Supporting the Utilization of a Platform Approach in the Engineer-to-Order Supplier Industry

Samuel André
ABSTRACT

Manufacturing companies are continuously faced with requirements regarding technology novelty, shorter time to market, a higher level of functionality, and lower prices for their products. This is especially true of suppliers that develop and manufacture highly customized products within the automotive industry. It is not uncommon that a request for a new product or subsystem goes out to several suppliers and that the one that can deliver the product most quickly and at the lowest price receives the contract. It is therefore vital for any supplier to answer to quotation requests rapidly and with a high level of precision while also ensuring that company assets are used efficiently. Other issues that apply to suppliers in the automotive industry are heavily fluctuating requirements during development projects, each customer’s individual preferences, and the ever-changing interfaces with the OEM product with which the supplier’s product is to be integrated. Platform strategies have been widely accepted in industry to serve a wide product variety while maintaining business efficiency. However, the challenge of applying a platform strategy at the supplier level in the face of the reality described above has not been fully investigated. Platform approaches tend to require a focused development of the platform, which in turn requires some knowledge about which future variants are to be derived from the platform. The research presented in this thesis investigates the state of practice in industry regarding the challenges, needs, and current use of platforms. To respond to the identified need, a platform approach is proposed that expands the scope of what a product platform has traditionally contained. This is undertaken to aid in the development of highly customized products when physical modules or component scalability does not suffice. The platform approach provides a coherent environment for heterogeneous design assets to be used in product development, supporting both the activity of designing and off-the-shelf solutions. The approach is based on identifying and modelling generic product items that are associated with descriptions governing their design. By describing the outcome of technology and product development like finished designs, design guidelines, constraints, etc., in a standardized format, the platform continues to evolve. To aid in using the platform approach, a support system called Design Platform Manager is introduced at a company active as a second-tier supplier in the automotive industry. The system enables the creation of generic product items that can be structured, instantiated, and associated with descriptions, which aids in realizing product variants. The aim of the platform approach and tool is to support the quotation and continued design processes by identifying valid knowledge to use as circumstances, such as requirement changes or new design iterations, warrant. The support tool and overarching model have been evaluated by company representatives, who reported good results.

Keywords: Product Development, Engineering Design, Quotation, Customization, Supplier, Platform, Design Reuse, Adapt, Concurrent Engineering
ACKNOWLEDGEMENTS

This thesis was carried out in the Department of Product Development, Jönköping University, Sweden. I would like to express my gratitude to my supervisor, Fredrik Elgh, who gave me the opportunity to pursue a doctoral degree. You have given me the freedom to try my own ideas and supported me with input, fruitful discussions, and your invaluable ability to see the bigger picture.

This work would not have been possible if it were not for industrial collaborators and external funding. I would like to thank the Swedish Agency for Innovation Systems, VINNOVA, for funding the ChaSE project, in which this research has been conducted. Thanks to Thule, GKN Aerospace, and Axelent Engineering for participating in the research project and making partial contributions to the work. A special thanks goes to Kongsberg Automotive for permitting me to have access to their facilities and personnel, which were absolutely essential to the successful completion of this thesis.

Thank you to my co-supervisor Roland Stolt for constructive criticism, valuable input, and careful proofreading in the writing process. Great thanks also go to Joel Johansson for the incredible level of knowledge of computer programming that he was so willing to share with me. Thanks also go to Tim Heikkinen and Morteza Poorkiandy for fruitful and challenging discussions. It is by challenging our beliefs that we develop.

I would also like to thank additional friends and colleagues at the School of Engineering, especially in the Department of Product Development. It is a joy to work together with you in this environment.

Finally, I would like to offer my thanks to my family, especially my wife Annika and our three children, Olle, Svante, and Tage, for the continuous encouragement and support which you have given me and for the many perfectly valid reasons to not be at work. Without you, life would make much less sense and not be anywhere nearly as enjoyable.
APPENDED PAPERS

The following papers constitute the foundation of this thesis:


Work distribution
Samuel André performed the data analysis and wrote the paper. Fredrik Elgh synthesized the hypothesis model. Roland Stolt and Fredrik Elgh supported by proofreading. All included authors contributed to the research design and data collection.


Work distribution
Samuel André wrote the paper, synthesized the main parts of the theory, and developed the computer support tool. Roland Stolt and Fredrik Elgh supported in the synthesis of the theory and in proofreading.


Work distribution
Samuel André wrote the paper, synthesized the main parts of the theory, developed the computer support tool, and designed the evaluation approach. Roland Stolt and Fredrik Elgh supported in the synthesis of the theory, conducting the evaluation, and proofreading.


Work distribution
Samuel André wrote the paper and developed the computer support tool. Samuel André and Fredrik Elgh synthesized the theory, conducted the evaluation, and executed the study. Fredrik Elgh supported by proofreading.
ADDITIONAL PAPERS

The following papers contribute partially to the results but do not form part of the thesis foundation.


ABBREVIATIONS

AD – Adaptable Design
B2B – Business-to-Business
BOM – Bill of Material
CAD – Computer-Aided Design
CC – Configurable Component
CE – Concurrent Engineering
CODP – Customer Order Decoupling Point
CTO – Configure to Order
DE – Design Element
DMM – Domain Mapping Matrix
DP – Design Platform
DPM – Design Platform Manager
DS – Descriptive Study
DSM – Design Structure Matrix
ETO – Engineer to Order
GPI – Generic Product Item
MTO – Modify to Order
PCB – Printed Circuit Board
PD – Product Development
PDM – Product Data Management
PLM – Product Lifecycle Management
PS – Prescriptive Study
PVM – Product Variant Master
RFQ – Request for Quotation
SBCE – Set-Based Concurrent Engineering
SC – Success Criteria
TD – Technology Development
TRL – Technology Readiness Level
CONTENT

4.1.1 Platform challenges and prerequisites in ETO industries ........................................ 32
4.1.2 Platform abstraction levels .................................................................................... 33
4.1.3 A platform model hypothesis for suppliers of customized systems .................... 33
4.2 Paper 2 .................................................................................................................... 33
4.2.1 A conceptual platform description ........................................................................ 34
4.2.2 Introducing design elements as carriers of design knowledge ............................... 34
4.2.3 Design element structuring and identification ....................................................... 35
4.3 Paper 3 .................................................................................................................... 36
4.3.1 A platform approach to support suppliers of customized systems ....................... 36
4.3.2 Supporting the use and expansion of an applied design platform .......................... 37
4.4 Paper 4 .................................................................................................................... 38
4.4.1 A refined and formalized platform approach and model ....................................... 38
4.4.2 The DPM and PDM ............................................................................................ 38
4.4.3 Platform expansion and use .................................................................................. 40

RESULTS ....................................................................................................................... 43
5.1 The first case evaluation ............................................................................................ 43
5.1.1 Evaluation analysis .............................................................................................. 43
5.2 The second case evaluation ...................................................................................... 44
5.2.1 Evaluation results ................................................................................................. 45

DISCUSSION .................................................................................................................. 47
6.1 An enabler for increased responsiveness to fluctuating requirements ....................... 48
6.2 Discussing the research questions .......................................................................... 49
6.2.1 RQ1: What is the current state of the utilization of product platforms for suppliers
developing highly customized products in B2B environments? .................................. 49
6.2.2 RQ2: How can a platform approach be conceptualized to support customization for
a company active in this environment? ....................................................................... 50
6.2.3 RQ3: How can such a platform approach be formalized and applied in practice? ...... 50
6.3 Verification and validation of research ..................................................................... 51

CONCLUSIONS ............................................................................................................ 53
7.1 Future work ............................................................................................................. 54

REFERENCES .............................................................................................................. 57
APPENDED PAPERS .................................................................................................... 63
INTRODUCTION

Manufacturing companies are continuously faced with requirements regarding technology novelty, shorter time to market, higher level of functionality, and lower prices for their products. This applies especially to the reality of companies that develop and manufacture highly customized products within the supplier industry. The traditional view of the product lifecycle introduces the customer into the sale and distribution phases. This type of business is often concerned with identifying and transferring customer needs into fixed specifications, which guides the product development of end consumer products. The engineer-to-order (ETO) supplier industry differentiates itself from this approach in that the customer is already involved in the scoping and quotation stages. This industry cannot work as the end customer sector can, since requirements are expressly set by the customer, which often is an original equipment manufacturer (OEM) or another supplier. It is not uncommon for products to be developed in cooperation with the customer and for projects to extend over several years. During the cooperative development phase, requirements often fluctuate. This has been investigated in the automotive industry (Almefelt, Berglund, Nilsson, & Malmqvist, 2006), and is said to be a natural process since knowledge is gained and prerequisites change throughout the project. Andersson (2003) states that misunderstandings are one of the sources of requirement changes, often due to the fact that the requirements are not specified clearly in the first place. These changes often stem from the complex interplay between various suppliers using the same interfaces as inputs to their development processes. When solutions require changes to the interfaces, other suppliers and thus their solutions are also affected. This in turn requires changes in affected subsystems or changes to the requirements themselves. On these occasions, it is crucial to manage requirement fluctuations and have an effective method to adapt to the ever-evolving situation. There are different views of the dynamics of requirements. Sutinen, Almefelt, and Malmqvist (2000) indicate that it is desirable to form a fixed list of the requirements to guide the product development (PD) process and to reduce risk (Halbleib, 2004). A slightly different view is given by Almefelt et al. (2006), who report that requirements should be established early but that stakeholders should be open-minded about changes. An understanding that
differs from these is the view of set-based concurrent engineering (SBCE), which states that late decision making and late detailing of the requirements – i.e. keeping a large design space – is desirable, since such a strategy leads to a steady convergence of solutions (Land, 1982). In Raudberget (2012), improvements in product performance, product cost, and the level of innovation are shown through the implementation of SBCE.

Customization refers to abilities and strategies that aim at the design and manufacture of tailored products for individual customers. Depending on precisely when the actual customization starts, four different business models can be identified: engineer to order (ETO), modify to order, configure to order, and select variant (Hansen, 2003). For the latter two, product platforms have substantial success as enablers of efficient customization.

One factor that amplifies the challenge for suppliers of applying a platform approach is the splitting of technology development (TD) and PD. TD often has a long-term goal of supplying a future market that is to some extent uncertain with new technology, whereas PD has a more short-term goal of fulfilling specific customer requirements. It has been suggested that the splitting of TD and PD can decrease risk in customer-focused projects (Säfsten, Johansson, Lakemond, & Magnusson, 2014). Lakemond, Johansson, Magnusson, and Säfsten (2007) emphasize that technology transfer must take place in a physical handover and that an understanding of one another’s work must be developed. Eldred and McGrath (1997), meanwhile, propose a high-level process to move from TD to PD. All these authors share the view that a technology transfer step is needed in the interface between TD and PD.

Platform strategies, as enablers of customization, have been widely accepted in industry to serve a wide range of product varieties while maintaining business efficiency. Early descriptions of product platforms focused on efficiently providing the market with a wide product variety while keeping the internal variations as low as feasible (Meyer & Lehnerd, 1997). Platforms have also been a way to reach different customer segments efficiently by achieving commonality in product components and interfaces. Balancing the tradeoff between commonality and distinctiveness is the key to success here (Halman, Hofer, & Vuuren, 2003). Recent research has focused on platforms with a more abstract definition; these platforms aim to reuse more of the skills and knowledge (i.e. assets) created in a company. From that perspective, Johannesson (2014) questions whether companies can afford not to apply a platform. However, how to apply a platform strategy at the supplier level in the face of the reality described above has not yet been fully investigated. Platform approaches tend to require a focused development of the platform and late customer involvement, which requires some knowledge about which future variants are to be derived from the platform. This is commonly done by companies that develop products for the end consumer, such as vehicle manufacturers. There are, however, limitations to developing such platforms for ETO-oriented suppliers in business-to-business (B2B) environments, since the business model forces them to develop highly customized solutions for every new project due to the limitations in their ability to foresee future customer requirements.

1.1 Aim and goal

The aim of the research is to make companies acting as suppliers and developing highly customized products more responsive to fluctuating requirements during the scoping, quotation, and subsequent development processes. The goal of this thesis is to investigate and support the current state in industry regarding platforms as a means to managing changes in requirements. The current state also includes the needs, challenges, and prerequisites of companies with these characteristics. A secondary
goal is to explore how a platform approach and support tool can be formulated to make it possible for a company to gain the benefits of what platforms can enable. This aim and goal are to a significant extent focused on contributing to industry. The scientific contribution is to add to the body of knowledge regarding the use of platforms in settings where platforms traditionally have been difficult to implement, to expand the boundaries of platform definitions, and to exemplify the application of an alternative to the traditional product platform approaches in the literature.

1.2 Research focus

The challenges for the suppliers described above regarding the splitting of TD and PD and the development of artifact-based product platforms create difficulties in their taking advantage of platform strategies. Product platforms are well established in the literature (Simpson, Jiao, Siddique, & Hölttä-Otto, 2014); more recently, technology platforms have also been of interest (Johannesson, 2014). However, there are not many examples in the literature that describe a coherent platform approach and the application of such an approach to supplier companies, given these prerequisites. An important issue in industry is the loss of knowledge created by focusing on specific product instances without the assistance of a suitable format to gain the benefits of formerly created knowledge in future PD projects. It is crucial that the key knowledge created during both TD and PD be described in a way that both enables easy adaptation during customization and that can be generalized in order to be used in future projects, an approach that, if successful, will expand the company’s overall knowledge. The research presented in this thesis focuses on how a company active in the supplier environment can gain the benefits of a platform approach when a component-based product platform is not a realistic option.

In order to visualize the line of argumentation and logic for the chosen research focus, a reference model was created, following Blessing & Chakrabarti (2009), as shown in Figure 1. The figure also illustrates the intended line of effect that the introduced support is expected to have on the company’s competitive edge. The key element that is intended to be supported is the possibility for the company to utilize a platform, which in turn should increase the level of design reuse. The chief effect should appear in the efficiency of the customizations performed by a company, which should improve the company’s competitive edge. The plus and minus signs on the arrows in Figure 1 are intended to show how the different concepts are assumed to increase or decrease one another.

1.3 Research questions

This broad research question is the overall focus of this work:

“How can suppliers of customized systems be supported in using a platform approach?”

To respond to this question, it has been broken down into the following three sub-questions:

- **RQ1**: What is the current state of the utilization of product platforms for suppliers developing highly customized products in B2B environments?
INTRODUCTION

This question regards the state of practice in ETO companies in terms of product platforms. The question concerns if and to what degree companies engage in platform development, what challenges there are, and under which circumstances and with which prerequisites their platforms are created and used.

- **RQ2**: How can a platform approach be conceptualized to support customization for a company active in this environment?

A key assumption which is supported by the literature (Högman, Bergsjö, Anemo, & Persson, 2009) is that ETO suppliers cannot fully apply, a traditional component-based product platform. This research question aims to investigate a suitable format and model to be used for such a company and to report on what building blocks it could contain.

- **RQ3**: How can such a platform approach be formalized and applied in practice?

To make use of a model at a higher level, it must be supported, and this question concerns how such support could be designed. The second part refers to applying the support and the validation of the results.

![Diagram](image)

*Figure 1. Reference model according to Blessing and Chakrabarti (2009), visualizing the line of argumentation and logic behind the chosen research focus and the intended effect of the introduced support.*
1.4 Scope and delimitations

This research focuses on a specific group of industrial companies that act as suppliers of highly customized products. The results are expected to have the possibility of being generalized across a broad range of companies, but this element is not specifically evaluated within the scope of this thesis.

PD concerns many artifacts, processes, and people. Organizational and management issues coupled with the approach and support introduced are not in the focus of this thesis. The introduced support is, however, intended for certain people working within the design field, such as design engineers and technical project managers.

1.5 Research project

The research conducted within the scope of this thesis is part of a research project called ChaSE, which is a shortened form for Challenge Fluctuating and Conflicting Requirements by Set-Based Engineering. The project’s aim is to determine how companies can develop adaptable solutions to respond efficiently to fluctuating and conflicting requirements. The project is a joint effort between Jönköping University, The Swedish Agency of Innovation Systems (VINNOVA), and four companies developing customized products.

1.6 Outline of the thesis

Chapter 1 introduces the work presented in this thesis. It contains the background, problem area, and delimitations and presents the research questions that are answered in the scope of this research.

Chapter 2 presents the frame of reference, which is an assortment of literature that has been studied to identify best practices, the current state of industry, and fundamental theories. The chapter concludes with a summary that identifies the research gap that this thesis strives to fill.

Chapter 3 outlines the research methodology applied in this work. It presents methods and models in a generic manner and their application in this work.

Chapter 4 summarizes the results from the appended papers and points to the progression and evolution of the research in the various papers.

Chapter 5 presents two evaluations of the results that have been found in this research.

Chapter 6 outlines the discussion, focusing on the results in light of the literature, the validity of the research, and the research questions.

Chapter 7 briefly summarizes the main conclusions and takeaways from the thesis as a whole. It also summarizes future work that might be conducted.
INTRODUCTION
This chapter presents a selection of research that is essential for or closely related to the research subject. Figure 2 shows how the fields are related and their importance. The figure also shows the area to which this thesis makes a contribution.

Figure 2. ACR diagram (inspired by Blessing and Chakrabarti (2009))
2.1 Product development

PD is defined as transforming a market opportunity to meet a customer need and the strategic goals of the company. This is achieved through a set of coherent activities that interact with one another (León & Farris, 2011). Ottoson (Holmdahl, 2010, p. 51) offers the following definition of PD’s aim (author translation from Swedish): “The aim of product development is to increase the quality of life for one human being without decreasing it for another.” However, the main idea is to use PD as the primary means of moving from concept to income. Several development process models have been tested and developed over the years, with the sequential model being the norm (Engwall, 2004). There is no single process model describing the PD process in a generic fashion; however, an attempt has been made by Ulrich and Eppinger (2011), as shown in Figure 3.

![Figure 3. A generic product development process according to Ulrich and Eppinger (2012)](image)

2.2 Customization

In today’s marketplace, customers expect products to satisfy their particular needs and, in many cases, to cost as little as possible. This stresses the need for strategies for customizing products at a low cost. Customization refers to abilities and strategies that aid designing and manufacturing tailored products for an individual customer. Simpson (2004) states that customers can no longer be lumped together in an enormous, homogenous market. They are individuals whose specific wants and needs can and must be ascertained and fulfilled. New products must differ from what is already on the market while meeting customer needs more completely than ever before. He proposes product families as a primary enabler, describing two basic approaches for the design of product families to achieve efficient customization: (1) Top down, which occurs when the company strategically manages and develops a family of products based on a product platform and its derivatives; (2) Bottom up, which applies when a company redesigns a group of products to standardize components and improve economics of scale. Depending on a company’s customization strategy, the way a product is specified differ. Suppliers with an ETO business approach often find themselves in an environment in which several intermediate steps that involve different stakeholders, other suppliers, and an OEM differentiate them from companies focused directly on the end customer. This introduces several interfaces and stakeholder interests that the ETO supplier must manage. Holistic research in this area, taking all or at least several of these perspectives into consideration, however, remains scarce. Tuli and Shankar (2015) describe lean-in collaboration between supplier and OEM and review the existing literature on supplier, OEM, and
customer integration. One interface towards the customer which is central to customization is the customer order decoupling point (CODP). This is normally defined as the point in the flow of goods at which forecast-driven production and customer order-driven production are separated (Giesberts & Tang, 1992). The CODP is often viewed as a point on a one-dimensional line that can also be coupled with the level of customization. Hansen (2003) uses four categorizations: engineer to-order, modify to order, configure to order, and select variant. Wikner and Rudberg (2005), however, propose a two-dimensional categorization for companies in the product realization process. The level of customization has often focused on production, with engineering viewed as taking place before production. These authors propose an approach that takes into consideration the engineering adaptation that occurs for each customer order, including both the engineering and production dimensions. Figure 4 shows a two-dimensional CODP with both the engineering and production dimensions. It should, however, be noted that only some of the points in the two-dimensional area are feasible in real-world contexts.

One tradeoff that is challenging occurs between flexibility and rigidity when it comes to customization. Fredriksson and Gadde (2005) investigated the implementation of module assembly units (MAUs), a combination of build-to-order and mass production, which has been shown to be a feasible strategy in order to balance the tradeoff. Fredriksson and Gadde also emphasize that the range of options for the customer must be constrained to control the number of variants. This is also stressed by Michael, Kilian, and Lucienne (2007), who argue for the necessity of constraints in the customization offer in order to decrease problems with customers’ insisting on custom modifications which in turn have a negative effect on the company’s financial. Other tradeoffs are investigated by Squire, Brown, Readman, and Bessant (2006), who used a quantitative approach. Their results show that customization has a significant effect on manufacturing cost, does not have a significant effect on quality, increases both time to delivery and average lead time significantly, does not have a significant effect on the reliability of delivery times or the percentage of products delivered on time, and does not reduce volume flexibility.

<table>
<thead>
<tr>
<th>Engineering dimension (ED)</th>
<th>Production dimension (PD)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>MTO&lt;sub&gt;PD&lt;/sub&gt;</td>
</tr>
<tr>
<td></td>
<td>ATO&lt;sub&gt;PD&lt;/sub&gt;</td>
</tr>
<tr>
<td></td>
<td>MTS&lt;sub&gt;PD&lt;/sub&gt;</td>
</tr>
<tr>
<td></td>
<td>ETO&lt;sub&gt;ED&lt;/sub&gt;</td>
</tr>
<tr>
<td></td>
<td>ETO&lt;sub&gt;ETD&lt;/sub&gt;</td>
</tr>
</tbody>
</table>

Figure 4. The two-dimensional CODP adapted from Wikner and Rudberg (2005)

2.2.1 Mass customization

Mass customization is a concept found in automotive, clothing, and computer manufacturing, along with the food industry, electronics, and mobile phones (Fogliatto, da Silveira, & Borenstein, 2012); it has two definitions (Hart, 1995). The first is the visionary definition: “The ability to provide the customers with anything they want profitably, any time they want it, anywhere they want it, any way they want it.” The second and more realistic definition is “the use of flexible processes and organizational structures to produce varied and often individually customized products and services at the low cost of a standardized, mass production system.” Hart (1995) continues by defining and describing the four
The frame of reference for mass customization is illustrated in Figure 5. Different types of companies pursuing mass customization are highlighted, including mass production, small series production, one-of-a-kind production, and mass customization. This diagram, adapted from Hvam, Mortensen, and Riis (2008), shows the progression from mass production to mass customization, with intermediate stages representing small series and one-of-a-kind productions. The diagram emphasizes the continuity and progression of strategies as companies adapt to different market demands and customer needs.

A detailed framework for designing for mass customization is presented in Tseng, Jiao, and Merchant (1996). The authors outline a process, emphasizing the creation of product family architectures in order to conduct family-based design and integration of departments within the company. Swaminathan

---

**Figure 5 Different types of companies pursuing mass customisation; adapted from Hvam, Mortensen, and Riis (2008)**
(2001) presents a framework for standardization strategies that enable mass customization by emphasizing parts standardization, process standardization, product standardization, and procurement standardization. The author also describes the concepts of modular designs and processes and how they interact and can be combined. Considerations that must be taken into account before adopting standardization strategies are presented in Figure 6.

![Figure 6. Choosing standardisation strategies; adapted from Swaminathan (2001)](image)

2.2.2 Product platforms

Several definitions of product platform can be found in the literature; depending on which definition is chosen, a product platform can be many different things. The four most commonly used definitions are as follows:

- The collection of assets [i.e., components, processes, knowledge, people and relationships] that are shared by a set of products. (Robertson & Ulrich, 1998, p. 20)
- A collection of common elements, especially the underlying core technology, implemented across a range of products. (McGrath, 1995, p. 39)
- A group of related products that is derived from a product platform to satisfy a variety of market niches. (Simpson, Siddique, & Jiao, 2006, p. 3)
- A set of common components, modules, or parts from which a stream of derivative products can be efficiently developed and launched. (Meyer & Lehnerd, 1997, p. 7)

Based on these definitions, it is clear that a platform can be described on many levels of concretization or abstraction. This is also reflected among suppliers, as shown by Högman et al. (2009), in which the company platform description is described using levels of abstraction. The authors investigate whether a platform strategy is applicable for reuse purposes to a supplier in the aerospace industry. The authors conclude that a modular platform is not seen as feasible in such a company, since most of the reuse is found on a higher level of abstraction. Halman et al. (2003), however, state that that companies in industry have not been keeping pace with research on platforms due to a lack of tools. The authors also identified a disjointed view of platforms in industry, which is exemplified by the four definitions presented earlier. The research in the field of product platforms has generally adopted an artifact-
oriented approach, supported by the evolution in PLM and configuration systems; i.e. the rules have been defined and organized in accordance with a product structure. However, variant management remains a challenge for many of the conventional PLM systems on the market. Bruun, Mortensen, Harlou, Wörösch, and Proschowsky (2015) describe a visual architecture representation and its operational handling in a PLM system intended to enable companies to overcome the challenging situation of identifying common modules when developing product families. The article describes the uses of interface diagrams which are converted via XML and uploaded to a PLM system.

Two approaches for creating platforms are found in the literature:

- **Module-based (configurable):** by carefully listening to customer needs and adopting a well-planned product architecture, success can be achieved by applying a module-based platform approach. The module based platform can be of two kinds:
  - Integral: Several functions share a physical element.
  - Modular: Each function is delivered by a separate physical element.
- **Scale-based (parametric):** designed so that a number of design variables can be varied and allow the design to stretch or shrink. This is suitable for optimization due to its continuous nature.

Modularity is proposed by several authors as the best enabler for customization (Hsuan Mikkola & Skjøtt-Larsen, 2004; Hundal, 2012; Hvam et al., 2008; Swaminathan, 2001). Different modularity strategies can be found in the literature (Hvam et al., 2008). “Component sharing modularity” and “component swapping modularity” use the same components to span both product variants and product families. With “sharing”, the same components are used across product families, while “swapping” involves introducing variants into a product family by adding small components. “Cut-to-fit” modularity has the property of parameterization, in which some of the modules can be adapted by modest changes, such as their dimensions. “Sectional modularity” means that modules can be combined freely (like LEGO bricks), by using the modules’ interfaces. “Bus modularity” – also called a bus platform – means that a platform is developed on which components can be mounted. These definitions of modularization types, however, intersect with the definitions of platform type (i.e. modular or scalable) since the “cut-to-fit” type includes scalability. This overlap and the inconsistency across definitions shows a disjointed view even in the literature.

A successful platform must be flexible, according to Suh, De Weck, and Chang (2007). The authors outline a process for designing a flexible platform that uses a combination of quantitative analysis and engineering knowledge. Platform flexibility by abstraction of the platform constructs has proven to be one successful way of managing the changing requirements. Pakkanen, Juuti, and Lehtonen (2016) describe a process that emphasizes the reuse of assets, given that there are limitations in existing designs. The method is directed at companies in the project business that deal with a high level of customization.

One risk with using a product platform approach is the tradeoff between commonality and distinctiveness (Robertson & Ulrich, 1998). Examples from the car industry show that lower-end models can cannibalize on higher-end models if distinctiveness is not pronounced enough. However, Simpson (2004) reports that by sharing underbodies, a company in the car industry can find a 50% reduction in capital investments. In the 1990s, car manufacturers that applied a platform strategy gained a 5.1% market share, while those that did not saw a 2.2% loss of share. Another tradeoff occurs between increased development efforts for the initial platform and uncertainty as to whether the right platform has been
chosen for the development of a sufficient number of derivatives to recover the added expenses (Halman et al., 2003). Cabigiosu, Zirpoli, and Camuffo (2013) provide evidence that the inherent complexity of automobiles reduces the opportunities for modularity among suppliers. They also investigate the interface definition process in the automotive supplier industry, concluding that interfaces diverge significantly and that the definition processes are neither technologically determined nor the mere result of product architecture choices. The study also points to other factors such as knowledge scope and capabilities that affect the interfaces more than architectural choices.

2.2.2.1 Models for platforms

For a platform to be described in the context of engineering design, a model of the platform is highly suitable. The term "model" is usually used to describe a simplification of reality. Therefore, a platform model, as defined in this thesis, is a simplified description of some aspects of a platform. Some approaches to create a platform model have been presented over the years, the highlights of which are presented here.

The product variant master (PVM) is a tool that has been described in several scientific articles and books; it can be used to model to some extent a product platform (Hvam et al., 2008). The main aim of the PVM is to map the product variants in a company and couple them with a generic product architecture to create a foundation for introducing configuration systems. The generic product architecture is referred to as the “part-of” structure and visualizes parent-child connections between systems, subsystems, and components. Coupled to the “part-of” structure is the “kind-of” structure that describes the nature of the different types of variants. In order to describe all variants, class-responsibility-collaboration cards (CRC) cards are used as described in Mortensen, Hvam, and Haug (2010). The product variant master and CRC cards are said to bridge the gap between domain experts and IT developers.

Bruun, Mortensen, and Harlou (2013) present an approach for using a visual product architecture model in combination with a PLM system to support the development of modular product families. The motivation for their research is based on the consequences of not having proper documentation or aligned data when developing modular architectures. One of the main aims of this approach is to support the designer in finding and reusing existing modules. Otto et al. (2016) propose a generic 13-step process for designing product platforms, which is based on the existing literature. The 13 steps are then associated with various platform development methods used in several industrial companies.

Another methodology that can be termed a platform model is the configurable component (CC) concept (Claesson, Rosvall, & Johannesson, 2005). Instead of modeling the connections between physical parts and modules, as in the PVM, the connections from functional requirements to design solutions are mapped. The modeling technique uses a number of object types, such as functional requirements, design solutions, and constraints. These create a hierarchy that starts from the main functional requirement, passes through design solutions and derived functional requirements, and eventually reaches the level at which the design solution can be embodied in a component. Levandowski, Raudberget, and Johannesson (2014) propose a methodology to model a platform in the early phases of development using the CC concept and SBCE.
2.2.2.2 Technology platforms

Component- and module-based product platforms have been found not to fit every business model by Högman et al. (2009). However, companies in ETO industries need ways to harvest the fruits of a platform definition that can give them advantages that are similar to a component-based platform. Johannessson (2014) even asks whether companies really have a choice about implementing a platform, since platforms can exist on all levels ranging from standard components to knowledge and relationships (Robertson & Ulrich, 1998), making them useful for all kinds of companies. Cooper (2006) suggests that one deliverable from technology development could be a technology platform, a claim further investigated by Högman (2011). Högman presents a technology platform definition that is not connected to a specific implementation, as a product platform is, but rather consists of design knowledge, product concepts, applied technology, and technological capabilities that support product realization. In Högman and Johannessson (2013), the authors investigate the application of a stage-gate process to aid in managing technology development. Levandowski, Forslund, Söderberg, and Johannessson (2012) analyse platform approaches from a PLM perspective and assess how well they cover the needs of the aerospace industry. Corin Stig and Bergsjö (2011) report in their paper that there is a need to access knowledge and ways to share knowledge of both new and mature technologies. The authors describe two tools to support these aims: a technology platform wiki to share information regarding key technologies and reusability, and a lightweight online checklist system to assure the maturity and platform compatibility of technologies.

2.2.3 Adaptable design

Adaptable design (AD) is an emerging field in design science. It has characteristics similar to customization, but customization is merely one part of the AD concept. AD takes the entire product lifecycle in consideration, emphasizing adaptability from product design through the use of the product and ending with recycling. In this way, AD considers product functionality, quality, manufacturability, cost, and environmental performance. Unlike traditional platform and modular approaches to adaptability with interfaces, designs in the AD context should be adaptable in order to adapt to new changes required to meet customer, environmental, and market needs. Since PD often leads to both a product design and a product, AD is defined for both outcomes (Gu, Hashemian, & Nee, 2004). In the field of adaptable engineering, a method for identifying the optimal product by considering changes in requirements, configurations, and parameters in the entire product lifecycle is proposed by Xue, Hua, Mehrad, and Gu (2012).

- **Design adaptability** refers to the ability of a product to be modified (the design) to produce another product.
- **Product adaptability** refers to the product’s ability to be adapted by the user to fulfil different purposes with one product, and thus replacing the need for multiple products.

Two types of adaptability can be identified:

- **Specific adaptability** describes the ability of a product design to be adapted for potential applications that can be foreseen at the time the product is initially designed.
- **Generic adaptability** describes a product’s ability to adapt when the environment changes unpredictably or unanticipated requirements arise; this is important in more complex systems.
The authors describe the different processes that are used to achieve design and product adaptability. The processes are broken down further in Gu (2004), who adds the concept of flexible interface systems. Zhang, Chen, Xue, and Gu (2014) describe the three different product architectures used in AD: closed, semi-open, and open architectures. The distinction is made based on the level to which a third party can continue to develop the product.

2.3 Requirement management

To understand why a product is developed, the purpose it serves, and what it should achieve, some statements must be made about the outcome of PD. Requirements tell the engineer what a product should actually do. They are the quantifiable and testable link(s) from customer need to a product’s functional attributes. Halbleib (2004, p. 1) defines requirements as “the agreed-upon facts about what an application or system must accomplish for its users.” A requirement statements answerers the question of what the system should do or what the product must do. However, it should never state how the product should fulfill the requirement. Whether PD is initiated by a proposal from a product planning process or a specific customer order, it is necessary to clarify the task before beginning development. In the clarification phase, information about the requirements that must be fulfilled by the product and about the existing constraints and their importance are gathered. This phase leads to a requirements list that focuses on, and is attuned to, the interest of the design process and subsequent working steps. The conceptual design phase should be based on this document, which is updated continually (Pahl & Wallace, 2007).

Requirement traceability links requirements to realizing components and vice versa. A requirement is traceable if one can detect (1) the source that suggested the requirement, (2) the reason why the requirement exists, (3) what other requirements are related to it, (4) how the requirement relates to other information such as function structures, parts, analyses, test results and user document, (5) the decision-making process that led to derivation of the requirement, and (6) the status of the requirement (Sutinen et al., 2000).

Nilsson (2004) ends his doctoral thesis by offering some ways to enable success in requirement management:

- Requirement and concept modelling: Describes the development of a computer tool.
- Incorporating customer needs through quality function deployment (QFD): Links QFD to the hierarchical breakdown of requirements, and proposes dividing a master matrix QFD into smaller ones to improve manageability.
- Stakeholder integration: Proposes to involve all stakeholders’ needs, not just the customer voice; a model is developed.
- Product and process modelling: Describes how to link product and process development by a model.
- Structuring manufacturing requirements: Proposes a way of structuring manufacturing requirements in a tool.
2.3.1 Changing and conflicting requirements

The requirement specification aims to describe product functions and constraints in the PD process and to offer a unified impression to all stakeholders involved in the project (Pahl & Wallace, 2007). The dynamic nature of requirements often results in changes or new requirements being added while others are dropped; this complex process has been investigated in the automotive industry by Almefelt et al. (2006). The authors state that it is a natural process, since knowledge is gained and prerequisites change throughout the project. Different customer groups can have different customer needs which can give rise to conflicting requirements (Jiao & Chen, 2006). Requirement freeze is a term found in the literature to describe the point at which requirements are no longer allowed to change. There are different views of when or even if this point should occur:

- It is desirable to form a fixed list of requirements to guide the PD process (Sutinen et al., 2000) and to reduce risk (Halbleib, 2004).
- Requirements should be established early, but stakeholders should be open-minded about changes (Almefelt et al., 2006).
- Late decision making and late formation of the requirements (i.e. keeping a large design space) are desirable, since such strategy leads to a steady convergence (Land, 1982). This is one of the key elements of SBCE (Raudberget, 2012).

Andersson (2003) identifies general factors for changing requirements:

- Requirements are changed through a development project due to competitor, product, market, and project evolution.
- Requirements are consciously and unconsciously reprioritized throughout the development project because of, among other things, the knowledge gained, approaching toll gates, and responsible actors.
- Misunderstandings often occur due to the fact the requirements are not specified clearly.

Since requirements tend to change or be dropped from or added to a project, they must be formally managed in some fashion. Changes in requirements also need to be reflected in product definitions, lifecycle systems, and property models (Sutinen et al., 2000). Stechert and Franke (2009) propose a model that links requirements with functions, which enables tracing of changes from impact to source. Nilsson (2004) reports on a case study in which the production line was developed concurrently with the product. This is a challenge when it comes to changing requirements in both production systems and products; it demands an iterative and dynamic requirement engineering process. Face-to-face communication and regular meetings are of immense importance in achieving cooperation and in clarifying requirement specifications and other key parameters.

In the field of robust design, the aim is to design products so as to make them insensitive to variation (Arvidsson & Gremyr, 2008). Agile and flexible design is used by Thomke and Reinertsen (1998) as a means of handling changing requirements and variations. The authors propose keeping requirements simultaneously frozen and liquid, making a comparison with the way a newspaper is structured so as to allow different time horizons for completion. Some parts are planned and written weeks in advance, while other parts are not finalized until the last minutes before printing. This implies that the requirements are planned to be frozen in succession, rather than all at once. This way, designers do not have to predict an uncertain future. However, trying to gauge the future in order to predict which requirements will be changed is actually proposed by Land (1982).
2.3.1.1 Agility as a means to manage the dynamics of requirements

Agile methods have become a great success in the domain of software development. Methods such as extreme programming and scrum are part of the concept of agile development. Ovesen and Dowlen (2012) investigate whether these methods are applicable to physical PD. The authors conclude that the methods applicable to software engineering are not completely transferable to physical PD. To gain the best of agile methods, other methods such as QFD and evaluation and selection matrices should be added. However, agile methods do show great promise when it comes to project development time and the development of customer requirements. Agility can also be seen from the flexibility point of view. Development flexibility can be regarded as the economic cost of modifying a product in response to external changes (e.g., changes in customer needs) or internal (discovering a better technical solution), according to Thomke and Reinertsen (1998). Flexibility is often used in a qualitative manner, but much can be gained by using quantifiable terms. The authors propose a “flexibility index” to measure how well a design actually responds to change.

Landaeta et al. (2011) describe scrum in development from an organizational learning perspective. Scrum is based on the principle that requirements are bound to change, and that it is a waste of resources to try to predict the future and set up too detailed a project plan. Scrum seeks to manage chaos by use of short-timeframe plans in which cost and time are fixed by the customer. The constraints are then negotiated within the project team to find a feasible scope. Several short iterations help to manage uncertainty; if something fails, the impact is not as severe as in traditional project management due to the limited scope. The team produces a working product increment by the end of each iteration, providing the customer with transparency and allow the product to be adapted through close collaboration between customer and development team. Scrum is characterized by self-managed teams, cross functional teams, and daily meetings. Scrum enables greater organizational learning due to the short timelines, unlike the classical lessons learned written at the end of a project. However, the philosophy lacks in terms of learning transfer between projects.

The concept of a skunkworks consists of a small hand-picked team that is formed and removed from the ongoing part of the business (Bommer, DeLaPorte, & Higgins, 2002). The team is given complete responsibility and operates in an autonomous fashion to develop new products or services. The team controls its own milestones and is entirely self-managed. The choice of procedures and responsibility for budget and resource allocation are also under the team’s control. Some reported success factors are:

- Adhering to a clear focus on the team’s mission.
- Extensive up-front planning efforts.
- Critically analyzing customer needs.
- Leveraging project overlaps.
- Early involvement of customer.
- Empowering the team.
- Breaking rules.
2.4 Reuse of design knowledge

The reuse of design knowledge has been studied by many researchers over the years. Tools and methods have been developed to support reuse of some aspects of both the design process and the artifact. One research field in which the formalization of knowledge is an enabling factor is the introduction of configuration systems. Haug, Hvam, and Mortensen (2012) emphasize knowledge acquisition, in which the knowledge of the domain experts is gathered and formalized. They also stress knowledge representation, which is described as a simulation of the relevant knowledge. These aspects are discussed together with seven strategies for implementing configuration systems. Stokes (2001) presents a complete framework and a detailed methodology known as MOKA that aims to collect and formalize knowledge in order to create knowledge-based systems. Knowledge is gathered from domain experts using standard templates, called ICARE forms, which together create an informal and formal model. Preston, Chapman, Pinfold, and Smith (2005) elaborate on the MOKA methodology and present a classification of KBE applications, giving examples of the different types: generative systems, advisory systems, and selection systems. These authors criticize MOKA in a constructive manner, proposing that a software tool would be helpful in the creation of the informal model for reasons such as checking for validity. They propose that MOKA should support an iterative process in developing the formal model, while a more concurrent approach is proposed for creating the code in the final system. This suggestion is an effort to mitigate the risk of programming and system integration problems that often occur if the coding is left for the very last stages. Creating the system at the same time as collecting the knowledge is proposed as a solution that creates a feedback loop that enables the experts to check whether the software model fits well with their own understanding of the area.

An important aspect of design reuse research is design rationale. Regli, Hu, Atwood, and Sun (2000) define design rationale as the explanation of why an artifact or some part of an artifact is designed in the way that it is. A complete design rationale includes all background knowledge, reasoning, tradeoffs made, and decisions taken throughout the design process. According to Elgh and Poorkiany (2012), access to the design rationale can support the development of new products, the modification of existing ones, or the reuse of finished products in a new context. The requirements concerning the scope and granularity of the design rationale to be captured depend on future needs. For example, in order to practice the reuse of rules in a new context, information such as scope, range, simplifications, and underlying assumptions are required. If a rule must be modified and adapted to specific circumstances, more information is required. Sun and Liu (2008) explore a method of delivering sufficient knowledge in order to enable design process reuse by attaching cognitive knowledge such as design intents and justification to the generic knowledge template. Alizon, Shooter, and Simpson (2005) present a methodology aimed at retrieving knowledge from existing product designs. This is carried out by filtering candidates based on their similarity to desired characteristics and their performance efficiency.

2.4.1 Tools for design knowledge reuse

There is a pressing need in industry for tools to practice reuse of knowledge. This is, however, a highly complex area, since knowledge can be of different kinds, in different formats, or in different locations (such as in PDM systems or in people). This complexity has hindered research to find a generic solution for knowledge reuse that integrates systems used by a company and is user-friendly enough to be employed throughout an entire organization. Huang, Jiang, Liu, Song, and Han (2015) describe the methodology for developing a knowledge map, which is a toolkit for visualizing and exploring contexts.
and relationships in distributed knowledge collections. Baxter et al. 2007 considers knowledge to be actionable information and problematize that many previous design knowledge reuse systems focus exclusively on geometrical data, which is often not applicable in the early stages. Future reuse models need to contain problem-solving methods, solution generation strategies, design intent, and project knowledge. Baxter et al. (2007) also stress that even if knowledge stored in computer-based systems is accessed, several additional factors must be met if it is to be reused: reusability, availability, and relevance. The authors propose a design knowledge reuse system that has two key elements, a process model and a product model. The process model provides a detailed structure, while the product model is a combination of product data and ontology. It is emphasized that a major contribution is derived from creating the prototype system, which forces the organization to formalize its knowledge. Christ, Wenzel, Faath, and Anderl (2013) emphasize a lack of user-friendly classification and structuring of engineering knowledge in today’s CAD software. Therefore, the retrieval of existing templates (used for knowledge capture) is a challenge and often results in the total recreation of designs. Their paper proposes a structured way of reusing feature templates to make up for the lack of doing so in a systematic way. The proposed approach is based on the generic product structure of a product.

2.5 Technology development

The way that research (TD) and development (PD) have been managed has evolved in recent years due to changes in the structure and demands of the economy. 60 years ago, R&D was seen as an overhead cost that was focused on pushing technology towards the market. As competition increased in the mid-1960s, a more short-term approach was adopted and market pull was emphasized to the neglect of long-term research. 10 years later, risk-reward analysis and cost reduction became important, leading to the elimination of waste. Efforts were focused on improving TD within a company. In the 1980s, company role models, such as Toyota and Sony, emerged. The focus shifted towards developing a complete product concept consisting of service, distribution, and product platforms. Additionally, integrated and parallel activities became goals to which companies began to strive. Most recently, it has become more and more common to share intensive technology investments by interacting with suppliers, distributors, and customers (Nobelius, 2002).

So, what is TD and what is it trying to enable? TD aims to develop knowledge, skills, and artifacts in order to enable PD (Högman, 2011). Deliverables can also appear in the form of demonstrated feasibility (Nobelius, 2002) or a technological platform (Cooper, 2006). It is further noted by Cooper (2006) that TD is important for a company’s long-term growth, but often is assigned low priority and represents a small portion of the total effort of a company. A technology has been defined as “a set of knowledge that forms a capability to achieve a practical result when applied to the design or development of a product, service or its manufacture or delivery” (Corin Stig, 2015, p. 7). Similarly, technology has been defined as knowledge applied to products or production processes (Säfsten et al., 2014).

Companies can gain a competitive edge by continuously and systematically investing in TD in strategic areas. TD can be described as a stream in which a company develops technologies and products that are in line with the company’s overall strategy (Clausing, 1994). In Figure 7, this is shown by visualizing a PD project (with a certain time frame and a targeted customer) that tries to fish out a - to some extent - pre-developed technology present in the company that can form the basis of a new product. However, what is the need to separate these two development processes? Since it is difficult to estimate the outcome of TD due to its fundamentally uncertain nature, a different management strategy
is needed. TD also differs from PD in its prerequisites, technical maturity, time horizon, need for competence, process repeatability, and completion point and deliverables (Nobelius, 2002). Deliverables from TD can appear in the form of demonstrated feasibility, knowledge, new technology, a technical capability, or technological platform (Cooper, 2006). It follows that separating TD from PD can reduce risk in PD projects (Lakemond et al., 2007). Lakemond et al. (2007) also stress that only verified technologies should be used in PD in order to minimize risk. Even though TD strategies and generic models are developed, TD projects are becoming rarer among companies due to their complexity and uncertainty, a lack of management knowledge, and the demands of smaller and shorter-term projects (Cooper, 2006).

Even though a separate process for TD has been proposed by several authors, difficulties still exist. Högman (2011) reports on TD in a case company and notes the following difficulties:

- Mismatch between engineers’ needs for predictive long-term goal formulation and the capability of producing such long-term anticipation.
- Difficulties in allocating sufficient priority to TD.
- Difficulties in selecting technologies for continued development.
- Difficulties concerning technology implementation.
- Reliance on a few strong individuals’ advocating for the incorporation and development of new technologies.
- Insufficient understanding of TD as primarily a process of understanding.

2.5.1 The technology and product development interface

The splitting of TD and PD creates interfaces between the two. A categorization of these interfaces is offered in Lakemond, Magnusson, Johansson, and Säfsten (2013): (1) contextual interfaces are interfaces at the edges of a new product development (NPD) projects, between the technology and the market; (2) technical system interfaces are those between the product and the production system; and (3) organizational interfaces occur when the project team interacts with the broader organization. The authors propose an interface assessment tool for use in the early stages of NPD to identify challenges, such as market uncertainty, technology uncertainty, complexity, and the degree of change in the product, the complexity and degree of change in the production process, organizational separation of TD and PD, and organizational separation of PD and production.

Nobelius (2004) offers a three-dimensional view of the interface between TD and PD:

- Strategic and operational synchronization.
  - Strategic synchronization concerns matching the strategies of technology and PD.
Operational synchronization concerns the point in time of introducing new technology into PD.

- Transfer scope. This refers to the decisions of what to transfer; it deals with concept, test results, and recommendations.
- Transfer management. Deals with how the transfer is carried out.

Lakemond et al. (2007) argue that strategic and operational synchronization is of greatest importance. However, they stress that transfer management must take place in a physical handover, and that an understanding of one another’s work must be developed. This is also stressed by Cohen, Keller, and Streeter (1979), who also indicate that the TD team must make a decision after transfer as to whether to maintain activity, support PD, defend its concept, or explore advanced or related concepts. Research personnel are needed in this transfer to offer support with tacit knowledge (Corin Stig, Högman, & Bergsjö, 2011). To increase the readiness of the applied research work in the eyes of the product developers, the number of information channels used for spreading the results can be increased, and there can be a greater focus on the development of concepts (Nobelius, 2004).

Johansson, André, and Elgh (2015) offer a practical example of how automated simulation models can be a part of transferring knowledge between TD and PD. Ravn, Gudlaugsson, and Mortensen (2015) describe a high-level architecture to describe technology prototypes. Levandowski, Corin Stig, Raudberget, and Johannesson (2015) propose an information model and process for technologies to support the knowledge transfer step. The model can be seen as a description connecting a core technology with, for example, those people who possess tacit knowledge about the technology, a prototype, tradeoff curves, technology readiness levels (TRLs), and technology implementation. The TRL scale is used to judge the level of maturity of a technology (Mankins, 1995).

2.6 Summary and research opportunities

This literature overview has shown that there is a need for increased efficiency in companies; customization has been pointed out as an important area to fulfil customer needs more completely. A platform approach has been shown to be an enabler for efficient customization, reuse, and production standardization. There is, however, a lack of research detailing how a platform definition can be utilized by suppliers working with an ETO customization strategy to gain the benefits of platform thinking. This stresses the need for methods based on industrial needs, challenges, and possibilities. An important step in knowledge reuse is the knowledge acquisition process, which consists of collection and formalization. However, there is little research covering how knowledge of different kinds and in different formats can be integrated into a platform description to achieve advantages similar to those found in component-based platforms.
FRAME OF REFERENCE
METHODOLOGY

The research methodology aims to describe how a study has been conducted. The methodology used and its reliability and validity together determine the quality of the research result. This section first aims to describe the basics of the methodologies used for this work before describing how these methods have been applied in this specific research.

3.1 Design research

Design has been defined in many ways, many of which depend on the culture and background of the author (Blessing, 2003). Many, however, agree that design brings together artifacts, people, tools, processes, and organizations, making it a highly complex and even somewhat chaotic domain, according to Horvath (2004). This may be one of the reasons why the validity of design as a research field has been questioned in the past. Design research is still quite a young field which is yet to be explored in full. It is therefore important to build methodological rigor for design researchers. Several design research methodologies have been developed and applied in the past, with a frequent common denominator being the inclusion of both a descriptive and a prescriptive element (Blessing & Chakrabarti, 2009; Duffy & Andreasen, 1995; Hubka & Eder, 2012). This is in contrast to many other research branches that focus mainly on understanding a phenomenon without affecting it.

3.2 Design support models

When it comes to developing computer-based models, Duffy and Andreasen (1995) propose the approach shown in Figure 8. The model is based on the hypothesis that any developed tool will have an impact on the design process when employed. The intention is that the models that are built are rooted in the reality of design and evolve to develop tools to support design. The approach consists of three types of models: phenomenon models, information models, and computational models. Phenomenon
models are based on observations and analyses of the reality of design and therefore reflect descriptive models. These models are then refined into information models that act as blueprints describing the desired state. The information models are then used to develop computer models and tools to support the design process. The prescriptive models are used to modify, test, and optimize the design process. By studying the effect of the prescriptive models on the process, insight can be gained into the process, which can in turn be used to improve the prescriptive models.

![Diagram](image_url)

Figure 8. Design modeling research approach, according to Duffy and Andreasen (1995)

3.3 Research design and analysis of data

The overall research approach used for this work was proposed by Blessing and Chakrabarti (2009) and is called a design research methodology (DRM). The outline of the approach is shown in Figure 9. The framework is partly based on the fact design science is not only striving to create knowledge about a phenomenon, but also trying to improve the design process itself. Assumptions based on both understanding and beliefs are made regarding how to accomplish this improvement. A clarification of the main phases follows:

- **Research Clarification (RC)** refers to the activity in which researchers try to find some evidence to support their hypothesis. A general understanding of the research field is sought, mainly by studying and analyzing literature. The initial reference model and its associated impact model are two important outcomes of this stage. These models link the key factor, which is what the intended support should affect, and the success criterion through a complex network of assumptions and support from existing literature. The reference and impact model developed for this work is shown in Figure 1.

- **Descriptive Study I (DS-I)** describes the stage at which the researchers have a clear focus and described goals. The literature is studied further but insufficient evidence is found. Understanding is increased by observing and interviewing designers (i.e. assessing the state of practice). Success criteria, measurable if possible, are created as a datum to evaluate the solution
later. The reference model is completed, clearly stating the path of argumentation from influencing factor to affected goal.

- **Prescriptive Study (PS)** refers to the stage in which a support to aid in the design process is introduced into the studied situation. This is executed by finalizing the impact model and describing where the support is to be introduced in order to reach the desired state. In this stage, support is verified by investigating how well the support functions by itself.

- **Descriptive Study II (DS-II)** describes the phase in which the support is validated, i.e. tested in the intended environment. This stage is also where the question of the support contributes to the planned success is assessed through comparison with the initial success criteria.

![Diagram of generic design research methodology framework](image)

**Figure 9** The generic design research methodology framework according to Blessing and Chakrabarti (2009)

3.3.1 **Type of research**

DRM proposes seven different ways of applying the methodology. Depending on the characteristics of the specific research that is conducted, each stage can be applied on different levels of depth. Review-based refers to a result produced by analyzing existing literature. Comprehensive, on the other hand, refers to a result and method which requires that new knowledge is created by empirical methods. These can be the result of an interview study or the development and introduction of a support. Initial refers to a step that has not been fully completed. The focus instead is on preparing a result to be used by other researchers. An explanation of each step follows and is shown in Figure 10.

1. **Type 1. Comprehensive study into criteria**
   When the criteria used to determine the success of a support are little understood, a comprehensive DS-I needs to be undertaken. The expected result from this type of research is a better understanding of the situation and which metrics are applicable.

2. **Type 2. Comprehensive study of the existing situation**
   This research type is similar to Type 1 in that an insufficient body of literature exists to establish
the necessary metrics. This case, however, allows for an initial PS to be undertaken in order to demonstrate how the understanding gained can be used to improve design.

3. **Type 3. Development of support**
   When enough literature exists to support both RC and DS-I, a comprehensive PS can be undertaken through the development of support. The developed support will go through an initial evaluation in DS-II.

4. **Type 4. Comprehensive evaluation**
   When literature can be found to support the first three stages of DRM but is lacking in the fourth step, this research type is applicable. The result of this stage will likely be suggestions for improvements or further development regarding the support.

5. **Type 5. Development of support based on a comprehensive study of the existing situation**
   This type can be seen as a combination of Types 2 and 3. The aim is to develop support, but in order to do so an understanding achieved through a comprehensive DS-I is needed. This is followed by an initial evaluation in DS-II.

6. **Type 6. Development of support and comprehensive evaluation**
   This type combines Types 3 and 4. The understanding of the situation is obtained from the literature which is sufficient to develop the support. The project recourses are of a kind that allows for an evaluation of the support and a revisit of either DS-I or DS as an initial or comprehensive study.

7. **Type 7. Complete project**
   In this type of research, comprehensive studies are conducted in each DRM stage. This type also includes iteration of the support development. This type is described as requiring substantial resources and is commonly executed by a research group unless an area with a very narrow scope is addressed.

<table>
<thead>
<tr>
<th>Research Clarification</th>
<th>Descriptive Study I</th>
<th>Prescriptive Study</th>
<th>Descriptive Study II</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Review-based</td>
<td>Comprehensive</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2. Review-based</td>
<td>Comprehensive</td>
<td>Initial</td>
<td></td>
</tr>
<tr>
<td>3. Review-based</td>
<td>Review-based</td>
<td>Comprehensive</td>
<td>Initial</td>
</tr>
<tr>
<td>4. Review-based</td>
<td>Review-based</td>
<td>Review-based</td>
<td>Comprehensive</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Initial/Comprehensive</td>
<td></td>
</tr>
<tr>
<td>5. Review-based</td>
<td>Comprehensive</td>
<td>Comprehensive</td>
<td>Initial</td>
</tr>
<tr>
<td>6. Review-based</td>
<td>Review-based</td>
<td>Comprehensive</td>
<td>Comprehensive</td>
</tr>
<tr>
<td>7. Review-based</td>
<td>Comprehensive</td>
<td>Comprehensive</td>
<td>Comprehensive</td>
</tr>
</tbody>
</table>

Figure 10. Research types adapted from Blessing and Chakrabarti (2009)

3.3.2 Data collection methods
The nature of this work relies on qualitative data collection methods. Interviews served as a way to get first-hand qualitative responses from design engineers possessing vital information. Interviews can be classified as follows (Williamson, 2002):
METHODOLOGY

- Structured: This technique is similar to a survey but is administered by an interviewer. The questions and their sequence are fixed, which does not allow for improvisation or open questions.
- Semi-structured: These interviews usually have a standard list of questions but lets the interviewer ask follow-up questions. This method is a mix between structured and unstructured interview approaches. The purpose is to capture the respondent’s perspective on the specific situation under study.
- Unstructured: This is a method used to explore a subject with the person being interviewed. It can be used as a preliminary step before creating a structured interview or a self-administered questionnaire.

Closely related to interviews are focus groups and workshops. These can be more or less structured, focusing on one or more predetermined issues, questions, or subjects. At first, questions are posed in an open manner after an initial discussion is held in a structured fashion.

An alternative data collection method is the self-administered questionnaire. This method offers good potential for gathering information from large groups in order to build quantitative data. It also has the ability to contain open-ended questions for qualitative purposes.

It is important in studying the knowledge-intensive PD process to examine documentation and products. Blessing and Chakrabarti (2009) report that the use of products, drawings, notes, and meeting minutes are all vital to understanding design and its processes. Documents can be either internal – describing processes, geometry, and calculations – or external, focusing on legal issues, etc.

3.3.3 Action research

Action research was developed as a way to conduct research in the field of education; it is suitable for situations when the research path is not evident from the start. A key characteristic that differentiates it from other research designs is the fact that the researcher interacts with the studied situation. Design research is similar to action research in that an aim of the research activity is to bring about change while also creating knowledge (Williamson, 2002). This approach is usually carried out in cycles iterating between action and reflection, as shown in Figure 11. Each action generates a result that leads to reflection. The next action stage is based on a plan that is the result of the reflection; the pattern continues for as many iterations as are required. The details of the action research process can seldom be planned systematically or too far into the future, since the solutions and insights are gained in an explorative manner. Williamson (2002) provides several examples of techniques that can be utilized in this process, such as convergent interviewing and focus groups.

![Figure 11 The most basic action research cycle adapted from (Williamson, 2002)](image)
3.3.4 System development

A quite new approach within applied research is systems development. This approach is based on the belief that development is always associated with exploration, advanced application, and operationalization of theory. This research approach arose in the area of information science as a way to manage the multi-disciplinary characteristics of and bridge the gap between the technological and social sides of this field (Williamson, 2002). It has also been called an example of action research, in which the researcher is part of and thus involved in the construction and testing of a method or information system in a real-world context (Burstein & Gregor, 1999). Systems development has become an important means for the development of support and realization and prototyping of models in engineering design. It is also argued by Nunamaker, Chen, and Purdin (1990) to be a central part of a multi-methodological information systems research cycle, as seen in Figure 12. Systems development becomes an intermediate step linking basic and applied science. One must emphasize that, although a prototype can be used as a proof-of-concept, it should not be viewed as a research contribution in and of itself (Nunamaker et al., 1990)

![Diagram of system development process](image)

Figure 12 A multi-methodological approach to information system research (adapted from Nunamaker, Chen and Purdin (1990, p. 94.))

3.4 Quality of the research

Evaluation is crucial to guarantee the level of quality in any research. Evaluation deals with comparing a number of criteria to data of some kind, such as requirements (Duffy & O’Donnell, 1998) or success criteria (Blessing & Chakrabarti, 2009). Evaluation can be divided in two components, verification and validation. However, the definition of these two components can vary with the author and research field. This study uses the definition presented by the discipline of Systems Engineering (Kossiakoff, Sweet, Seymour, & Biemer, 2011, p. 393): verification is the process of determining whether a system (in this case research) implements the functionality and features correctly and accurately. In the case of a software tool, verification involves determining whether the tool functions in and of itself. Validation, on the other hand, is the process of determining whether the system satisfies the users’ or customers’ needs. In the case of a software tool, verification is about testing the tool itself in search of errors in how it functions. Validation implies judging success in the context for which the tool was intended. According to Williamson (2002), validity refers to the extent to which a research instrument
measures what it is designed to measure. The characteristics of this research are mainly qualitative; Olesen (1992) provides five criteria that can be used in order to assure validity:

- **Internal logic** – known and accepted theories are the basis of the research, and the work is stringent from problem to the result.
- **Truth** – the theoretical and practical result can be used to explain real phenomena.
- **Acceptance** – the research is accepted by the research community. The tools introduced are accepted by practitioners.
- **Applicability** – the use of the introduced tools leads to enhancements over the situation if they had not been used.
- **Novelty value** – new solutions are presented or new ways of looking at a problem are introduced.

### 3.5 Application of the research methodology

The research presented in this thesis has been executed using a combination of different research methods. However, DRM, as proposed by Blessing and Chakrabarti (2009), is the approach that frames the complete research work as a whole. The application of DRM includes a loopback of the last two stages, meaning that both PS and DS-II are completed twice. The following section focuses on how the research design was established for each paper that is included in this thesis. Figure 13 shows a summary of the connections between papers, research question, DRM stages, and the characteristics of the paper content. The papers on which this thesis rests show a progression and deepening in the subject rather than a broadening of the subject. Each paper has been peer-reviewed, both within the research group and as a part of the publication process of the relevant conference or journal.

- **Paper 1** is mainly concerned with the RC stage and the interface between the RC and DS-I stage. The foundation of the paper relies on a thorough systematic literature review that pointed to a research gap. The outcome of the systematic literature review underpins the VINNOVA-financed project ChaSE, in which this work has been carried out. To enter DS-I, a semi-structured interview study was planned in collaboration with the research group. The interview study was conducted at four companies, with two to four people at different levels in the organization representing each company.

- **Paper 2** focuses mainly on DS-I but also begins to enter the PS-stage. This paper used one company as a case example; the model presented was applied to this case. However, the developed model was created using the results from Paper 1, which was partly based on interviews at four companies. The data collection in this paper was based on unstructured in-depth interviews and document reviews. The synthesizing of the initial theory was made using a combination of action research and systems development. Systems development was used as a way of sketching the concepts that were emerging by using coding and building user interfaces. These concepts and the prototype system were presented to the company representatives and iterated based on their comments, like the action and reflection steps in action research.

- **Paper 3** is a further development of Paper 2. Both unstructured in-depth interviews and review of documents were used. The development of the model that was initiated in Paper 2 was conceptualized using UML. Both the information model and the computer model were iteratively refined using the mindset developed by Duffy and Andreasen (1995) and prototyped using a systems development working approach. The prototype was verified by modelling an array of systems and testing their functionality. A first validation of the developed support was
METHODOLOGY

made using a self-administered qualitative questionnaire given to three company representatives who had attended a presentation and received a tutorial regarding the computer support. Therefore, this paper lies in the first loop of PS and DS-II stage.

- **Paper 4** includes results from semi-structured interviews and workshops. The first workshop defined the success criteria to be used for evaluation purposes. Succeeding workshops focused on how the knowledge already residing in the company could be used to deal with the questions posed by the overarching project description. The final evaluation used a self-administered questionnaire and enabled the respondents to grade the level of success criteria fulfillment.

The development of the overarching model was conducted as a joint effort in the research group, using several documented workshops. This paper describes the second loop of PS and DS-II, including a refined support tool and the second evaluation.

With the result and methodological work that has been conducted in these four papers in mind, the most appropriate DRM research type is **Type 7**, due to the comprehensive work that has gone into each DRM step and the loopback that led to completing PS and DS-II twice.

<table>
<thead>
<tr>
<th>Paper</th>
<th>RQ</th>
<th>DRM step</th>
<th>Empirical study</th>
<th>Theory</th>
<th>Tools</th>
<th>Application of tools</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>I</td>
<td>RC</td>
<td>DS-I</td>
<td>PS</td>
<td>DS-II</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>II</td>
<td>RC</td>
<td>DS-I</td>
<td>PS</td>
<td>DS II</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>II, III</td>
<td>RC</td>
<td>DS-I</td>
<td>PS</td>
<td>DS II</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>I, II, III</td>
<td>RC</td>
<td>PS (1)</td>
<td>DS-II</td>
<td>(2)</td>
<td></td>
</tr>
</tbody>
</table>

Figure 13. Summarizing the connection between papers, research questions, DRM stages, and paper content characteristics; inspired by Levandowski (2014, p. 39)
SUMMARY OF PAPERS

This chapter presents the main results from the appended papers, which are used to underpin the answers to the research questions. The main focus of the research was to develop a platform approach for ETO suppliers and thus increase their responsiveness to fluctuating requirements that occur throughout the PD process. This approach has been developed in steps, starting with investigating the needs, possibilities, and environment of such a platform, passing through the development of a conceptual model, and concluding with a formalized model and approach. The name Design Platform (DP) has been used in Papers 2-4 and is also used in a broader context in Elgh, André, Johansson, and Stolt (2016). This thesis focuses first and foremost on an application of the concept presented in Elgh et al. (2016) and therefore uses the term “platform approach” to refer to the main result of this thesis. The realization of the approach in a computer support application has been developed and presented concurrently in Papers 2-4. The order of the papers thus corresponds to different levels of maturity, formalization, and evaluation of this approach and tool. It is therefore important to note that the platform approach has evolved throughout the papers and that each paper section in this chapter refers to the model and approach presented in the corresponding paper, rather than to the final result presented in Paper 4. Figure 14 visualizes the connection of results, the evolution between the papers, and the level of model formalization and refinement of the results obtained.

4.1 Paper 1

Companies can gain a competitive edge by continuously and systematically investing in TD in strategic areas. Paper 1 argues that this is a challenge for suppliers of customized systems due to the large difference between the various systems into which their products are to be integrated, the markets for which their products are intended, the use of a given product, and each customer’s individual
preference. In this paper, four different companies ranging from OEM to B2B suppliers were interviewed regarding their qualitative views on how they engage in TD and PD, how they create, describe, and maintain product platforms, and how they respond to and manage changing requirements during the development of products. The objective of the paper is to establish the companies’ views, challenges, possibilities, and ways of working regarding the presented areas and to assess whether product platforms are used to meet the demands of efficient product customization.

4.1.1 Platform challenges and prerequisites in ETO industries

The focus in the interviews presented in Paper 1 were the company platforms or elements that could be described in a platform context. One company used the term “platform” internally. For the others, the platform constructs and their relations were identified by the research team.

The interviews showed that the definition of technology differed between companies. This was evident in a variety of ways, such as the planned finishing time of TD projects. For the companies developing subsystems to be integrated in to an OEM product, the division of TD and PD was more important, due to risk management concerns. One company developed products that lacked interfacing systems, which made the division of TD and PD less important. A similarity among the companies was that there was always a product in mind when starting a TD project. TD also had different deliverables in the companies. Two of the companies aimed to realize new technology in a physical prototype and documents, while another company aimed to describe new methods and instructions.

All companies in the study were faced with the challenge that requirements fluctuate. However, the view of requirements differed between the companies; for example, one used the term “requirement freeze” and strived to establish all requirements early in the process. Another company had a more dynamic view of requirements and saw a large design space as an advantage. One company had a strategy for using robust design to withstand changes in requirements.

The interviews made clear that the companies aimed to work in a platform-based fashion. However, the company platforms were for the most part identified by the research team rather than being used or described within the companies.
4.1.2 Platform abstraction levels

The companies all had well-defined and similar processes for PD. A common aim in all companies was to use a platform approach to some extent. The platform constructs could be defined on different levels of abstraction. In Figure 15, the companies are roughly situated according to the abstraction level of their platforms. The types of variant specifications, following Hansen (2003), are also coupled to the abstraction level of the platform. It should be noted that each company had characteristics that could be coupled to several of the variant specifications types; in the figure, however, they are placed on the level that fit them most closely. Company factors that increased the need for a higher platform abstraction level were small production volumes, high product customization, and high product complexity. It thus follows that the higher the abstraction level of the platform, the more engineering would have to be carried out to deliver a product. Moreover, the higher the abstraction level of the platform, the more the companies tended to describe it as an explanatory model rather than one composed of physical components.

![Figure 15. Identified platforms and platform content categorized in different levels of abstraction](image)

4.1.3 A platform model hypothesis for suppliers of customized systems

Paper 1 ends by showing a mapping of different development assets as a platform model hypothesis for ETO companies, as shown in Figure 16. This model hypothesis is first mentioned in Elgh (2013); it differs from the traditional artifact-based platform models in that it contains an array of models, methods, and tools that might be possible to integrate in a platform model. The figure describes how TD is separated from PD and how the deliverables from TD and PD build up a platform in which the technology is effectively described and can be adapted to fit the different PD projects. The PD personnel can then use the platform for creating customized variants. The platform also contains the maintained knowledge that is continuously developed in the company projects. The efficiency of the proposed platform is coupled to its ability to adapt, how well it can handle fluctuating requirements, and how effectively variants can be created from it.

4.2 Paper 2

Paper 2 elaborates on how a company can describe the outcome of TD and PD on a conceptual level by introducing descriptions at different levels of concretisation. This is discussed in connection with a platform approach as a means to increase the reuse of design knowledge. The conceptual platform approach introduced was inspired by the model hypothesis presented in Paper 1 and by the state of the investigated companies.
4.2.1 A conceptual platform description

The aim of the conceptual platform approach presented in Paper 2 is to enhance the reuse of the technologies and designs developed in a company. Reuse goes hand in hand with platform thinking as a way to keep the design effort efficient and manageable. The product descriptions that are the output of the PD process (e.g. CAD drawings) are concretised to a high level, which also usually means a narrowing of the design space. Due to this narrowed design space, the product instance will have limited possibilities to adapt to a new situation when the prerequisites or requirements change; i.e. the possibilities to reuse the instance will be restricted. What is often lacking is a platform description to support the development that can gain the benefits of platform thinking.

If design knowledge is captured, structured, saved, and is retrievable, it can be reused in future development projects as part of the platform definition. By being proactive and exploring a design during TD and PD, this knowledge could be saved and reused by the addition of descriptions at different levels of concretisation. This study investigated and presented how this can be achieved by saving and structuring blocks of knowledge, referred to here as design elements (DEs). These descriptions, which partly constitute the platform, leads to something other than the component- and module-based product platform. The platform is now not composed only of the physical elements that compose the product but also by elements that support the designing of the product. Therefore, the name “design platform” is more suitable than “product platform,” since it refers the activity as well as the thing.

4.2.2 Introducing design elements as carriers of design knowledge

The proposed concept builds on an object-oriented view of a product description. The generic product description is thus a class; when instantiated, it becomes a product instance (object). There are different levels of classes in the concept; the top level is called the “design description class” and can resem-
ble a structure with subclasses, metadata model classes, and DE classes (Figure 17). The generic product structure is based on the of the PVM tool (Hvam et al., 2008), but is altered here to fit the purpose more successfully. A “part-of” structure describes the class hierarchy of subsystems and components, whereas a “kind-of” structure describes the types of instances. Every class in the “part-of” structure is described by a metadata model. Every object corresponding to a certain class is then coupled to this metadata model. Each type of DE corresponds to a class. The generic set of DEs is inspired by the ICARE forms (Stokes, 2001) and consists of:

- **Entity**, which is a description of a specific component or subsystem and includes, for example, function and behaviour.
- **Activity**, which is used to describe a task or process that often includes an execution order; attributes include inputs, outputs, triggers, and objectives.
- **Rule**, which describes a guideline or a set of valid relations for the designer to employ; rules can be described by mathematical formulas, tables, or in text form, and rules can describe design parameters and how they affect different variables.
- **Constraint**, which describes a limitation usually based on some boundary condition, such as manufacturing equipment or customer requirements.

The design description is a living document during the PD and TD phases and is continually filled in with knowledge and relational descriptions as the product is developed.

### 4.2.3 Design element structuring and identification

The dependency structure matrix (DSM) tool was used to identify connections between parameters, design variables, and items (Figure 18). Item refers either to a component or a subsystem. By combining the DSMs, domain mapping matrices (DMMs) were created not only to see the connections inside one matrix (e.g., parameters) but also between the matrices (parameters, items, and design variables). This was done in order to map the given parameters to items and thus cluster which parameters will be the input to designing an item or configuring a system. All three DSMs play a part in defining DEs. Since both parameters (input to the DEs) and design variables (output from the DEs) are coupled with a generic item class, they will partly define the DEs. Some parameters and design variables will be limited to a specific item, others will span several items, and some will only be on the architectural level. The mapping was used to
identify which parts of the system were customized and in what ways. A first prototype of a support system to manage DEs was introduced. The designer was able to model DEs on spreadsheet templates in MS Excel. Initially, the application read all created spreadsheets in a specific folder location. The information in the spreadsheets was used to create objects in the application, which were categorized and displayed to the user in order to obtain an overview of the DEs that were present.

4.3 Paper 3

Paper 3 continues to build on the concept presented in Paper 2 with further refinements and applications. A first evaluation is carried out to investigate the progression and set the direction of the continued research project. The platform approach is realized in a prototype software application called the Design Platform Manager (DPM). The platform approach and software tool aid the case company in describing not only finished designs and how they relate to a generic design, but also elements like methods, task descriptions, constraints, and design rules. The platform model is allowed to evolve over time so that its definition supports the studied company in being more adaptable to fluctuating requirements. The evaluation shows good result in terms of increasing the level of reuse, speed, and accuracy during quotation and in supporting the design engineer.

4.3.1 A platform approach to support suppliers of customized systems

The platform approach is defined by the following characteristics in paper 3:

- Descriptions of product instances and their interrelation with a generic description, which means that the platform evolves as the instances are created in succession.
- Descriptions of the building blocks of the designs and design processes and both generic and specific descriptions of those elements.

Both kinds of descriptions focus on the reuse of company assets. The starting point in defining the model is the identification and definition of generic product items (GPIs). These can be seen as the common foundation to which different constructs and descriptions can be linked. These items might not correspond to an already existing design; however, it is known that the item must be included in the finished design. The platform approach proposes that the item can be bi-directionally linked to other kinds of descriptions, in the sense that the item might require that a specific description is used in the construction of the item or vice versa; the starting point is a DE that in turn defines the item structure. Each structural level in a GPI is associated with valid descriptions that can be used in the construction of that specific item. This can then contain state-of-the-art descriptions, the latest versions that have been proven to work by experience or evaluation. The expansion of the model is executed by setting up or modifying the model by modeling GPIs and creating descriptions that are linked to relevant information. The use phase (which in the published version of Paper 3 is referred to as the “execute” phase) involves the information in the model being used by instantiations of the generic product items to become product variants.
4.3.2 Supporting the use and expansion of an applied design platform

In order to support the use and expansion of the platform model, the conceptual application first presented in Paper 2 is further developed. The application features a functionality that allows the user to create GPs and couple DEs to the different levels in the GPs and the instantiations of variants. Instantiation implies that a variant of a generic product item is created. The user interface of the application is shown in Figure 19. The application uses the following features:

- Spreadsheet templates for creating DEs.
- XML for saving GPs, instances, and the associations with DEs; these are saved locally on a computer.
- A user interface that has been coded in the scripting language Visual Basic which is used to:
  - Model GPI structures.
  - Read DE spreadsheets.
  - Associate GPs with DEs.
  - Instantiate GPs into variants.

The paper presents the application of DPM in the case company to support the quotation process. DEs are created and added to the DPM by mapping the quotation process and the product in focus. The evaluation of the DPM in the case company is presented in Chapter 5 of this thesis.

Figure 19. Screenshot of the application user interface with principal explanations
4.4 Paper 4

Paper 4 outlines a refined and detailed platform approach that is an application of the DP concept (Elgh et al., 2016) and represents an evolution from the insights revealed in the previous papers. The approach in Paper 4 is expanded to realize the platform definition offered by Robertson and Ulrich (1998) to a higher degree; they define a product platform as the assets residing in a company. The terms “assets” and “resources” are used interchangeably here. The evaluation performed in Paper 3 demonstrated that the platform approach and support tool were enablers for design engineer support in customizing products. Based on the constructive criticism the platform approach and the support tool are further developed in Paper 4. The evaluation presented in this paper is outlined in Chapter 5 of this thesis.

4.4.1 A refined and formalized platform approach and model

Companies offering a high level of customization often have an identified product concept consisting of items at different levels of realization. Knowledge and experience are also part of the product concept and support the designing of specific variants. The platform approach allows for modeling of these items on a generic level and their association with existing solutions or other resources that support their realization. A specific platform model is composed of different objects related to process, synthesis resources, product constructs, assessments resources, solutions, and projects. An applied formalized UML representation is shown in Figure 20. The uppermost concrete class is called GenericProductItem; it is a generic description of a product concept. Each object that is instantiated from this class becomes a variant and is unique to a specific project. The GenericProductItem has the property TopComponent, which can be of either the Assembly type or the Part type. The Assembly and Part classes are holders of the resources needed or intended to support the realization of the items and not the solutions themselves. The class Component has a list of Construct classes. Construct is the general type of class that is used to model and point to resources of different kinds.

The different classes that inherit Constructs are ProcessResource, SolutionResource, SynthesisResource, AssessmentResource, GeometryResource, and Project. The Constraint class also inherits Construct but can equally be part of SolutionResource, SynthesisResource, and AssessmentResource. Objects instantiated from SolutionResource relate to finished designs and have thus been created with some boundaries, as defined by Constraint objects. SynthesisResources have explicit constraints that are stated and relate to the method, guideline, or tool that an object represents. AssessmentResource has implicit constraints with implications that are made visible through elements such as behavior models. In the case of the application presented in this paper, DEs were used to embody and formalize design engineer knowledge and to enable the modeling of resources. These too inherit Constructs but also contain additional attributes and methods.

4.4.2 The DPM and PDM

The DPM functionality is extended from Paper 3 to allow for the creation of associations with content in the company PDM system. The DPM enables the user to model GPIs as trees in which the main
structure is composed of assemblies and parts as holders of resources. Each level in the structure can have associated resources. The different GPIs and other object properties can be viewed, added, and altered. The content of the PDM system can be searched through a PDM viewer in the DPM and associated with the modelled structure. DEs, GPIs, Assemblies, Parts, and variants are all saved in the joint database.

Figure 21 shows the principal architecture, main functions, and how DPM and PDM coexist. The PDM system focuses on file management, keeping track of versions, metadata, and process flows, while the DPM focuses on offering a coherent view of the resources that are part of the platform and used for the creation of variants. In order to build the PDM save functionality, the PDM database was used by the addition of relations (tables) that incorporate the information model shown in Figure 20. This was carried out to enable concurrent usage and to minimize data redundancy. The process included the addition of tables for the GPI and variants, Assemblies, Parts, and Constructs. To associate an object saved in the PDM related relations with objects in the PDM system, a string attribute link was used to point to the specific database record in the PDM database relations.
SUMMARY OF PAPERS

4.4.3 Platform expansion and use

The paper presents how the platform approach is used to map the case company product. The generic DP model shown in Figure 20 was used to map the existing state of the company by the identification and categorization of which resource types existed and what was to be supported in the future. The results from the interviews and workshops identified the current ways of working and the challenges and needs regarding product platforms. Those results made clear that there was a need for increased efficiency in the development process by increasing the reuse of knowledge and supporting the work of developing adaptable solutions.

The paper further describes the details of how the platform approach is applied and supported. Detailed examples of what is added to the platform are given by outlining the product concepts, guidelines, and more by using DEs as the format to increase the use of resource types. Figure 22 exemplifies how a development team uses the platform approach in the development of new product variants. During the quotation process, the support application (DPM) can be used to browse for solutions to and support for the design work. This creates an ability to reuse physical solutions and other knowledge that has been associated with the generic product items and variants. During the downstream development, changes in requirements are often what create a need to adapt solutions. On these occasions, the DPM can again be used to find resources to respond to the changes that arise. Variants can be created by instantiating the generic structure and the chosen resources. The variant structures are continually expanded with the knowledge that is gained throughout the PD process and is finally stored as a model containing solutions and rational. The figure also indicates that the outcome of development projects is described and fed back into the model for future reuse purposes.
Figure 22. Example of resources used by the development team to generate solutions and expand the platform
SUMMARY OF PAPERS
EVALUATION OF RESULTS

As part of the research methodology applied in this thesis and in order to strengthen the level of its generalizability, two iterations were made from the PS stage to the DS-II stage. These two iterations included two evaluations where the first evaluation can be considered a “status check”. This section focuses on the evaluation of the platform approach and associated support tool in the company at which the support tool was introduced.

5.1 The first case evaluation

Paper 3 presents an evaluation of the software application and associated method used as a project gate to investigate the company’s view of the results and to establish future directions. The DPM was presented to, demonstrated to, and discussed with potential users at the case company. The three respondents represent three potential users of the tool at different hierarchical levels in the organization. The questionnaire employed contained questions mainly based on the success criteria that were identified jointly with the company at the beginning of the project and how well the presented method and software support met those criteria. The questionnaires, which were filled in individually, are summarized in Table 1. The results of the evaluation, along with comments from the engineers participating in populating the DPM, formed the basis for further development and refinement.

5.1.1 Evaluation analysis

The outcome of the evaluation showed good results and fulfillment of success criteria in a qualitative manner. The DPM was judged by the participants to:

- Support the designer.
- Decrease the time needed to respond to a quotation request.
EVALUATION OF RESULTS

- Support in assuring that requirements are fulfilled.
- Increase the possibility of reusing company assets.

Areas of improvement were focused on the software application prototype in terms of visualization, the need for system maintenance, and the need for CAD integration.

Table 1 Summarization of evaluation results

<table>
<thead>
<tr>
<th>Engineering Manager</th>
<th>Design Engineer</th>
<th>Quality responsible/Project leader</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. What possible use do you see for a method and supporting software system like this at the company?</td>
<td>Help and formalise and store existing data and structure (systems) to make it easier for designers to find relevant information that can be used for new designs.</td>
<td>Good possibilities, but it can be difficult and take time to implement.</td>
</tr>
</tbody>
</table>

2. In what way and to which degree do you think that this way of working (using the method with the prototype software as a support) could:

a. Decrease the time to respond to a quotation request? | Shorter response time due to better overview, knowledge, and access to designs and information. | Yes. If we get an RFQ from a customer for a system, we can use the software to find relevant information on current products and see whether they will meet the customer requirements. |

b. Support in assuring that requirements are fulfilled? | Requirements are always present. It can possibly contribute to our response to the customer’s requirements as a check-point/control station/gate. | Yes. Review the customer requirements and use the software and compare these with existing designs/products. |

c. Support the designer? | Better overview. Accessibility of data and a basis. | Yes. Easier for new designers to build a structure for a system and find existing products and information. |

d. Increase the possibility of reusing company assets such as:

i. Components and subsystems? | Better overview and access to necessary data, easier to find what has been done previously, which saves resources and time and minimizes mistakes. | Can be used to review existing components and see whether the components can be reused. |

ii. Knowledge? | Some complement as to how we currently work, visualization of knowledge. | Yes. Allows easy access to stored knowledge. |

3. What do you think are the drawbacks of the proposed solution?

- Needs to be developed regarding visualization, otherwise good. | Relies on good input data/information when setting up the system. Good maintenance of knowledge and data. | All-new experience and input data needs to be entered and saved in a proper way. |

4. What needs to be improved to ensure success criteria fulfillment and usefulness?

- Visualisation is needed. | Relies on good documented information on existing designs and products. This is more of an issue in how we work rather than the concept of the method itself. | Access to indicated cost for the selected design. More photos showing the selected products and components. |

5. What areas do you think need further investigation?

- Base facts e.g., customer-owned tool, limitations in use of the ongoing components. | Integration with CAD; visualize CAD models. | Access to CAD models or “black box” that describes the design solution. |

5.2 The second case evaluation

The SCs and indicators were used as a basis for creating a questionnaire that was given to each company representative participating in the evaluation. For each SC, a statement was given regarding the fulfillment of that SC. The statement was preceded by a clarification of its meaning. Respondents were then able to judge and report to what degree they agreed with the statement on a scale from 1 to 5, with 1 being “Strongly disagree” and 5 being “Fully agree.” A score of 3 corresponds to neither agreeing or disagreeing with the given statement, indicating an unchanged state. The SCs, indicators, and statements can all be seen in Table 2. The ID is an identifier for each SC in which the letters correspond to a categorization: T stands for Transparency, L for Lead time, Q for Quality, and P for Productivity. The rank column corresponds to how the complete set of SC were ranked by the case company in terms of relative importance. In the evaluation, four people participated from the company: the designer, lead engineer, engineering manager, and project manager.
Table 2. Success criteria, indicators, associated statements, and ranking.

<table>
<thead>
<tr>
<th>ID</th>
<th>Success Criteria</th>
<th>Indicator</th>
<th>Statement</th>
<th>Rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>T1</td>
<td>Reuse knowledge</td>
<td>Time to access and understand relevant information</td>
<td>It will be easy to find relevant information using the method/system.</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>The information stored using the method/system can easily be understood.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>The results or outputs from the method/system can easily be understood.</td>
<td></td>
</tr>
<tr>
<td>L2</td>
<td>Time spent to respond to quotation</td>
<td>Time</td>
<td>We will be able to decrease the time currently used to respond to quotation.</td>
<td>2</td>
</tr>
<tr>
<td>L3</td>
<td>Time spent on project</td>
<td>Number of design hours per project</td>
<td>We will decrease the number of labor hours in our projects by implementing such a method/system.</td>
<td></td>
</tr>
<tr>
<td>L1</td>
<td>Short start-up time</td>
<td>Time spent to introduce new user</td>
<td>The proposed method/system contributes to decreasing learning time.</td>
<td>3</td>
</tr>
<tr>
<td>Q1</td>
<td>Assure that requirements are fulfilled</td>
<td>Number of changes after verifying tests</td>
<td>We will be able to reduce the number of changes that must be made after verifying tests (i.e. we will increase our ability to ensure that requirements are fulfilled in an earlier stage).</td>
<td></td>
</tr>
<tr>
<td>Q3</td>
<td>Number of loops</td>
<td>Number of formal design loops required to achieve series production</td>
<td>The method/system will decrease the number of unplanned changes in series production.</td>
<td></td>
</tr>
<tr>
<td>P3</td>
<td>Reuse components</td>
<td>Number of carryover parts</td>
<td>We will be able to increase the number of carryover components.</td>
<td>4</td>
</tr>
<tr>
<td>P2</td>
<td>Support the designer</td>
<td>Assessment by the designer</td>
<td>The method/system will support the designer to a higher degree than the existing situation.</td>
<td></td>
</tr>
<tr>
<td>Q2</td>
<td>Lower number of errors</td>
<td>Number of changes in series production</td>
<td>The method/system contributes to decreasing the number of unplanned changes.</td>
<td>5</td>
</tr>
<tr>
<td>P1</td>
<td>Resource utilization</td>
<td>Number of designs created/design hour</td>
<td>Having a method/system like the one presented will make utilizing our resources more effective.</td>
<td></td>
</tr>
</tbody>
</table>

The evaluation was conducted in three stages:

- The first stage included a demonstration of how the platform approach had been applied at the company. This included a presentation of the developed method, its realization, and its connection to the generic platform approach.
- The second stage aimed to clarify the first stage by letting the participants ask questions.
- Finally, each participant completed the questionnaire by assigning scores to each statement and leaving clarifying motivations, suggestions for improvement, and highlighting strengths.

5.2.1 Evaluation results

Figure 23 show a summary of the answers to the questionnaire. The mean and corresponding standard deviation for each SC have been calculated. The standard deviation represents the level of agreement between respondents. The bar patterns indicate the rank group of each particular SC.

The averages for each SC are in all cases above 3.8, which indicates an overall improvement. The standard deviation is below +/-1 in all cases. Q3 and L3 received the highest scores and indicate the company’s belief that the ability to reduce the number of formal design loops will be strengthened and that the project time will decrease. T1, L1, P3, and P1 followed in score magnitude and pointed to the possibility to increase the reuse of knowledge and components and increased resource utilization and decreased startup time. The comments that were given concerned:

- The visual appeal of the user interface was seen as an enabler for using the support but was in need of improvement. This clearly points to the importance of a having a well-designed user interface, but might also indicate that the respondents focused more on the software prototype than on the underlying approach.
- The need for data input in system was seen as time-consuming.
EVALUATION OF RESULTS

- The need for a structured method to use the tool was emphasized, as was the requirement that such a method must be applied by all users of the tool. This was seen as something that would be difficult to achieve.

Several statements in the questionnaire directly or indirectly concerned lead time during development. Some answers in this regard can be seen as inconsistent or even contradictory. A pattern can be found regarding the score magnitude and number of comments. In general, comments were added to statements with lower scores, of which the number was low in this study. By contrast, there was a lack of comments regarding the statements which received higher scores.

![Figure 23. The results of the evaluation](image)
DISCUSSION

This thesis has reported on research regarding the needs and prerequisites in industry concerning the use of platforms, along with the development, introduction, and evaluation of a platform approach and support tool to answer the identified need. The aim of the research is to make companies that act as suppliers and develop highly customized products more responsive to changing and conflicting requirements during both the scoping and subsequent development processes. This aim has a long-term character, and was not expected to be fully achieved within the scope of this thesis. It is, however, important to note that the result presented point in that direction. The goal of this thesis is to investigate and support the current state in industry regarding platforms as a means to manage changes in requirements. That this goal was achieved is supported by the evaluation results. The introduced platform approach evolved from a hypothesis in Paper 1 to a refined, formalized, and applied platform approach in Paper 4. This chapter discusses the final platform approach described in Paper 4.

The use of the platform approach is intended for but not limited to companies developing highly customized products in B2B environments. The need for such a platform is based on the efficiency that platforms have been shown to assist in gaining a competitive edge. Issues, however, exist in the inability to foresee future requirements and to preplan variants for a future market. Another common issue for these companies is the splitting of TD and PD as a strategy to reduce the risk in PD, which puts greater pressure on those responsible for developing just the right technologies for future products.

The platform approach not only concerns the customization phase but also involves enabling a coherent format and method to support describing adaptable solutions, even during the TD stage. The work that is conducted in TD projects is closely tied to what the company must be able to customize later in time. The platform approach should therefore be supported by a set-based working approach rather than focused on individual solutions. Describing sets and ranges rather than specific products is the outcome in focus. The platform approach thus supports development outcome like guidelines, activities, rules, parametric models, etc. all which enable spanning of a wide design space. These descriptions are prepared in a manner that supports easy access and adaptation in later stages. The platform
approach supports formalizing what best practices and thus allows for leaving out more uncertain parts of the process if necessary.

The definitions regarding product platforms have evolved from being artifact-based to definitions that consist of knowledge and people. If this latter definition is used, every company that possesses knowledge has a product platform. However, this definition is not especially helpful; it only puts a new name on something that already exists. This thesis argues that in order for something to be called a platform intended for PD it must meet the following criteria:

- A coherent description such as a model that expresses what is included in the platform and the applicability of the content.
- A level of generality that will be the common denominator among any variants that are derived from the platform. These can be either a generic structure or generic building blocks attached to the structure or to one another.
- To support the forming of variants.

6.1 An enabler for increased responsiveness to fluctuating requirements

The traditional way to use platforms is to preplan variants and optimize the platform design to achieve both high commonality and high distinctiveness. To use this approach for suppliers in the ETO business increases the risk of losing projects to another supplier that agrees to a higher level of customization because its predeveloped variants do not comply with a particular customer’s requirements. The platform approach, as presented in this thesis, is about acknowledging fluctuations, realizing that change is inevitable, and adopting a flexible platform model and working approach that will not facilitate synthesizing a variant but will permit companies to be better equipped when offering higher levels of customization.

The expansion of the platform approach is about being proactive during PD by finding best practices and being zealous when creating the descriptions that arise when the platform is used. It is at this stage that the platform evolves by the addition of new knowledge and design descriptions that will be resources in future projects. Introducing generic assets in formats such as parametric CAD models, task descriptions, and tradeoff curves enables a set-based approach that uses definitions of spaces rather than point-based solutions. The need for an ability to master and manage fluctuating requirements requires the use of a platform approach; “master” in this sense means being responsive to fluctuations, not in the sense that requirements can be avoided. The use of the platform approach aids development loopbacks when iterations need to be made and evolves as new knowledge is added to it.

Revisiting the evaluation results in Chapter 5 indicates that the increased responsiveness to fluctuating requirements is supported by several factors. The first case evaluation pointed at supporting the designer in the quotation process; an increased possibility of reusing company assets and assuring that requirements were fulfilled was also emphasized. These factors support managing fluctuating requirements and reduce the time spent on quotation, as emphasized in the evaluation. The second case evaluation rated the specific SCs of the research project. This evaluation emphasized increased reuse of knowledge and components, decreased time spent on quotation, and increased utilization of company assets. This supports the increased responsiveness to fluctuating requirements by having different company assets formalized and available for use when changes arise.
6.2 Discussing the research questions

The thesis as a whole aim to answer three research questions. Below is a discussion that connects the research questions to the results in order make their connection as clear as possible.

6.2.1 RQ1: What is the current state of the utilization of product platforms for suppliers developing highly customized products in B2B environments?

The results used to answer this question are mainly the outcomes of a systematic literature review of several fields and interviews at four companies.

The initial assumption regarding the question of applying a traditional product platform at the supplier level in ETO companies is supported by both the literature and the interviews. The interviews note that there is a need to engage in platform development while acknowledging that there is a real challenge to achieving this aim. This view is supported by Högman et al. (2009) and Johannesson (2014). The challenge lies in the inability to preplan variants for future customer orders because those specific requirements cannot be known. Requirements also tend to change throughout the development as a result of unexpected changes in customer requirements or interacting system interfaces. The interviews also investigated how platforms were currently used. The companies had not explicitly formed a model to describe their platforms, if indeed one existed at all. By investigating company characteristics in combination with the elements that were identified as part of each platform, some conclusions could be drawn. Company factors that increase the need for a higher platform abstraction level were small production volumes, high product customization, and high product complexity. The term “level of abstraction” refers to how close to a realized product a given model is; for example, a guideline is more abstract than a manufactured component. It follows that the higher the abstraction level of the platform, the more engineering needs to be done to deliver a product. Moreover, the higher the abstraction level of the platform, the more the companies tended to describe it as an explanatory model rather than something explicitly defined.

TD is a prerequisite that is closely linked to the development of platforms. The connection between the two has, however, not been studied to a great extent in the literature. The definition of TD differed among companies that participated in this thesis, meaning that the companies took different amounts of risk in their PD by integrating TD to different extents. The separation between TD and PD was stricter if many interacting system interfaces existed with the customer with whom the product where to be implemented. The type of deliverables also differed, ranging from physical prototypes to feasibility studies using tradeoff curves.
DISCUSSION

6.2.2 RQ2: How can a platform approach be conceptualized to support customization for a company active in this environment?

Both manufacturing equipment like machines and tools and standardized manufacturing processes are a necessity in today’s manufacturing companies. It is common practice to keep track of what machines there are, what their limits are, what activities they can perform, and their operating costs. Development resources can be thought of in a similar manner as important assets in which a company can make investments. However, the capabilities of the development team and the solutions and tools that exist are rarely described in the coherent and comprehensive way that is common with tangible assets. The platform model proposed in this work strives to move nearer to using development assets like engineering knowledge as part of the platform. The platform should not only describe tangible items but also the support of designing the items. It should host adaptable solutions that enable efficient customization at later stages. In order for companies to develop an ability to create such a platform, change needs to be acknowledged. Further, a set-based approach needs to be applied in the scoping, quotation, and order processes so as to define design spaces that allow for adaptation. Reuse must permeate the development work by generalizing solutions. The platform must be able to host the full range of descriptions created in a company so as to maximize flexibility. The aim of such a platform is thus more focused on achieving efficiency in development rather in production, which is achieved by, for example, component commonality.

Judging by the results from the interviews and workshops presented in Papers 1 and 4, different types of needs and resources can be identified in the companies. During a workshop, company representatives were asked about what would make them manage fluctuating requirements more effectively. Their answers included being adaptable to changes and accessing previously created knowledge in different ways and formats. There were several possible formats mentioned, such as guidelines, tradeoffs, design rationale, and more. In order to work in a platform-based fashion, these ways of working and the diversity of description formats need support, and with the current platform models available there was a gap in existing research. The methodologies and models for working with platforms long focused on physical, component-based product platforms. The results of this thesis indicate that the positive effects of using a platform can also be leveraged by other constructs and resources already present in the companies. There are also possibilities of enriching the set of already existing descriptions and allowing them be part of the platform. Using a more dynamic platform definition will also permit the platform and its capabilities evolve over time as new knowledge emerges.

6.2.3 RQ3: How can such a platform approach be formalized and applied in practice?

As stated above, the platform approach can be used in one of two main ways. It can be used as a mindset and approach to relate to the development resources present in a company. It can also be used to formalize the model explicitly and conduct the necessary work related to setting up the model. This research question relates foremost to the second way of using the model.

First, the complete platform definition must be brought together in one coherent view that allows the user to work at different levels of abstraction that include both structural and detailed levels. To make such an explicit model useful in an industrial setting, support is needed to obtain a coherent view of the platform. The view must encompass the GPIs, the related variants, and their constructs. Using the model also must include the possibility to instantiate the model to create variants. The support cannot
be a standalone application but must be used according to a strategy that integrates PDM. This is important for the following reasons:

- To use and access the data stored in the PDM system easily.
- To reduce the risk of data redundancy.
- To enable multi-user and concurrent utilization.

This thesis has presented the Design Platform Manager as a means to support the formalization of the model. The application supports all the above criteria by the creation and application of a generic information model (Figure 20). The range of models that can be used in the platform approach has also been extended by using DEs as a format to describe design knowledge and artifacts. The approach offers an integration with the PDM system by simplifying the introduction of the tool. The DPM has been applied in practice and evaluated in that context. An array of product systems was modeled using the tool. DEs were created and saved in the PDM system to be linked to the models created in the DPM. Additionally, other resources that were important for the realization of product variants were identified and linked to the DPM. The only location used for storage is the company PDM database. The evaluation of the tool shows good results in terms of functionality.

6.3 Verification and validation of research

Verification and validation of this work has been carried out in several ways and can thus be viewed from different perspectives. A generic issue regarding any research in design is the complex nature of design research itself, which brings together many disciplines and where the intended effects might need several years to take full effect. This reality makes measurements and cause-and-effect relationships hard to establish with reliability, especially within the timeframe of a licentiate thesis. Design research is usually concerned with developing methods and models to support the design process. In order to verify and validate these methods and models, tools are developed in order to implement and observe the methods and models in their intended setting. Achieving validity is of paramount importance, but it must be preceded by verification that ensures that the tools functions in and of themselves. In this research study, verification has been conducted by developing the DPM support tool and using it to model an array of different product systems formally designed by the case company. These models have been iteratively shown to the case company, which confirmed that they were correct. In order to strengthen the validity of any research, it can be discussed in the light of Olesen’s (1992) five criteria: Internal logic, truth, acceptance, applicability, and novelty value. Internal logic is achieved if the research involves a well-managed project consisting of several phases and gates to guarantee quality, which is true of this study. Furthermore, this research is founded upon and supported by systematic literature reviews in several fields. Due to the participation of companies throughout the process, truth has been approached, since the research has been forced to relate at every turn to the companies’ realities. The continuous feedback from the case company in the different phases of the project helped to guarantee that there is an industrial significance to the work. Acceptance has been achieved by conducting evaluations and including the companies in the development of the theory. Review processes for conferences and journals are the most common for the global research community to demonstrate its acceptance of this work. The applicability criterion has been fulfilled by the users of the support presented in this research having judged its usability to be good and relevant. The presented platform approach and support tool can be said to have novelty since no previous examples that have been identified sharing the same scope and aim.
This thesis concerns the use, possibilities, and possible solutions regarding the development and application of platforms at an ETO-oriented company. Systematic literature reviews in several fields have resulted in the literature review presented in this thesis. The identified research gap regarding whether ETO-oriented companies can adapt and apply a traditional product platform was highlighted, along with surrounding factors that affected the companies. The appended papers show a progression that starts in Paper 1, which investigates companies’ challenges, needs, and prerequisites and strengthens the importance of the identified research gap. Papers 2-4 present the development and application of a model to support these companies in an evolutionary manner. The developed platform model and support tool aim to suit these companies’ present and likely future realities by allowing different kinds of development assets to become part of the platform definition. The main conclusion that can be drawn from this thesis is that there is both an interest and a need in the investigated companies to engage in platform development and utilization in an ETO context. This thesis has shown that traditional product platform definitions are either not applicable or too abstract to be directly applied to the ETO environment. However, the platform approach presented in the thesis, which is based on the design platform concept (Elgh et al., 2016), has been shown to enable platform-based development while also making use of the diversity of development assets that already exist in a company. These innovations lie largely within the contributions that have been made to industry. The scientific contribution has been to expand the body of knowledge regarding the use of platforms among companies at which traditional product platforms have been difficult or even impossible to implement. The presented platform approach takes on issues with existing product platforms described in the literature in terms of their ability to evolve and manage heterogeneous content. The evaluation of the presented platform in a company has also enriched the existing literature on platforms within the areas of engineering design.
CONCLUSIONS

The following conclusions have also been drawn from the results of this research:

- Little to no research had investigated how suppliers of highly customized systems can develop and apply a platform that fits that context.
- The investigated companies all wish to work using a platform-based approach, but succeed to different extents.
- The investigated companies were all faced with fluctuating requirements during the scoping, quotation, and subsequent development processes.
- The investigated companies all possess different assets that could be used in scoping, quotation, and product development. The companies all would like to reuse appropriate assets but are currently not doing so to any significant degree.
- The companies experience challenges in attempting to obtain an overview of existing designs and other assets to be reused.
- The companies emphasize that the reuse of knowledge could be a key enabler in increasing competitiveness.
- In order to realize and apply a platform defined consisting of assets [i.e. knowledge, people etc.] as defined by Robertson and Ulrich (1998), a model and working approach are required.
- Design Elements have been shown to be enablers for systematizing and publishing knowledge.
- The Generic Product Items have been shown to be a way to model a generic product concept and to link knowledge described using Design Elements.
- The introduced platform approach shows great promise as a way for suppliers to gain the benefits of a platform-based approach by offering access to the development assets already existing at the company in a formalized approach.
- The Design Platform Manager has been shown to be a way to realize, in part, the platform approach. The tool can manage the platform model by creating Generic Product Items, variants, and Design Elements that are stored in the company’s PDM system database.
- The platform approach and support tool have undergone an evaluation that shows an overall good result in terms of supporting the designer, reusing knowledge, and decreasing lead times. The areas to be improved involve the support system, the user interface, and the level of automation regarding data input.

The scope of the need identified in this thesis was founded on a literature review and on interviews at four companies. The platform approach and support tool was then developed with this literature and the four companies in mind so that results would be generalizable to a significant extent. The results have only been applied to one specific company in this thesis, which limits the degree of generalization that can be undertaken with confidence. However, there is little in the results to suggest that they would not be generalizable over a wide range of companies. The Design Platform concept, which is the base of the platform approach presented in this thesis, has been applied to the four companies in Elgh et al. (2016) which adds to the applicability and generalizability of the result.

7.1 Future work

Future work will focus on further development, refinement, and implementation of the platform approach and support tool at more companies. Additional exemplifications are expected to increase the applicability of both the model and the tool. The PD process is often focused on two views: product and process, and the platform approach and support tool presently support a largely product-centered
view. Therefore, one focus of future research will be to develop a process view of the platform approach, allowing the modelling of use sequences of development assets. The Design Platform Manager will be extended with functionalities that allow for the modelling of processes and sequences for each variant that is created. When using the Design Platform Manager, the user will not only be presented with what resources have been used but also the order in which they were employed. A more intensive focus will also be placed on visualizing the platform approach to make its content and abilities clearer to users, which is expected to increase the possibility of using the DPM as a communication tool.

Future research will also place greater focus on feeding of the platform model with technology development so as to serve industry needs in the most holistic way possible.
CONCLUSIONS
REFERENCES


REFERENCES


REFERENCES


REFERENCES


REFERENCES


