CRITERIA

The criteria limit solutions raised by an issue [23]. Among developers a majority experience that design requirements are unclear or do not exist. This is shown in Figure 5. Object leaders and system owners believe that the level of requirement specification is sufficient.

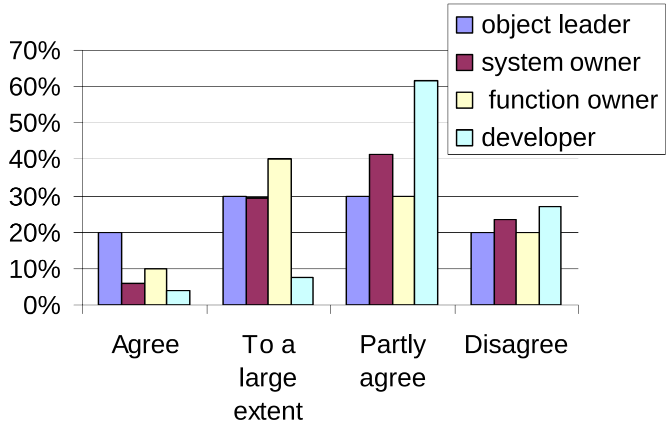

Figure 5 The percentage of each role that answered to the question how well the statement "Our design requirements are clear" corresponds to current work.

Respondents answering that requirement do not exist correlates 0.63 with those who prefer expert support while the group who find requirements unclear correlate to 0.72 with those who evaluate more than one solution alternative.

The general shortcoming of the requirements specification handling process indicated in interviews as well as in the survey can be divided into three basic needs.

- Requirements must be specified early in the process
- Tools for handling requirements are needed
- The responsibility for requirement specification must be clearly allocated to specific roles in the organization.

5.2.3 ALTERNATIVES

An alternative is an option generated to address or respond to a particular issue [23]. In several cases more than one solution concept has been developed. Often one solution is already implemented and the improved solution makes an alternative. But in some cases only one alternative is feasible.

The handling of design alternatives differs between individual groups of staff. From the survey it is noted that the use of multiple design alternatives is predominant among engineers with 3-5 years experience of system development. This is shown in Figure 6.

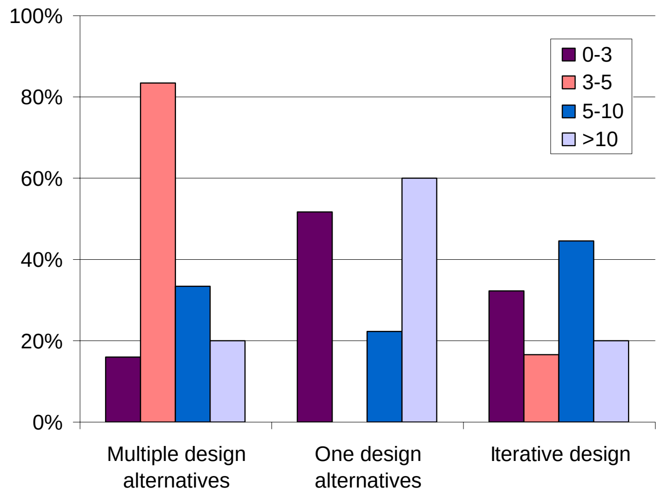

Figure 6 The relation between number of alternatives and years of work experience. Use of multiple alternatives is predominant among employees with 3-5 years experience.

The observed tendency that novice designers only use one alternative supports the idea of method training for new employees. The training should cover advantages of parallel development lines, which is necessary to produce alternative solutions. The need for training is not limited to just the newly employed but should rather be viewed as a continuous process where at least one member of each team is further educated each year.

5.2.4 EVALUATION

Evaluation is the activity of argumentation supported by information developed through prior knowledge, analysis, experimentation, or information gathering (e.g. expert advice) [23]. Respondents in general acknowledge the use of unstructured decision methods (Figure 7) but on the other hand emphasize the lack of structured decision methods. Among structured methods, expert support is most commonly used today. This method is also the most preferred together with checklists. The system owner role correlates strongly with preferred structured methods (formal design review meeting 0.59, checklist 0.55, expert support 0.66; rating however, is not correlated).

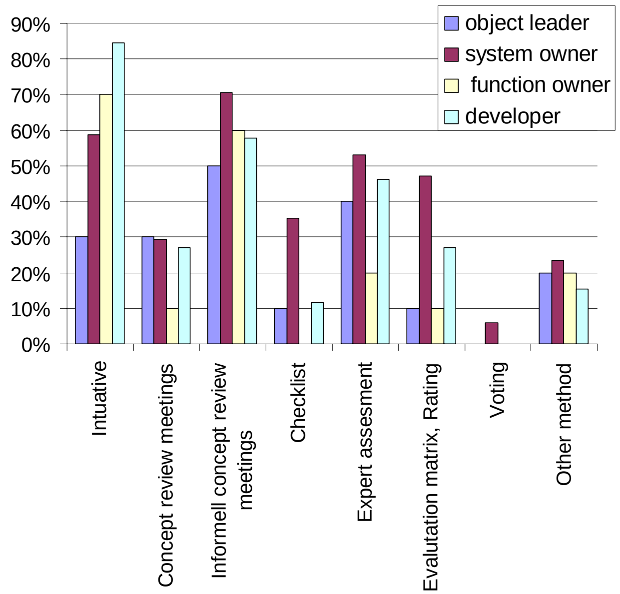

Figure 7 Current use of evaluation methods within each role. The present process is dominated by unstructured methods.

Expert support is provided by the technical career, which is an acknowledged career alternative within the company for those who has chosen to become experts in their field of work. The career starts as engineer and an experienced engineer can then be promoted to senior engineer. The highest level of the career is technical manager and senior technical manager which are positions. The analysis has shown a need for support in methods and technology, especially for the recently employed engineers. According to the role description a senior engineer should be able to educate and coach other engineers.

A question in the web survey about what structured decision methods are used today, strongly correlates with the use of multiple alternatives in question 10. At the 90% significance level correlations with the use of more than one solution alternative were 0.55 (checklist), 0.55 (expert support), 0.6 (ranking method), 0.79 (formal design review meeting).

Among those who use iterative desigin as a method for development the majority has worked 0-1 years. The group is evenly spread between departments but correlates strongly (0.60, 9 of 13 respondents) with the perception of insufficient requirement specification. There is a strong correlation between engineers with 3-5 years of working experience and those who use several alternatives (Figure 6).

One development team illustrated how alternatives are evaluated on the basis of three architectural principles:

- simple is best
- smallest number of variants
- minimal interface between modules

From the interviews and the web survey the need of structured methods has become evident. From an organizational point of view new roles must be established to support these methods. Three new roles are suggested:

- Maintenance and support of methods. Due to circumstantial changes and improved knowledge methods will always need maintenance. Maintenance responsibility adds a new role to the organization.
- Gathering and processing of basic data for decision tools. The large amount of information needed often makes it difficult to apply structural methods. The attention is focused on the method and the resources requirement for handling of data tend to be overlooked. This work should be performed by a central function since much of the data is common for several systems.
- Expert support. Experts already exist at R&D but the expert role has to be well defined. The experts should be trained in a number of the structured methods given in Chapter 4. Further it must be clear where experts are located in the organization.

The formal role description of senior engineers in theory provides us with exactly what is needed, but obviously it does not work in practice. The main explanations are that the persons do not have time and that the role is unclear for others. The technical career therefore needs to gain more acceptance and measures are needed to improve its attractiveness, but also to clarify the role of an expert. Today there are only a few positions as technical manager covering the large developing organization working with system development.

In Figure 8 the results from the survey has been compared to the similar surveys mentioned above [9][20][2]. The Finnish survey had evaluation matrix and rating defined as two different methods, in the comparison they are summarized to one, “Evaluation matrix, Rating”. The methods have otherwise been matched exactly when data was available and set to zero if not. The comparison shows one possible explanation why developers would like to increase the use of concept review meeting and checklist, the use of those methods is extremely low in comparison.

Decision making by review meetings is supported by Christensen and Krainer [4]. They suggest that the project review meeting should be used as decision point in projects with high degree of uncertainty.

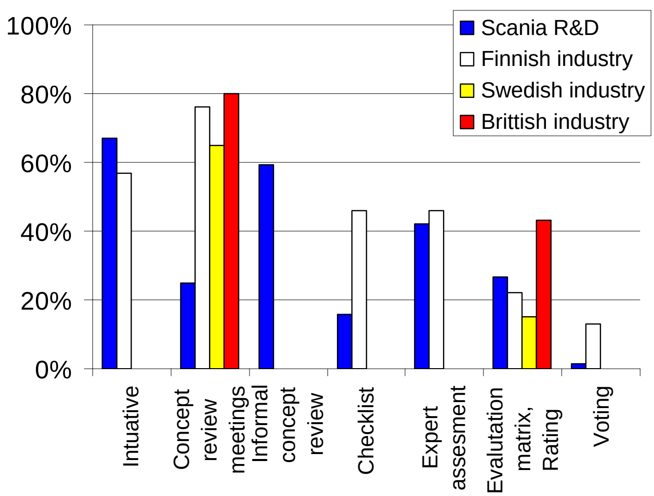

Figure 8 Utilization of methods in system design

5.2.5 DECISION

A decision is the agreement to adopt a certain alternative to resolve the issue [23]. In general there is no established process for decision making. Most information about decision making was received from respondents within engine system development. The recently adopted RUP-process is considered as effective and decisions made within the change control board are regarded as well founded. Issues, in the event management system are prepared prior to the meeting and argued by the issuer at the CCB. More complex issues are argued by the most experienced participants at the meeting. Further, decisions made at the CCB are given the support of the management group which is present at the meeting.

- A change control board should be established for chassis, power train and cab according to RUP. The representative at the system integration meeting will report the decisions made at the local meeting and present new issues that need to be considered.
- A system architect must be appointed and given the task of representing the system at the system integration meeting.

The successful use of the event management system in the CCBs leads to well founded decisions. It is reasonable to assume that this process could be used successfully within several development teams. The use of one common process will integrate the overall development process and eventually solve the problem of the "global CAN-village" within system development.

The importance of clearly documenting each decision must be emphasized. It should be documented how the decision was made and amongst which different alternatives the choice was made. The transfer of knowledge depends on this documentation and is crucial for the next development team entering into a related issue [4].

To ensure the confidence in the "system integration meeting" each part of the organization must be represented and the task must therefore be prioritized.

6 CONCLUSIONS

The quality of the developed systems is used by the company as a sales argument and the product as whole is considered to be state of the art. The reasons for this success found within this work are well motivated engineers working in an open minded climate. Some locally adopted solutions were found to have very high potential and should be further used.

- The use of Change Control Boards provides a structured way of handling tasks.
- The system architect role manages and coordinates design changes
- Formulating and using basic design principles that tie the project together.
- Evaluating different alternatives using trade-off curves

The main problem found was to be the general confusion about where decisions are made. This problem is connected to the finding that the level of the electrical issues discussed at the project monthly meeting is not harmonized. New employees where found to feel a lack of expert support and the use of structured methods were found to be very low.

We suggest and prioritize three improvements to respond on those problems which are further explained in the following sections.

1. Strengthen the role of the technical career
2. Improve knowledge transfer trough documenting design know-how
3. Educate engineers in the use of structured methods

These improvements are illustrated in Figure 9.

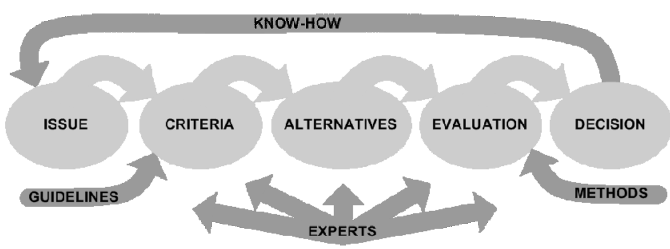

Figure 9 Proposed improvements related to the IBIS model [23]

6.1 IMPROVEMENT OF THE TECHNICAL CAREER

The low number of formally appointed experts makes the role invisible and is also very low compared to other more traditional parts of the organization. Two or three new technical managers should be appointed within system developing organizations. Possible technical areas to be covered are application software, operating system and human machine interface.

Each newly employed engineer should be given a suitable senior engineer as coach during there first year. The senior engineer does not need to be within the same organization, but should be an expert within the same field. The first meeting should take place 2-3 months after the employment starts and the second and third at 4-5 months interval. Topics to be discussed should be methods, technology and personal network. To ensure that meetings are made this should be made a mandatory part of the introduction for newly employed.

To further enhance knowledge transfer, senior engineer should be made available for support in methods and technology for all developers. Time for knowledge transfer must therefore be allocated to the senior engineer. This measure will clarify the role of a senior engineer and enable knowledge transfer.

6.2 DESIGN KNOW-HOW THROUGH CHECKLISTS AND GUIDELINES

Well written guidelines are available for electrical and electronic system development, but this work must be updated and more widely promoted. Guidelines are available for electrical and electronic system development, but this work must be updated and more widely promoted. Guidelines solve the earlier stated problem with insufficient requirement specifications by supporting the engineer.

Checklists should be developed to aid the developer in each design step and thereby ensure product quality. There is a checklist available for mechanical design, but this must be made suitable for system development. The document describing the development process in the powertrain department could be used as a starting point. An updated checklist would make the project decisions harmonized by clarifying when decisions need to be made at what level.

Design know-how is currently not well documented within system development, but a method and template is developed and in use in other areas. Design know-how for system development should be documented using this template.

6.3 EDUCATE ENGINEERS IN THE USE OF STRUCTURED METHODS

Structured methods make the decision process visible and ensure that it can be repeated. The cost of pursuing the wrong alternative is avoided and recently employed engineers can carry out an evaluation of alternative concepts.

The survey shows that the use of unstructured methods such as an intuitive choice is high, but the use of structured methods mentioned in the related work section is very low. The knowledge and use of structured decision methods should be increased by adding this topic to the introductory ECU-system course. This is important for the recently employed.

ACKNOWLEDGMENTS

This work has been financially supported by the Knowledge Foundation and the Swedish Agency for Innovation Systems (VINNOVA).

REFERENCES

[1] Ahmed, S.; Wallace, K. M.: Understanding the knowledge needs of novice designers in the aerospace industry. In: Design Studies 25 (2004) Nr. 2, S. 155-173

[2] Araujo, C.S.: The utilization of product development methods- A survey of UK industry. In: Journal of Engineering Design 7 (1996) Nr. 3, S. 265-277

[3] Börjesson, Anna: Making Software Process Improvement Happen. Gothenburg, Sweden, IT University of Göteborg, 2006

[4] Christensen, S.; Kreiner, K.: Projektledning: att leda och lära i en ofullkomlig värld.

[5] Dobrica, Liliana; Niemela, Eila: A Survey on Software Architecture Analysis Methods. In: IEEE Transactions on software engineering 28 (2002), S. 638-653

[6] Frost, R. B.: Why Does Industry Ignore Design Science? In: Journal of Engineering Design 10 (1999) Nr. 4, S. 301-304

[7] Hoch, Detlev et al. : The race to master automotive embedded systems development. . Germany : McKinsey Company, Automotive and assembly sector business technology office, 2006

[8] Hurst, K. Engineering Design Principles, ISBN: 0340598298, Elsevier Science & Technology Books, May 1999

[9] Janhager, J.; Persson, S.; Warell, A.: Survey on Product Development Methods, Design Competencies, and Communication in Swedish Industry. : Proceedings of TCME, Tools and Methods of Competitive Engineering. Wuhan, China, 2002

[10] Jacobson, I., Christerson, M., Jonsson, P., Overgaard, G., Object-Oriented Software Engineering: A Use Case Driven Approach. Addison-Wesley Pub Co, ISBN0201544350, 1992

[11] Karlsson, J.: Marknadsdriven Produktledning-Frn kundbehov och Krav till Lönsamma Produkter. In: VI report, Focal Point AB (2003)

[12] Kazman, Rick; Klein, Mark; Clements, Paul: ATAM: Method for Architecture Evaluation. : Carnegie Mellon Software Engineering Institute, 2002

[13] Kazman, Rick; Asundi, Jai; Klein, Mark: Making Architecture Design Decisions: An Economic Approach. , Nr. 2002-TR-035 : Carnegie Mellon Software Engineering Institute, 2002

[14] Larses, Ola: Architecting and modeling automotive embedded systems. Stockholm, Royal Institute of Technology, 2005

[15] Merecer Management Consulting and Hypovereinsbank, Studie-Automobiletechnologie 2010. Munchen Aug 2001.

[16] Morgan, J. M.; Liker, J. K.: The Toyota Product Development System: Integrating People, Process, and Technology. Productivity Press, 2006.

[17] Pretschner, A. et al.: Software Engineering for Automotive Systems: A Roadmap. In: International Conference on Software Engineering (2007), S. 55-71

[18] Pugh, Stuart: Total design: integrated methods for successful product engineering. . Wokingham: Addison-Wesley, 1990. - ISBN 0-201-41639-5

[19] Saaty, T. L.: How to make decisions: The analytic Hierarchy Process. In: Journal of Operational Research 48 (1990), S. 9-26

[20] Salonen, Mikko; Perttula, Matti: Utilization of concept selection methods – a survey of finnish industry. . Helsinki, Finland : Helsinki University of Technology, 2005

[21] STATISTICA, http://www.statsoft.com/

[22] Thiel, S.; Hein, A.: Modelling and using product line variability in automotive systems. In: Software, IEEE 19 (2002) Nr. 4, S. 66-72

[23] Ullman, D.G: The Ideal Engineering Decision Support System. URL: http://www.robustdecisions.com/theidealenginsyste1.pdf, 2007-10-01.

[24] Ulrich, K. T.; Eppinger, S. D.: Product design and development. McGraw-Hill New York, 1995.

[25] Yin, R. K.: Case Study Research: Design and Methods. Sage Publications Inc, 2003.

Paper B

Evaluating Flexibility in Embedded Automotive Product Lines Using Real Options

Håkan Gustavsson
Scania CV AB and Mälardalen University
hakan.gustavsson@mdh.se

Jakob Axelsson
Volvo Cars and Mälardalen University
jakob.axelsson@mdh.se

Abstract

Embedded automotive architectures and software need to support a large number of vehicle product lines over many years of production. This leads to a complexity and many uncertain factors when developing such systems and a need for support in the design process. An evaluation method using Real Options provides the opportunity to analyze the cost of designing for flexibility to cope with a future growth of a product line, based on the estimated value of the future functionality.

In this paper Real Options is applied on a case within the automotive industry. To improve the usability an evaluation process is defined to aid engineers. This process provides a way of valuing system designs and enables the engineer to think about the future in a systematic manor. The value of a flexible design can thereby be quantified and the proposed process shows how it can be accepted by practitioners.

1. Introduction

Today most innovations made within the automotive domain are driven by electronics. According to a 2006 study made by McKinsey [10] they expect the total value of electronics in automobiles to rise from the current 25% to 40% in 2010. The automotive customers demand new functionality with every new product release and the time-to-market is constantly shortened.

Most design decisions of automotive electronic and electrical (E/E) architectures are done during the early phases. Often, the E/E architecture needs to support a full product line of vehicles or vehicle variants that are released over a number of years. They must allow a large degree of variability to cope with the demands of different customers.

To be able to satisfy this growing demand the Original Equipment Manufacturer (OEM) needs to develop architectures that can evolve throughout its lifetime without forcing premature architectural changes. Similar products in some other industries solve this problem by simply adding extra resources to cope with future demands. The cost sensitive automotive industry has to optimize the use of the system's limited resources, but in the meantime also be flexible. The design decisions are usually based on many factors that pull in different directions such as maintenance, portability, usability etc. The complexity of the system and the many uncertain factors create a need to define methods which can provide guidance in the design process.

In this paper, Real Options is applied on a case within the automotive industry. To improve the usability an evaluation process is defined to aid practitioners such as developers and architects. The evaluation process provides a way of valuing system designs and enables the practitioner to think about the future in a systematic manor. The value of a flexible design can thereby be quantified and the proposed process shows how it can be accepted by practitioners within the automotive industry.

1.1. Paper outline

In the first section the evolution of financial options into Real Options is discussed and briefly also the social and organizational aspect of using Real Option. Three different methods of valuing Real Options are then studied. The question if Real Options are suitable to value the flexibility in embedded system design is answered in Section 3. An evaluation process is presented in Section 4. The evaluation process is then applied on a real case from the automotive industry in Section 5. Various related work is then presented and followed by conclusions and future work.

2. Introducing real options

In this section, the concept and background of options in general and Real Options in particular is introduced.

2.1. Financial options

Using options theory is one approach to deal with the high level of uncertainty when making design decisions in the early phases. The theory derives from finance where an option is the right but not the obligation to exercise a feature of a contract at a future date [11]. A typical example is a stock option which gives the right but not the obligation to buy a certain stock at a given price on a predefined date. An option has a value because it gives its owner the possibility to decide in the future whether or not to pay the strike price for an asset whose future value is not known today. An option therefore provides a right to make the costly decision after receiving more information.

There are two different types of options, American and European. A European option may only be exercised on the predefined exercise day whereas an American option can be exercised any time until the exercise date.

2.2. Real options

Real Options could be seen as an extension of financial option theory to options on real (nonfinancial) assets. Copeland [9] defines a real option as: ”the right, but not the obligation, to take an action (e.g. deferring, expanding, contracting, or abandoning) at a predetermined cost called the exercise price, for a predetermined period of time - the life of the option. ” Since the 1990s options theory has started to be utilized within the field of engineering. It is then called Real Options and was developed to manage the risk of uncertain design decisions. In 2001 de Neufville coined the expressions Real Options "in" and "on" projects. Real Options "on" projects treat the enabling technology as a black box while Real Options "in" projects are options created by changing the actual design of the technical system. Real Options on projects provide a more accurate value of the project and Real Options in projects support the decision on what amount of flexibility to add. ”Real Options on projects are mostly concerned with an accurate value to assist sound investment decisions, while Real Options in projects are mostly concerned with go or no go decisions and an exact value is less important.” [18]

2.3. Social considerations

Real Options do not only provide a way of valuing system designs, but it also forces the developer to think about the future in a systematic manor. By giving future flexibility a value it assists the developing organization in making decisions and also enables a way of predicting the growth of the complete system [13]. Leslie concludes the article ”The real power of Real Options” with ”The final, and perhaps greatest, benefit of Real Option thinking is precisely that - thinking”[13]. The possibility of changing the way people think might also be the hardest part in bringing acceptance to new methods such as using Real Options. The new method must not only be better than the one it is replacing, it should also be triable, observable and have low complexity [9].

2.4. Valuing real options

One of the advantages with Real Options compared to many other architecture evaluation methods is the possibility to value different system designs and thereby finding the most economically sound investment. This is probably the most complicated part of using Real Options, and during the years since ”Real Options” was coined there have been several approaches to calculating its value. They all have various assumptions and we will in this section evaluate the most appropriate for our case. There are three general solution methods [1]:

- Black-Scholes-Merton model. This method calculates the option value by solving a partial differential equation including the value of a replicating portfolio.
- Binomial model. The dynamic programming approach lays out the possible future outcomes and folds back the value of an optimal future strategy.

- Monte Carlo simulation. The simulation approach averages the value of the optimal strategy at the decision date for thousands of possible outcomes.

We will now present the first two models in more detail, whereas the third model is beyond the scope of this study.

2.5. Black-Scholes-Merton model

The Black-Scholes-Merton (BSM) model, for which they later received the Nobel price, was created by Black and Scholes 1973 and is widely used on financial options. The BSM model makes two major assumptions that concern our case: it demands a replicating portfolio and it only supports European type options.

A replicating portfolio contains assets with a value matching those of the target asset. The replicating portfolio of financial options can easily be found on the stock exchange as the stock value, but when looking at Real Options that are not traded it can be very difficult to find.

Considering our case it seems very unlikely that assets needed is exercised at a predefined time. Sullivan [16] discusses the assumptions made and writes: ”They will not hold for some, perhaps many, software design decisions.” More recently Copeland [9] argues: ”There are valuation methodologies that effectively capture the complexities and the iterative nature of managerial decisions, and the Black-Scholes-Merton model is not the only, or even the most appropriate, way to value Real Options.” Also Amram [1], who provides a four step solution using BSM, states: ”The Black-Scholes solution is appropriate for fewer Real Options applications, but when appropriate it provides a simple solution and a quick answer.” The conclusion is that the BSM model is suitable for financial options, but hard to use in our case.

2.6. Binomial model

The binomial model does not need a replicating portfolio [6] and also supports American type options. The initial value, A, changes with each time interval and either goes up with the probability p to Au or down to Ad until its final date [1]. The value of the asset (A) at each decision point is given through Equation (1) with r being the risk free interest rate and σ the volatility and the time period Δt.

$$A = (pA_u + (1-p)A_d)e^{-r\Delta t} \qquad (1)$$

Assuming that the underlying asset has a symmetric up and down movement u = 1 / d, then the up and down factor is given trough:

$$u = e^{\sigma\sqrt{\Delta t}} \quad (2)$$

$$d = e^{-\sigma\sqrt{\Delta t}} \quad (3)$$

The probability of an up movement is then:

$$p = \frac{e^{r\Delta t} - d}{u - d} \quad (4)$$

Looking back at our case the value of the flexibility option would change during the development stages (see Figure 1).

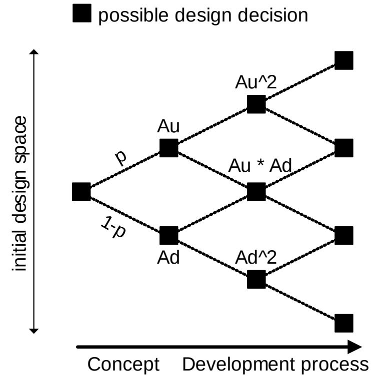

Figure 1. The decisions made narrow the initial design space.

3. Real options in embedded system design

There are as many Real Options in embedded system design projects as in any other engineering project. Those systems contain a large amount of design variables and parameters that can be valued as Real Options in projects.

3.1. Automotive embedded systems

The building blocks of an automotive E/E system consist of electronic control units (ECUs) connected to communication networks. The communication networks are usually divided into subnetworks and the communication between those are made through gateway ECUs connected to a backbone. Different sensors and actuators are connected to the ECUs depending on the function allocated to the ECU.

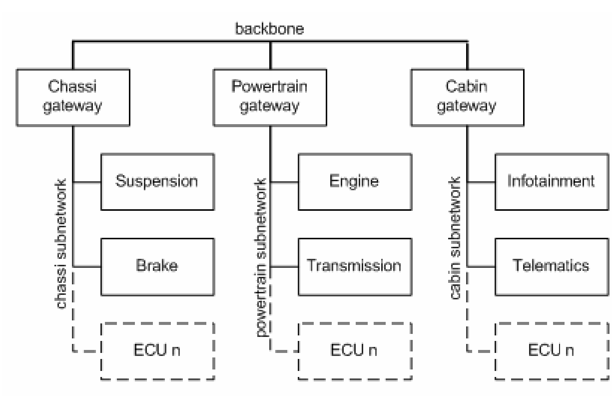

Figure 2. A typical vehicle communication network

3.2. Suitability of real options

To find out if Real Options would be a support in embedded system design one needs to clarify the characteristics of this domain. As stated earlier [10] the large volume and cost of the product makes errors in the design very expensive. Also, conflicting requirements found late in the development phase cause a high cost. At the same time there is a very high level of uncertainty during this design phase and important decisions are made by a small group of engineers [2]. The automotive embedded systems are characterized by being mechatronic systems which adds complexity. The systems are often resource constrained and trade-offs between the system behaviour and the resources required is of great importance [13].

When to use Real Options is explained by many authors. Copeland [9] states that ”It is making the tough decisions - those where the Net Present Value is close to zero - that the additional value of flexibility makes a big difference.” This is in our case true when developing a new functionality where the market demand is very uncertain. If the design would include a real option to abandon or change course the risk taken could be minimized. Under these conditions, the difference between real option valuation and other decision tools is substantial.

3.3. Real options in automotive systems

There are many new functions that are about to be introduced or already introduced that have a large impact on the electrical system of automotive vehicles.

Using Real Options as a method to evaluate alternative solutions enables the possibility to value the flexibility of the technical solution. A solution that is more likely to withstand change due to future demands has therefore a higher value when evaluated using real options compared to traditional evaluation methods. To enable the possibilities of future reuse the system needs to be designed with interfaces between components (both SW and HW) that are prepared for future needs.

The design will be different depending on how long the system is planned to withstand future change. To evaluate what level of flexibility is appropriate one must therefore first provide the rough requirements of future needs. Given the estimated value of the future functionality a real option analysis will then show what amount of flexibility should be added to make the investment adequate. Current and future technical demands of the system together with economical and organizational demands call for a systematic evaluation process.

4. Evaluation process

To improve the usability we have defined an evaluation process that can aid practitioners such as developers and architects of embedded automotive systems. Practitioners working with embedded systems are not used to value design alternatives with economic valuation methods. To make the practitioners utilize and trust the method it is important to present a step-by-step process how to carry out the valuation. During the evaluation process the different stakeholders will have to specify their gut-feeling in figures and consider if flexibility has an added value. The evaluation process presented below consists of eight steps with a description and some concrete advices.

Step 1 - Describe the design alternatives

Each valid design alternative is described to identify what resources are used. This can be simplified by reusing patterns from previous designs.

Step 2 - Perform traditional valuation

The traditional method to derive the value of an investment is by calculating its Net Present Value (NPV) taking into account the value today of cash received or paid in the future. To calculate NPV a discount rate is used, often corresponding to the current interest rate.

Step 3 - Find sources of flexibility

It would not be wise to analyze all the real options available. When designing a function distributed over a communication network there are some assets that are generic and can easily be used by other functions. Those represent the source of flexibility or Real Options. Commonly they are hardware assets such as inputs, outputs or communication capacity. If there is such an asset, the difference in NPV could be due to the cost of designing for flexibility. If there is no source of flexibility the result given through the valuation in Step 2 is true, and the evaluation is completed.

Step 4 - Estimate value of flexibility

Each resource is analyzed to distinguish if it has a future value. When available it provides an increased amount of flexibility or available design space and thereby an added value.

The value will often be due to the revenue of future functions which represent the underlying asset (S) and can be calculated through a simplified model (5). The product cost is the estimated costs during the system lifecycle.

S= volume x (customer price - product cost) (5)

Step 5 - Estimate the cost of utilizing flexibility

The price of implementing flexibility is usually a future function or extension of an existing function. The price to be paid is therefore the added cost of implementing this future functionality. The added cost is the exercise price of the real option (see Figure 3).

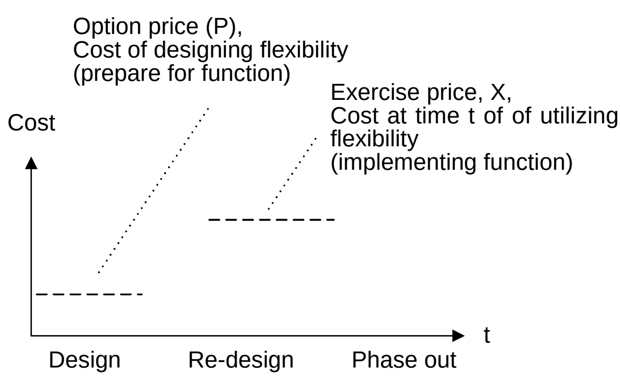

Figure 3. The price and excerise price of the option

Step 6 - Perform valuation using Real option

The value of the flexibility can be calculated using real option valuation. The quantitative data needed (Table 1) to perform a real option valuation should be extracted for the design concepts as follows:

- The planned lifetime of the platform needs to be estimated. If the function has not been implemented before the expiration date the value of the real option is lost.
- The current value of implementing flexibility is the result from Step 4.
- The cost of utilizing the flexibility is given from Step 5.
- The volatility is a measure of the annual up or down movement of the option value and often represents the uncertainty of future customer demands. This can be estimated through historical data or expert assessment.

By using the binomial model the value of the option premium can be calculated.

Option on stock	Real option in embedded systems
Option price (C)	Cost of designing for flexibility
Exercise price (X)	Cost of utilizing flexibility
Underlying asset value (S)	Current value of implementing flexibility
Volatility (σ)	Uncertainty of costumer demand
Time to expiration (T)	Lifetime of the current system
Option value (V)	The value of designing flexibility

Table 1. Factors affecting the value of an option

Step 7 - Compare the alternatives

Real option theory provides an extension to the traditional NPV valuation by adding the value of flexibility. The so called expanded NPV is the sum of the static NPV and the value of the option premium [17]:

Expanded NPV = Static NPV + Option premium (6)

The best investment is therefore to choose the design alternative with the highest Expanded NPV.

Step 8 - Make decision

Real Options provide the opportunity to analyze the cost of designing for future growth of a platform, based on the estimated value of the future functionality. It is important to stress that decision are often based on factors that are not valued using the presented evaluation process. Other factors that influence the decision are the choice of supplier, time-to-market, project priority or organization. The last step is therefore to make the decision based on the trade-off between all influencing factors.

5. Case study: Network usage

To analyze the process and its usefulness it is applied on a real case taken from the automotive industry. The problem was how to integrate a new feature implemented in software into an existing E/E architecture. A key element of the problem is in which ECU the new functionality should be implemented.

5.1. Step 1 - Describe the design alternatives

A pre-study has found two alternative ways to provide this feature (Figure 4).

Design alternative 1 provides this feature by connecting the communication link directly to the current cabin gateway ECU through an existing but unused bus interface, and the advantage is a low development cost.

Design alternative 2 uses a new ECU to create the external communication. Alternative 2 is more expensive in development cost and component cost, but does not use the last available communication link in the cabin gateway.

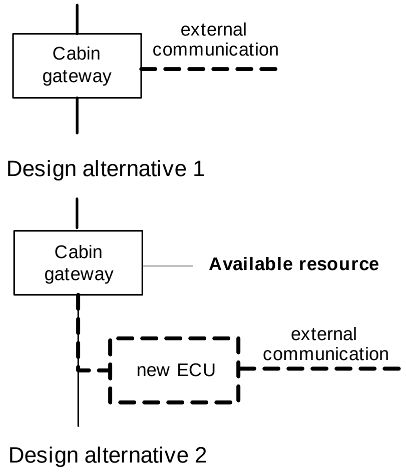

Figure 4 Two design alternatives to provide the demanded feature.

5.2. Step 2 - Perform traditional valuation

The development cost of Alternative 1 is zero and SEK 5 million (Swedish krona) for Alternative 2. The cash flow of alternative 1 is higher due to its low component cost. The difference in NPV between the two alternatives is SEK 6.9 million given the annual discount rate of 11%. The analysis of the valuation tells us to choose Alternative 1, but this does not take the value of flexibility into account.

		Alternative 1	Alternative 2
Development cost:		0	-5
Cashflow	1st year	15,5	15
	2nd year	15,5	15
	3rd year	15,5	15
	4th year	15,5	15
	5th year	15,5	15
	NPV	57,3	50,4
Difference:		6,9	

Table 2. The calculated NPV of the two design alternatives in million SEK.

5.3. Step 3 - Find sources of flexibility

The communication link is a limited resource which can be of interest to a large number of functionalities, but those functionalities cannot be safely mixed with an external device. Alternative 2 thus gives a higher flexibility for future functionality than Alternative 1.

5.4. Step 4 - Estimate value of flexibility

Network communication is a limited resource within the automotive industry. Each network has a predefined maximum capacity and the utilisation is also dependent on the physical location of the network cable. There is a growing market demand to monitor and control different vehicle functions through the use of external devices. To meet this requirement one must provide a way to connect external communication devices to the vehicle.

The expected value of the future function (underlying asset, S) is estimated to be SEK 10 million using the simplified model (5).

5.5. Step 5 - Estimate the price of flexibility

The exercise price SEK 2.9 million of finally implementing the function is an average of the potential functions found in the product portfolio. The exercise price includes the cost of ECU, sensors, cables, and developing application software.

5.6. Step 6 - Perform valuation using Real option

The communication link provides flexibility to the system and its value can be calculated using real option valuation. The product portfolio gives us a set of functionalities which could require the use of the communication link. The data needed is provided through an internal pre-study. The planned lifetime of the platform is 5 years.

The minimum goal of the investment in the alternative is to exceed the interest gained from the companies risk free interest rate (5%). The volatility is predicted to be 25% mainly due to the uncertainty of future demand. The current value of the option is calculated to SEK 7.7 million (see Figure 5).

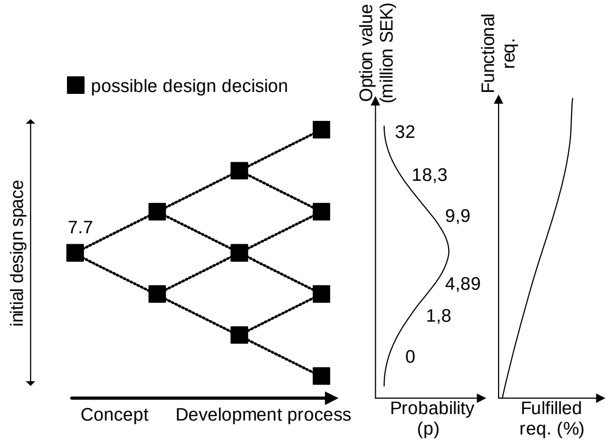

Figure 5. The future option value increases with the number of requirements implemented.

5.7. Step 7- Compare the alternatives

Alternative 2 would be a sound investment if the value of the option premium is higher than the calculated difference (SEK 6.9 million) in Table 1. The option premium was calculated to SEK 7.7 million, which means that adding the flexibility is a good investment compared to the alternative without flexibility.

5.8. Step 8 - Make decision

The results show that the future option value increases with the number of requirements implemented (Figure 5). If only a low number of requirements will be demanded the value of the option will be lost. It also shows how the risk changes with the probability. This risk could be eliminated by not implementing the possibility to support a certain requirement. This would lead to a limited design space where an improved functionality cannot be implemented without a redesign of the system.

6. Discussion

The results show that investing in a flexible design would most likely be a sound investment if a large part of the future requirements were implemented during the system life cycle. The diversity of the proposed functionality makes it very uncertain what functionality will be implemented, which also is the reason why flexibility has a value. The prediction of the volatility and the value of the underlying asset are crucial to the results. One of the strengths when using real option valuation is that the uncertainty is taken into account and not left out of the calculation. It also provides a valuation method that can be used to analyze different future scenarios. Similar analyses can be done to estimate the value of future functions by iteration of sales volumes, customer price, etc.

7. Related work

Real Options is far from being the only method developed for valuing architectures. There are few methods that makes an economic consideration, CBAM [12] being an exception. Real Options is unique by also considering the flexibility and the architectural evolution over time [3]. Our literature survey has found three research contributions that involve the usage of real options in system design involving software or hardware. None of them addresses embedded systems or the automotive domain explicitly.

Browning et al. [8] extend Real Options ”in” projects to architecture options and present a theoretical example where stakeholder overall value increases with 15% by designing the system for the right amount of adaptability. The framework presented shows a way to implement the optimal degree of flexibility. The initial research proposes using the model of Black and Scholes to calculate the value of the Real Options, but do not present a case. Browning shows that architecture options provide the information to better predict the need for system upgrades and thereby increasing the lifetime value of the system.

Bahsoon et al. [3] use the concept of ArchOptions to value the stability and scalability of software architectures. ArchOptions are valued using the model of Black and Scholes and a replicating portfolio is therefore needed. The portfolio is valued by the requirements it supports during the operation of the software system.

Banerjee [6] argues the need for flexibility and presents the solution of flexibility options compared to a fixed design. The value of the flexibility option is calculated using the binomial model that does not need a replicating portfolio and also supports American type options. The work done by Banerjee seems to be what best meets our prior stated problem definition.

8. Conclusion & Future work

This paper has presented an evaluation process for practitioners using Real Options theory that enables analysis of both economic and engineering factors. It presents a possibility to put an economic value on system adaptability and could therefore support the design decisions in the early phases. Real Options provide the opportunity to analyze the cost of designing for future growth of a platform, based on the estimated value of the future functionality.

When developing an embedded system using Real Options each function would first buy the right but not the obligation to use the asset at a future date. The real option approach could when fully developed provide not only evaluation but also prediction of future needs. Real Options on system design is a newly added extension of the option theory and there is not a developed method available. There is research needed to find ways on how to calculate volatility based on available data. There is also a need to make case studies focusing on the acceptance of the process in the developing organization. Also, it would be interesting to dig deeper into the software aspects of an embedded system, and analyze the value of building more flexible software, e.g. based on frameworks like AutoSAR.

9. Acknowledgements

This work has been financially supported by the Knowledge Foundation and the Swedish Agency for Innovation Systems (VINNOVA).

10. References

[1] Amram, M. and Kulatilaka, N. Real options. Harvard Business School Press Boston, Massachusetts, 1999.

[2] Axelsson, J. Cost Models with Explicit Uncertainties for Electronic Architecture Trade-off and Risk Analysis. In Proc. 16th International Symposium of the International Council on Systems Engineering, Orlando, Florida, July, 2006.

[3] Bahsoon, R. and Emmerich, W. ArchOptions: A Real Options-Based Model for Predicting the Stability of Software Architecture. In Proc. of the 5th ICSE Workshop on Economics-Driven Software Engineering Research, 2003.

[4] Bahsoon, R. Evaluating Software Architectures for Stability: A Real Options approach. In Proc. of the Doctoral Symposium of the 25th International Conference on Software Engineering, 2003.

[5] Bahsoon, R., Emmerich, W., and Macke, J. Using ArchOptions to Select Stable Middleware-Induced Architectures. In IEE Proceedings on Software, IEE Press 152(4), p.176-186, 2005.

[6] Banerjee, P.. Describing, assessing and embedding flexibility in system architectures with application to wireless terrestrial networks and handset processors. M.Sc. Thesis, Massachusetts Institute of Technology, System Design and Management Program, June 2004, 2004.

[7] Brach, M. A. Real options in practice. John Wiley and Sons, Hoboken, NJ, 2003.

[8] Browning, T. R. and Engel, A.. Designing Systems for Adaptability by Means of Architecture Options, In Proc. 16th International Symposium of the International Council on Systems Engineering, Orlando, Florida, July, 2006.

[9] Copeland, T. and Antikarov, V. Real Options: A Practitioner's Guide. TEXERE Publishing Limited, New York, 2001.

[10] Hoch, D., Huhn, W., Naher, U. and Zielke, A. The race to master automotive embedded systems development. McKinsey Company, Germany, Automotive and assembly sector business technology office, 2006.

[11] Hull, J. C. Options, Futures, and other derivative securities. 2nd Edition. Prentice Hall International Editions, New Jersey, 1993.

[12] Kazman, R., Asundi, J., and Klein, M. Making Architecture Design Decisions: An Economic Approach. Technical Report CMU/SEI-2002-TR-035, Carnegie Mellon Software Engineering Institute, 2002.

[13] Larses O. Architecting and Modeling Automotive Embedded Systems. Ph.D. Thesis. Dept. of Machine Design, KTH, Stockholm, Nov. 2005.

[14] Leslie, K.J. and Michaels, M.P. The real power of real options, The McKinsey Quarterly, vol. 3, McKinsey & Company Inc., 1997

[15] de Neufville, R., Real options: dealing with uncertainty in systems planning and design, In Proc. 5th International Conference on Technology Policy and Innovation, Technical University of Delft, Netherlands, 2001.

[16] Sullivan, K. J., Chalasani, P., Jha, S., and Sazawal, V. Software Design as an Investment Activity: A Real Options Perspective. In Proc. Real Options and Business Strategy: Applications to Decision Making, 1999.

[17] Trigeorgis, L. A Conceptual Options Framework for Capital Budgeting, Advances in Futures and Options Research, vol.3, p. 145-167, 1988.

[18] Wang, T. Real Options in Projects and Systems Design - Identification of Options and Solution for Path Dependency. Ph.D. thesis, Massachusetts Institute of Technology, Cambridge, 2005.

Paper C

A Framework For The Evaluation Of Resource Efficiency In Automotive Embedded Systems

Erik Persson
Royal Institute of Technology, KTH
Industrial Engineering & Management
erikperaxel@gmail.com

Håkan Gustavsson
Scania & Mälardalen University
hakan.gustavsson@mdh.se

Abstract

This article discusses the resource utilization of embedded systems in the automotive industry. Traditionally, the major cost driver – or resource input – has been regarded as the hardware cost. Issues such as software development costs and maintenance costs have historically been neglected. In order to address this, the article embraces the more comprehensive view on resources that a resource can be regarded as anything which could be thought of as a strength or weakness of a given firm. In this article the major drivers of resource consumption are identified. The work has also included several interviews with employees in order to find empirical data of the embedded systems in vehicles.

This paper proposes a method to evaluate the resource efficiency of user functions implemented through the embedded system. By the use of Data Envelopment Analysis – which has proven to be a useful method – the resource utilization of six user functions is evaluated. Future work of particular interest would be to perform a more extensive case study.

1. Introduction

Historically, there has been a focus on hardware costs in the automotive industry whereas the cost of software has been neglected, or at least considered hard to estimate and thus often overlooked. This is something pointed out in a report by McKinsey & Company, where it is stated that automotive players still view hardware as their main differentiating factor, and that software on the other hand is viewed as necessary but easy to change and free of cost [1].

Today most innovations made within the automotive domain are driven by electronics. A study made by Mercer Management Consulting and Hypovereinsbank in 2001 [2] claims that the total value of software in cars will rise from 4% to 13% by 2010.

Looking at resources from only a traditional hardware perspective is limiting. The next step, so to speak, would be a more integrated view on the ECU and the embedded system: to consider the spending on not only hardware but also on software. However, this view is also limited in the sense that it looks at the embedded system as an isolated entity, and does not take into account the implications for the resources of the company.

The idea of looking at firms as a broader set of resources goes back to the work of E.T. Penrose and her book “The theory of the growth of the firm” from 1957 [13], a book that has laid the foundation for the more recent “resource-based view” of firms.

It is of importance to be able to quantify the degree of cost-efficiency of a solution and its resource utilization. One reason to this is that it facilitates the evaluation and comparison of different design solutions and makes it possible to better value the resources that are consumed by the system. In order to make an adequate design decision, one must consider numerous factors. There are obvious aspects such as size, cost and capacity of a component, yet other less tangible factors are very important, factors such as customer preferences, development cost, production volume and time to market. All these factors – and many more – influence the necessary input of resources as well as the magnitude of the output, in other words, these factors affect how well the system is being utilized.

To address this problem, the following research question was formulated: How can one quantify the resource utilization in automotive embedded systems in the automotive industry?

2 METHOD AND OBJECTIVE

The purpose of this paper is to investigate the economic resource utilization in automotive embedded systems. The initial phase of the work consisted of formulating a problem statement and research questions. The subsequent step was to conduct an initial literature survey to gain further insight in the field, find out the state-of-the-art research and to be able to formulate relevant interview questions. The next step consisted of conducting the interviews. These interviews proved to be not only a way of collecting data, but also they were instrumental in grasping the problem in its context. The extensive literature survey also confirmed that the work on economic resource utilization in embedded systems is scarce in the research community.

Once the theoretical framework was established, a model based on this framework could be created. This model has then been used in the company case study, with empirical data from the company. The studied company is an international well known vehicle manufacturer of commercial vehicles and should be comparable to the rest of the industry. Proceeding from this study, an analysis of the results and the applicability of the model has been formed, providing the conclusions drawn from the work.

3 DATA ENVELOPMENT ANALYSIS

The theoretical framework of Data Envelopment Analysis (DEA) will be used to construct an approach to evaluate resource efficiency in embedded systems. The theory outlined below comes from the books “Data Envelopment Analysis”[4] and “Handbook of Data Envelopment Analysis”[5] both written by Cooper et al.

It is common to evaluate the efficiency of for instance a business firm by dividing its output by the corresponding input. The output is the positive outcome, and should generally be as large as possible. The input reflects the effort needed to attain this output, and it should generally be as small as possible. To measure the performance of a company it is very common to use key performance indicators (KPI). Some of these measures follow the definition of efficiency. Examples may be:

- Gross and net margin
- Revenue per employee

- Sales per employee etc.

Efficiency is a measure of performance and it is defined as follows:

$$Efficiency = \frac{Output}{Input}$$

However, the KPIs mentioned above are based on one input and one output (single input and single output). Hence, these measures are often misleading when overall efficiency is to be measured. The improvement of one output may require the increase of an input that is not reflected by the KPI. Consider an increase in the measure “Sales per employee”. This measure does not tell us anything about how costly this increase in sales was. Maybe it required extensive investments in the production plant or in terms of marketing?

To try to get around this problem, one uses normally many KPIs to reflect the different aspects of a company. However, a method that can take in several evaluation factors at the same time to measure the efficiency would be desirable. This can be done through DEA.

Data Envelopment Analysis is a relatively new method for measuring and evaluating performance when several inputs and outputs are included in the same measure. It evaluates the performance of a set of peer entities called Decision Making Units (DMUs) which convert multiple inputs into multiple outputs. The definition of a DMU is generic and flexible – it can be a company, a business unit, a hospital or even an ECU. This work uses the original CCR DEA-model which was found to be sufficient.

The efficiency is simply put calculated by dividing output by input. The difference when multiple input and output are used is that they may measure completely different factors in different units – so that the total input and output must be weighted. This is accomplished as the efficiency is calculated as the weighted sum of outputs divided by the weighted sum of inputs.

$$Efficiency = \frac{y_1 u_1 + y_2 u_2 + \ldots + y_s u_s}{x_1 v_1 + x_2 v_2 + \ldots + x_m v_m}$$

Here is output element i, is input element i, is the weight associated to output i and is the weight associated to input i. DEA uses linear programming theory to determine the weights associated to each input and output.

A DMU is considered efficient if it exhibits the following properties: the efficiency equals one and all weights are greater than zero. Otherwise the DMU is inefficient.

One of the major benefits of Data Envelopment Analysis is that the operator does not have to determine the weights subjectively and thus the relative importance of different factors. Instead, the model calculates through Linear Programming for each DMU (for instance ECU) the best possible set of weights in order to maximize its efficiency, under the constraint that the efficiency of all the other DMUs does not exceed one. Another benefit is that a mixture of quantitative and qualitative factors can be used and that the units can be different.

4 EVALUATION FRAMEWORK

In the basic version of Data Envelopment Analysis, all criteria are assumed to be of equal importance. However, it is possible to incorporate a priori knowledge such as price information to make sure that the most important criteria are the most influential to the analysis. This is done by adding constraints to the optimization problem. When introducing constraints, it is important that the factors involved by the constraints are measured in the same units, which is the case in this analysis, as all inputs are measured using the same grading scale. The following constraints will be used in the Data Envelopment Analysis:

$$0.25v_{HW} \leq v_{SW} \leq 0.67v_{HW} \quad (1)$$

This constraint implies that the cost of software is estimated to be between 25% and 67% of the cost of hardware. These numbers are based on the study made by Mercer Management and Hypovereinsbank [2]. These numbers apply to passenger cars, however it is well known that that the evolution of commercial vehicles lags the evolution of passenger cars. For instance, Zientz [12] states that truck manufacturers have tended to introduce electronic solutions only when the maturity of the new technology has already been proven by its application on the passenger car market. Hence it is very reasonable to apply this interval in the analysis conducted. A relation between hardware cost and software development cost has already been established. Looking at Figure 1 one can see that independent of time, maintenance and development costs have roughly been equally large. This implies that the maintenance cost should be between 25% and 67% of the cost of hardware.

$$0.25v_{HW} \leq v_{Maint} \leq 0.67v_{HW} \tag{2}$$

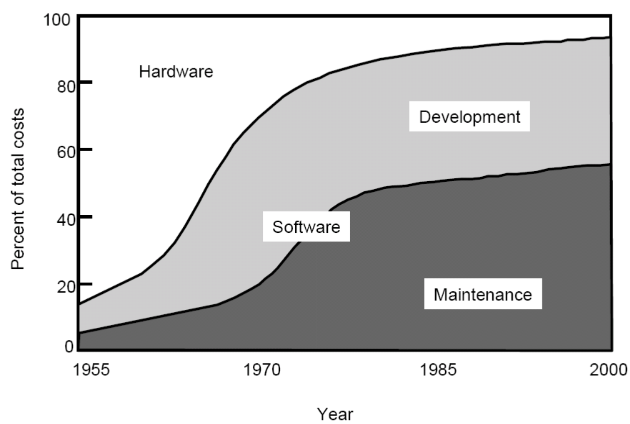

Figure 1 Hardware and software cost trends [7]

4.1 CHOICE OF INPUT

The following evaluation criteria have been identified as the most important:

Hardware cost: This criterion has traditionally been regarded as the far most important cost factor. And it is very important, however it is not the only parameter.

Software development cost: As stated previously there is an increasing importance of software costs.

Maintenance cost: For embedded real-time systems, maintenance costs may be up to four times higher than development costs[9]. According to Fornaciari, development costs and maintenance costs are of approximately the same size [7]. These figures motivate taking this factor into account as well.

Wiring harness cost: During this work a brief survey regarding the cost of wiring harness was conducted. The result was that a very coarse estimate is that the wiring harness cost equals the hardware cost of an ECU.

Time-to-market: In general, a vendor whose product reaches the market quicker than its competitor has a better chance of reaching supremacy in that product group. Debardelaben et.al states the following [6]: “Time-to-market and life cycle costs are key factors in the success of these products in the competitive electronics marketplace. These costs, therefore, should have a dominant influence on the design of embedded microelectronic systems."

Quality

Quality is one of the most important issues for commercial vehicles. Moreover, quality is the number one feature of the studied company. In this analysis quality costs are seen as opportunity costs, which should be minimized. In DEA, inputs are factors that should be minimized and outputs should be maximized. Hence, quality is regarded as an input.

Hardware resource	Weight
I/O	4
Processor	2
Flash memory	1
EEPROM	1
RAM	1

Figure 2 Assignment of weights and assumptions made

In Figure 2, the assignment of weights is presented. These are estimations based on empirical findings on the prices of these components from the interviews. At the company, products are developed according to the Product Identity. It is divided into two parts, "prestige" and "performance". Prestige refers to if the product helps to meet the expectations customers have on its products. Performance obviously refers to the performance of the product, and its constituents are listed in the figure below. The measures "Prestige" and "Performance" are assumed equally important. Hence they have a weight relation of 1 to 1.

4.2 CHOICE OF OUTPUT

Revenue: This is the most important output. It captures sales volume and value added.

The approach that will be used in this framework is to use qualitative data. An example of qualitative data is grades. In the true meaning, grades are quantitative data, as the answers have been transformed into numbers. However, in this context they will be referred to as qualitative data. This can be contrasted to for instance sales volume, where absolute data can easily be identified, and need not to be transformed into a relative measure using some grading scale. With such a transformation obviously precision of the data is lost. One of the strengths with Data Envelopment Analysis is that a combination of qualitative and quantitative data can be used.

There are several benefits of using qualitative data. First of all, with this approach it is relatively fast to collect the required and valid data and a questionnaire can be formulated. A proposed questionnaire is presented in Appendix A. Moreover, many factors are intrinsically difficult to estimate, such as the hardware cost of a particular user function. For instance, the processor load due to a particular user function is virtually impossible to measure. In such cases isolating resource consumption drivers and investigating them qualitatively probably gives a more valid result.

Moreover, the same units must be used when assigning weight constraints. Weight constraints are used to weight the relative importance of various factors. It should however be pointed out that it is best to use quantitative data to the furthest extent possible when applying Data Envelopment Analysis. However, in this case, the estimations of quantitative data would not be of a sufficiently high quality, and hence the use of qualitative data would yield more reliable data.

The risk of poor quality estimates is the reason for why the aspects of quality and time to market will not be included in the analysis performed in this paper. In particular, it is difficult to assign these issues to a particular function. However, these are important issues, but as estimations of these costs are expected to be highly unreliable, they will be excluded.

5 THE EVALUATION TOOL

A tool was developed to evaluate resource utilization. The evaluation tool consists of two parts; a basic Microsoft Excel sheet and a Matlab model. All calculations are based on the theoretical framework presented by Cooper[4][5] and the assumptions outlined in this paper.

The first step is to collect the data for the evaluation tool. This is done through the questionnaire (Appendix A), preferably answered by the "function owner", which is the title of a person responsible for a user function. A user function is at the company referred to as functionality that is unique in the system, is clearly useful to the user as such and is triggered by the user. In general a user function incorporates not just one particular ECU, but an extensive part of the whole electronics system. In other words, in most cases a user function is a distributed function.

The next step is simply to populate the Excel-sheet with the numerical answers to the questions. The model then calculates one single value for each input and output that will be used for the Data Envelopment Analysis. These values are transferred to Matlab where the calculations are made.

5.1 WORK FLOW OF THE ANALYSIS

1. Perform a graphical 3-variable analysis of the data.

Use the output and the two most important inputs as data in order to get an overview. In this case the inputs will be hardware cost and software cost, yielding a total of three variables or evaluation criteria. Software development costs and maintenance costs are assumed equally important, hence their grading is averaged to form a compound software cost. In the Matlab model no weight constraints are introduced at this stage.

2. Calculate the corresponding efficiency scores analytically and compare with the graphical result.

3. Extend the analysis to 4-variables or evaluation criteria.

Calculate the efficiency scores analytically and compare to previous results. In this case the software cost will be split into development cost and maintenance cost, in order to provide 4-variables.

4. Introduce the weight constraints.

Calculate the efficiency scores analytically and compare to previous results. In this case there are two weight constraints that will be introduced.

5. Sensitivity analysis

Investigate how much an evaluation criterion must be improved in order for an alternative to become efficient.

5.2 CASE STUDY – USER FUNCTIONS

In this section, six user functions will be evaluated. The questionnaire created has been used to map these fictitious user functions. The user functions where for validated for relevance by industry experts. Only optional user functions have been evaluated. Their sales volume is between 5 000 and 35 000 which are reasonable numbers for optional functions. The basic characteristics of these six functions are as follows:

User function 1: It is not an advanced function and the program code is small, and it requires little ECU hardware resources. The wiring harness cost is on the other hand almost average. Due to the small code size, the software cost is low both regarding maintenance and development. Its valued added is average, however its sales volume is high (25 000).

User function 2: It is an advanced function with a large application code. The wiring harness cost is high. As a consequence, its total hardware cost is high. Due to a large application code and little code reuse, the development cost is high. The maintenance cost is average, as requirements of the function are not very prone to change. Its value added is slightly above average, due to its increase of customer satisfaction, however its sales volume is quite low (8 000).

User function 3: It is not an advanced function and the application code is quite small. On the other hand, it requires much I/O and the wiring harness cost is above average. Hence, the total hardware cost is above average. Due to a small application code the software cost is low both regarding maintenance and development. Its value added is almost average, but its sales volume is very high (30 000).

User function 4: It has an application code that is slightly above average in size. In addition, the required I/O and the wiring harness cost is above average, yielding a total hardware cost slightly above average. As the required reliability is high and little code reuse has been possible, the development cost is above average. However, due to the use of good programming style and documentation, the maintenance cost is average. Its value added is slightly above average, and its sales volume is good (15 000).

User function 5: It is a very advanced function with a large application code. It uses much I/O and a relatively expensive wiring harness. Thus, the total hardware cost is high. Its development cost is high, and the maintenance cost above average. On the other hand its value added is high, as it is a necessary function and it increases the performance. Its sales volume is very high (35 000).

User function 6: It requires advanced calculations, but the I/O and wiring harness cost is below average, yielding an average total hardware cost. The development cost is also average, however the software maintenance cost is quite low. The value added is high, as the function increases the prestige and is good compared to those of competitors. However, its sales volume is low (5 000).

5.3 CASE STUDY

5.3.1 STEP 1 – GRAPHICAL 3-VARIABLE

Software cost is calculated as the average of software development and software maintenance costs in Swedish Krona (SEK). Using the graphical representation of Data Envelopment Analysis yields the following result

	Hardware cost	Software cost	Revenue
Units:	Grade: 1-9	Grade: 1-9	Virtual SEK
UF			
1	3,1	3,4	110 000
2	6,7	5,6	41 600
3	5,3	3,3	126 000
4	5,6	5,7	84 000
5	6,9	6,7	252 000
6	4,6	4,3	34 000

Figure 3 Input and output for step 1

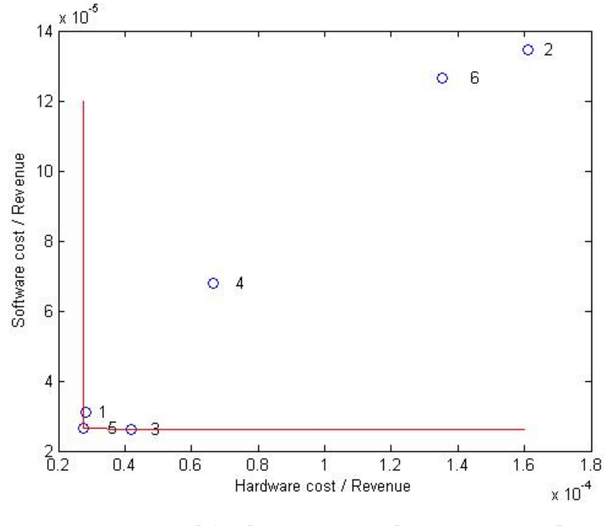

Figure 4 Graphical Data Envelopment Analysis

From the plot it can be seen that the least efficient (basically the further from the efficient frontier the less efficient) user functions are functions 4, 6 and 2, which also corresponds to the user functions with the lowest sales volumes. Functions 1, 5 and 3 are the most efficient. It is clear that user function 1 does not belong to the efficient frontier (even though it is close), and hence cannot be regarded as efficient. A point belonging to the efficient frontier is regarded efficient and the efficient frontier is defined as follows:

There is no point on the frontier line that can improve one of its input values without worsening the other.

At first sight, this leads to the conclusion that function 3 is not efficient. If the line were completely horizontal this would be true. The line connecting user function 5 and 3 is however not horizontal. A more detailed analysis shows that the line has a slightly negative slope. This is also reflected by the weights produced by the analytical analysis. A completely horizontal line is equivalent to a corresponding zero-weight. Looking at the weights, the weight for hardware cost is small, but non-zero.

A problem of this is where to practically draw the line of what is zero. In this case the weight for hardware cost for function 3 is a factor 30 smaller than the weight for function 5. In this case, from a practical perspective, function 3 should not be regarded as efficient. However, in the following analysis this will be disregarded, and the formal definition for efficiency will be employed.

It is straight-forward to explain the relative results for functions 3 and 5. The revenue of user function 5 is roughly twice that of 3. On the other hand, the software cost is just half. However, the hardware cost is more comparable in size of the two. This means that the user functions are comparable with respect to the vertical axis, but function 5 dominates the horizontal axis. If both inputs of function 3 would have been half of those of function 5, then they would have been equally efficient. Please remember that efficiency is a ratio, and that a doubling of the outputs is cancelled by a doubling of the inputs required.

5.3.2 STEP 2 – ANALYTICAL 3-VARIABLE

The analytical Data Envelopment Analysis (without constraints) yields the following efficiency scores:

UF	*Efficiency*	*Efficient*
1	0,97	No
2	0,20	No
3	1,00	Yes
4	0,40	No
5	1,00	Yes
6	0,21	No

Figure 5 Efficiency scores from step 2

As can be seen, the results correspond to those of the graphical analysis. User function 3 and 5 are both efficient; their efficiency score is 100% and all weights are greater than zero. Hence they conform to the definition of being efficient. This can also be seen by analyzing the graphical version; user function 3 and 5 both lie on the efficient frontier. It should also be noted that user function 1 has a very high efficiency. One may think that the gap of 3% to user function 3 and 5 can be more or less disregarded considering the imprecision of the data employed. However, quite large changes in the data are required in order for user function 1 to become efficient. This is discussed in step 5.

This initial analysis shows that sales volume, which directly affects the revenue, seems to be the major differentiating factor.

5.3.3 STEP 3 – ANALYTICAL 4-VARIABLE

The next step in the analysis is to extend the problem to a 4-variable problem, with three inputs and one output. The additional input is the result of splitting software cost in two; software development cost and software maintenance cost.

	Hardware cost	**Develop-ment**	**Maint. cost**	**Revenue**
Units: **UF**	Grade: 1-9	Grade	Grade	Virtual SEK
1	3,1	3,3	3,4	110 000
2	6,7	6,7	4,6	41 600
3	5,3	3,2	3,4	126 000
4	5,6	6,2	5,1	84 000
5	6,9	7,2	6,3	252 000
6	4,6	5,2	3,4	34 000

Figure 6 Input and output for step 3

This new analysis shows similar efficiency scores:

UF	Efficiency	Efficient
1	0,98	No
2	0,23	No
3	1,00	Yes
4	0,41	No
5	1,00	Yes
6	0,25	No

Figure 7 Efficiency scores from step 3

All weights of user function 3 and 5 are greater than zero; hence 3 and 5 are regarded as efficient. Little has changed compared to the first analysis. Just like before, virtual revenue (or sales volume) is the dominant factor. Still user function 1 almost 100% efficient.

Concerning the weights, the problems of almost non-zero weights outlined in step 1 are not present in this case, rather the weights are more comparable in size.

5.3.4 STEP 4 –INTRODUCING WEIGHT CONSTRAINTS

The next step of this analysis will be to introduce the two weight constraints previously outlined. So far all evaluation criteria have been assumed equally important. This is however not completely true, something that will be compensated for using weight constraints.

This analysis shows new results:

UF	Efficiency	Efficient
1	0,94	No
2	0,19	No
3	0,82	No
4	0,41	No
5	1,00	Yes
6	0,21	No

Figure 8 Efficiency scores from step 4

This time only user function 5 complies with the definition of being efficient. The biggest difference compared to the previous step is that user function 3 decreases its efficiency considerably. Before the weight constraints were introduced, the optimization algorithm could compensate for the fact that hardware cost of user function 3 (relative to its other two inputs) was large. Of the analyses conducted so far, this analysis is the most representative of the real world situation, hence it is the most correct.

5.3.5 STEP 5 – SENSITIVITY ANALYSIS

This section investigates what happens if an evaluation criterion is changed.

Sales

In the statistics used, the top 3-segment in sales is functions 5, 3 and 4. The bottom segment is 1, 6 and 2. The three analyses together show the following ranking with respect to efficiency scores of the functions: 5, 3, 4, 1, 6 and 2, which corresponds to an ordering with respect to sales volume. The exact same analysis was then made with the following sales volume:

User Function	*Sales volume*
1	25 000
2	8 000
3	70 000
4	15 000
5	75 000
6	5 000

Figure 9 Result from step 5

The top 3 segment in sales is in this case functions 5, 3, and 1. The bottom segment is 4, 6 and 2. The three analyses together show the following ranking with respect to efficiency scores of the functions: 5, 3, 1, 4, 6 and 2, which also corresponds to an ordering with respect to sales volume. This underlines the importance of sales volume.

Value added

Some examples of the implications of a change in the value added of a user function: If the value added by user function 1 is increased by 15%, then this function becomes efficient, whereas this figure is 25% for user function 3. On the other hand, user functions 2 and 4 need improvements by around 400%, which is unreasonably large. If user function 5 decreases its value added by 10% it is no longer efficient. It should be pointed out that all percentages are interpreted in the cardinal meaning; an increase of a grading with 10% means that it is regarded as 10% more important or 10% more expensive. Moreover, the percentages are approximate values. Since there is only one output, as compared to three inputs, it is clear that a change in revenue is more influential than a change in just one of the inputs.

Hardware cost

A decrease of only 10% would make user function 1 efficient, but a decrease of 35% is required for function 3. If user function 5 increases its hardware cost with 20% it is no longer efficient. For neither user functions 2, 4 and 6 a decrease of 90% is sufficient, which is the same result for development and maintenance costs as well. Even if both of the software related inputs are reduced with 90%, none of them become efficient, however their efficiency scores are increased.

Development cost

For user function 1 a decrease of 20% is required, and a decrease of 80% is necessary for function 3. User function 5 can increase its cost 75% before it is no longer efficient. Comparing these figures to those of hardware cost, it is evident that hardware cost is more influential, which should also be the case considering the weight constraints employed.

Maintenance cost

For user function 1 a decrease of 25% is required, and the corresponding figure for function 3 is 80%. User function 5 can tolerate an increase of 50%. Once again hardware costs prove more influential, but development and maintenance costs are roughly equally important.

5.3.6 ADDITIONAL STEP

Up until now, we have considered the hardware cost only to be incurred when the respective function is chosen. This is overlooking the fact that the ECU must be dimensioned to cope with any customer choice. To reflect this, the hardware cost is multiplied by the sales volume of the corresponding ECU, yielding the true total hardware cost. The revenue is still defined as the product of value added and sales of the user function. It is only in those cases that the customer actually pays for the function.

However, as the inputs are now measured in different units, the two weight constraints above cannot be used. Using no constraint implies that all evaluation factors are regarded as equally important. This is true for the relation between development and maintenance, however not completely accurate for the relations to hardware.

Let us make the following reasonable assumptions:

- User function 1 is chosen in 100% of the cases that the related ECU is mounted.
- User function 2 has its own ECU, that is, the related ECU implements only user function 2.
- User function 3 is chosen in 75% of the cases that the related ECU is mounted.
- User function 4 is chosen in 50% of the cases that the related ECU is mounted.
- User function 5 is implemented by an ECU that is always mounted.
- User function 6 is chosen in 25% of the cases that the related ECU is mounted.

These assumptions in combination with the user function sales presented previously, yields the following ECU sales:

ECU	*Sales volume*
1	25 000
2	8 000
3	40 000
4	30 000
5	75 000
6	20 000

Figure 10 ECU sales

The corresponding efficiency scores:

	Total hardware cost	**Develop-ment**	**Maint. cost**	**Revenue**
Units:	Grade: 1-9	Grade	Grade	Virtual SEK
UF				
1	3,1	3,3	3,4	110 000
2	6,7	6,7	4,6	41 600
3	5,3	3,2	3,4	126 000
4	5,6	6,2	5,1	84 000
5	6,9	7,2	6,3	252 000
6	4,6	5,2	3,4	34 000

Figure 11 Input and output

UF	Efficiency	Efficient
1	1,00	Yes
2	0,55	No
3	1,00	Yes
4	0,49	No
5	1,00	Yes
6	0,30	No

Figure 12 Efficiency scores

The weights are non-zero, hence user functions 1, 3 and 5 are efficient. The weights of these three are all of the same magnitude, which means than no evaluation criterion is neglected, providing a good overall efficiency measure. However, having weight constraints would ensure that the overall efficiency is always well reflected.

This incorporation of ECU sales has resulted in an increase in the efficiency scores of functions 2, 4 and 6. User function 5 is still efficient due to strong revenue. However, function 5 has relatively speaking become worse; it is part of an ECU that is always mounted, incurring a very high hardware cost. The worsening of function 5 has made all functions relatively better, increasing their efficiency scores. Most improvement is made by functions 1 and 2, as their portion of the ECU hardware is always used by the function. However, function 1 has in absolute numbers increased very little, as 100% as the maximum score, and the function was already before this final analysis exhibiting a high score.

6 SUMMARY AND CONCLUSION

To address this problem, the following research question was formulated: How can one quantify the resource utilization in embedded systems in the automotive industry?

In order to answer this research question a theoretical framework was created. First of all, the method of Data Envelopment Analysis was explored, and its applicability to this problem setting was explained.

Data Envelopment Analysis can calculate a compound efficiency from the input and output that a particular user function takes in. These inputs and outputs had to be selected and somehow quantified. Based on findings in research papers, the most important factors or evaluation criteria were selected. In order to quantify these evaluation criteria, the main drivers of resource consumption were identified for all of the above mentioned criteria, except for time-to-market and quality. These two factors proved too difficult to estimate with a sufficient accuracy, and had to be omitted in the analysis. By the use of a questionnaire, the performance of a user function regarding the evaluation criteria can be assessed.

The analysis was conducted incrementally, finally providing an analytical model including weight constraints to better reflect the reality. The analysis showed the importance of sales volume and value added, which together form the output revenue. The final step of the analysis was to include sales statistics not only for the user functions, but for the ECUs as well, to better reflect the cost of having to dimension for any customer choice. Taking this into account changed the results of the analysis to a certain extent.

The analysis can be said to confirm the business economics principle that price and sales are decisive factors. For instance, even if a function is very advanced or ingenious it must be sold, and it must be sold at a good price. In conclusion it can be said that the design process is a complex process, and decision support tools may be of great use. It is intuitively appealing to promote design solutions that utilize the available resources in the best way, that is, they are more resource efficient. Hence, the resource utilization framework outlined above may prove very helpful when evaluating historic design decisions, as well as constituting a guideline in current design processes.

7 FUTURE WORK

Some future work remains: the truly interesting part would be to apply the proposed framework on existing user functions, that is, to make an extensive non- fictitious case study. An extension would also incorporate more quantitative data in the analysis. For instance the framework of COCOMO (Constructive Cost Model), created by Barry Boehm [3] may be useful in order to quantify software development costs and software maintenance costs. This would also provide more accurate weight constraints compared to the assumptions made in this work. To include the aspects of time-to-market and quality could also be a future extension to this work.

ACKNOWLEDGMENTS

This work has been financially supported by the Knowledge Foundation and the Swedish Agency for Innovation Systems (VINNOVA).

REFERENCES

[1] Hoch, Detlev et al. : The race to master automotive embedded systems development. . Germany : McKinsey Company, Automotive and assembly sector business technology office, 2006

[2] Merecer Management Consulting and Hypovereinsbank, Studie-Automobiletechnologie 2010. Munchen Aug 2001.

[3] Boehm, B. et.al. Software cost estimation with COCOMO II. Upper Saddle River, New Jersey, USA: Prentice Hall, 2000.

[4] Cooper, W. Data Envelopment Analysis : a comprehensive text with models, applications, references, and DEA-solver software. Hingham, MA, USA: Kluwer Academic Publishers, 1999.

[5] Cooper, W (Editor). Handbook on Data Envelopment Analysis.

[6] Debardelaben, J. et.al. Incorporating cost modeling in embedded-system design. Design and Test of Computers, IEEE, vol. 14, pp. 24-35, July-Sept. 1997.

[7] Fornaciari, W. Cost modeling for embedded digital systems design. (Milan University of Technology, course literature 2006-2007). 2006.

[8] Majumdar, S.K. On the utilization of resources: perspectives from the U.S. telecommunications industry. Strategic Management Journal, Vol. 19, pp 809-831, 1998.

[9] Sommerville, I. Software engineering. Wokingham: Addison-Wesley, 1996.

[10] Sundararaman, K. Design for manufacturability – fault model extensions for RF circuits with an economic perspective for manufacturability. MSc thesis, New York, 2004.

[11] Wennerström, P. En matematisk modell för leverantörsurval. MSc thesis, Stockholm, 2006.

[12] Zientz, W. Electronic systems for commercial vehicles. AutoTechnology 5, pp 40-43, 2007.

[13] Penrose E.T. The theory of the growth of the firm, Oxford university press, New York, 1957

APPENDIX A

EXAMPLE OF QUESTIONNAIRE

Questions regarding user function X (UFx):

1- How many pins does the UFx use?

2 -UFx is advanced.

3 - UFx has an application code that is large.

4 - UFx has an application code that is complex.

5 - UFx requires many calculations.

6 - UFx requires advanced calculations.

7 -UFx uses many variables.

8 - UFx uses many nested or recursive functions.

9 - UFx requires much parameterization.

10 - UFx is distributed/interdependent.

11 - The required reliability of UFx is high.

12 - Re-use of software has not been possible with UFx.

13 - The hardware platform that UFx uses is often upgraded.

14 - UFx is prone to be affected by new regulations (for instance regarding emissions).

15 - UFx is old.

16 - Poor programming style and low quality program documentation have been used.

17 – The wiring harness of UFx is long.

18 - The wiring harness of UFx is located in a harsh environment (e.g. in the engine house).

19 - How necessary is UFx?

20 - How much does UFx increase customer satisfaction?

21 - How does UFx affect the parameter “prestige” of the Product Identity?

22 - How does UFx affect the parameter “performance” of the Product Identity?

23 - How does UFx perform compared to similar user functions among competitors?