JÖNKÖPING UNIVERSITY
School of Engineering

Doctoral Thesis

The Design Platform Approach – Enabling Platform-Based Development in the Engineer-to-Order Industry

Samuel André

Jönköping University
School of Engineering
Dissertation Series No. 48, 2019

Doctoral Thesis in Machine Design

The Design Platform Approach – Enabling Platform-Based Development in the Engineer-to-Order Industry
Dissertation Series No. 48

Publisher:
Jönköping University, School of Engineering
SE-551 11 Jönköping
Tel.: +46 36 10 10 00
www.ju.se

Printed by BrandFactory AB 2019

ISBN 978-91-87289-51-4

ABSTRACT

Manufacturing companies are continuously faced with requirements regarding technology novelty, shorter time to market, a higher level of functionality, and lower prices on their products. This is especially the case for companies developing and manufacturing highly customized products, also known as engineer-to-order (ETO) companies. The traditional view of the product lifecycle introduces the customer only at the sale and distribution phase, which is often concerned with identifying and transferring customer needs into fixed specifications that guide the development of end-consumer products. In the ETO industry, however, the customer is involved already at the scoping and quotation stage, and a significant amount of engineering needs to be performed for every customer order. Thus, ETO companies cannot work according to the traditional model described above since specific requirements are set directly by the customer, or a detailed requirements specification is missing and must be developed in cooperation with the customer. It is not uncommon that products are developed in joint ventures with the customer and run for several years, during which requirements change.

Product platform approaches have been generally accepted in the industry to serve a wide product variety while maintaining business efficiency. However, how to apply a product platform approach in ETO companies that face the reality described above is a challenge. Product platform approaches tend to require focused development of the platform, which, in turn, requires some knowledge about the future variants to be derived from the platform. The research presented in this thesis investigates the state of art and practice in the industry regarding the challenges, needs, and current use of product platforms. To respond to the identified need, a product platform approach is proposed that expands the scope of what a product platform has traditionally contained. The purpose of this proposal is to aid the development of highly customized products when physical modules or component scalability do not suffice. The resulting approach, the Design Platform Approach (DPA), provides a coherent model and methodology for heterogeneous engineering assets to be used in product development, supporting the activity of designing and existing solutions. The approach is based on identifying and modelling generic product and process items, which are the generic building blocks of the product, its structure, and the process of designing them. The generic product and process items are associated with the generic assets governing their design. By describing engineering assets that are the outcome of technology and product development, such as finished designs, design guidelines, constraints etc., in a standardized format, the DPA successively evolves.

This thesis outlines the DPA in detail and presents cases of applications that have focused on different aspects of the approach. Tools to support the DPA are presented and evaluated in different kinds of industries along with the specific methods used and literature summarization.

Keywords: Product Development, Engineering Design, Quotation, Customization, Supplier, Product Platform, Design Reuse, Adaptation

SAMMANFATTNING

Tillverkande företag blir kontinuerligt utmanade med krav på kortare ledtider, lägre priser på sina produkter och en högre nivå av funktionalitet och teknik. Detta är särskilt fallet för företag som utvecklar och tillverkar högt kundanpassade produkter, även kända som engineer-to-order (ETO) företag. Den traditionella synen på produktlivscykeln introducerar kunden i försäljnings- och distributionsfasen, som ofta berör identifiering och överföring av kundbehov i kravspecifikationer som styr produktutvecklingen av produkter för slutkonsumenter. ETO-branschen skiljer sig i att kunden redan är involverad i offertstadiet och att en betydande mängd ingenjörsarbete behöver utföras för varje kundorder. ETO-företag kan således inte fungera som tidigare beskrivna företag eftersom specifika krav ställs direkt av kunden. Även motsatsen kan inträffa då en detaljerad kravspecifikation saknas och behöver utvecklas i samarbete med kunden. Det är inte ovanligt att produkter utvecklas i gemensamma projekt med kunden och att projekt drivs under flera år under vilka krav tenderar att ändras.

Plattformsstrategier har accepterats inom industrin för att effektivt kunna hantera ett brett produktsortiment samtidigt som företagets effektivitet upprätthålls. En utmaning är dock hur ETO företag som står inför den verklighet som beskrivs ovan bör applicera en plattformsstrategi. Plattformsmetoder tenderar att kräva en fokuserad utveckling av plattformen vilket i sin tur kräver viss kunskap om vilka framtida varianter som ska skapas från plattformen. Forskningen som presenteras i denna avhandling undersöker litteratur och praktik inom industrin gällande utmaningar, behov och användning av plattformar. För att svara på det identifierade behovet föreslås en plattformsmodell och metod som utökar omfattningen av vad en produktplattform traditionellt har varit. Syftet är att stödja utvecklingen av höganpassade produkter när fysiska moduler eller skalbara komponenter inte räcker till. Det resulterande tillvägagångssättet, Design Platform (DP) -modellen, ger ett sammanhängande sätt för att hantera ingenjörstillgångar som ska användas vid produktutveckling och inkluderar både konstruktionsprocessen samt befintliga produktlösningar. Tillvägagångssättet bygger på att identifiera och modellera den generiska produkten och processen som är produktens generiska byggstenar, dess struktur och dess process. Dessa kopplas samman med de generiska tillgångarna som stödjer konstruktion och återanvändning. Genom att beskriva ingenjörstillgångarna, som är resultatet av teknik och produktutveckling, som färdiga konstruktioner, riktlinjer för konstruktion, krav etc. i ett standardiserat format, utvecklas plattformen successivt.

Denna avhandling presenterar DP-modellen och implementationer som har fokuserat på olika aspekter av DP-modellen. Flera verktyg för att stödja DP-modellen presenteras och utvärderas i olika branscher samt diskuteras i ljuset av den forskningsmetodik och litteratur.

ACKNOWLEDGEMENTS

This thesis was carried out in the Department of Industrial Product Development, Production and Design, Jönköping University, Sweden. I would like to express my gratitude to my supervisor, Prof. Fredrik Elgh, who gave me the opportunity to pursue a doctoral degree. You have given me the freedom to test my own ideas and have 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 (Verket för innovationssystem, VINNOVA) and the Knowledge Foundation (KK stiftelsen) for funding the research projects in which this research has been conducted. I am grateful 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, Inter Mekano Tools, and Flexator for permitting me access to their facilities and personnel, which were essential to the successful completion of this thesis.

Thank you to my co-supervisor, Dr. Roland Stolt, for constructive criticism, valuable input, and careful proofreading in the writing process. Great thanks also go to Joel Johansson for the impressive level of computer programming knowledge that he was so willing to share with me. I am grateful to PhD candidate colleagues for fruitful and challenging discussions.

I would also like to thank additional friends and colleagues at the School of Engineering, especially in the Department of Industrial Product Development, Production and Design. It is a joy to work together with you in this environment and I sincerely hope that our collaboration will continue.

Finally, I would like to thank 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.

Paper A Samuel André, Roland Stolt, Fredrik Elgh, Joel Johansson, Morteza Poorkiany (2014). *Managing Fluctuating Requirements by Platforms Defined in the Interface Between Technology and Product Development.* Proceedings of the 21st ISPE International Conference on Concurrent Engineering, 8-11 September, Beijing, China.

Work distribution
Samuel André performed the data analysis and wrote the paper. Fredrik Elgh synthesized the platform model idea. Roland Stolt and Fredrik Elgh proofread the paper. All included authors assisted with the research design and data collection.

Paper B Samuel André, Roland Stolt, Fredrik Elgh (2015). *Introducing Design Descriptions on Different Levels of Concretization in a Platform Definition.* Proceedings of the 12th IFIP WG 5.1 International Conference, PLM 2015, 19-21 October, Doha, Qatar.

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 the synthesis of the theory as well as the review.

Paper C Samuel André, Roland Stolt, Fredrik Elgh (2016). *A Platform Model for Suppliers of Customized Systems – Creating an Ability to Master Fluctuating Requirements*. Proceedings of ASME IDETC/CIE International Design Engineering Technical Conferences & Computers & Information in Engineering Conference, 21-24 August, Charlotte, North Carolina, USA.

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

Paper D Samuel André, Fredrik Elgh, Joel Johansson, Roland Stolt (2017). *The Design Platform – a Coherent Platform Description of Heterogeneous Design Assets for Suppliers of Highly Customized Systems.* Journal of Engineering Design, 28(10-12), 599-626.

Work distribution
Samuel André wrote the paper and synthesized the main parts of the theory. Samuel André, Roland Stolt, and Joel Johansson conducted the case applications and gathered the empirical data. All authors supported the synthesis of the theory, evaluation design, and review.

Paper E Samuel André, Fredrik Elgh (2018) *Modeling of Transdisciplinary Engineering Assets Using the Design Platform Approach for Improved Customization Ability.* Journal of Advanced Engineering Informatics, 38, 277-290.

Work distribution
Samuel André wrote the paper and developed the computer support tool. Samuel André and Fredrik Elgh synthesized the theory, evaluation design, and the execution. Fredrik Elgh supported by proofreading.

Paper F Samuel André, Fredrik Elgh (2019). *Supporting the Modelling and Managing of Relations in the Design Platform*. Proceedings of the 22th International Conference on Engineering Design (ICED), 5-8 August. Delft, The Netherlands.

Work distribution
Samuel André wrote the paper, synthesized the theory, and developed the computer support tool. Fredrik Elgh supported in proofreading.

Paper G Samuel André, Martin Lennartsson, Fredrik Elgh (2019) *PLM support for the Design Platform in industrialized housing for efficient design and production of customized houses.* Submitted to a journal.

Work distribution
Samuel André and Martin Lennartsson wrote the paper. Martin Lennartsson synthesized and analysed the empirical data. Samuel André synthesized the conceptual solution. Fredrik Elgh supported by proofreading.

ADDITIONAL PAPERS

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

- Roland Stolt, Samuel André, Fredrik Elgh, Joel Johansson, Morteza Poorkiany (2015). *Managing Risk in the Introduction of New Technology in Products.* Journal of Aerospace Operations, 3 (3-4), 167-184.
- Roland Stolt, Samuel André, Fredrik Elgh, Petter Andersson (2015). *Manufacturability Assessment in the Conceptual Design of Aircraft Engines – Building Knowledge and Balancing Trade-offs.* 12th IFIP WG 5.1 International Conference, PLM 2015, 19-21 October, Doha, Qatar.
- Joel Johansson, Samuel André, Fredrik Elgh (2015). *Simulation Ready CAD-models as a Means for Knowledge Transfer Between Technology Development and Product Development.* 20th International Conference on Engineering Design (ICED), 27-31 July, Milan, Italy.
- Samuel André (2016). *Towards a Platform Approach Supporting the Interface Between Technology and Product Development.* 14th International Design Conference DESIGN 2016, 16-19 May, Dubrovnik, Croatia.
- Fredrik Elgh, Samuel André, Joel Johansson, Roland Stolt (2016). *Design Platform – Setting the Scope and introducing the Concept.* 14th International Design Conference DESIGN 2016, 16-19 May, Dubrovnik, Croatia.
- Roland Stolt, Joel Johansson, Samuel André, Tim Heikkinen (2016). *How to Challenge Fluctuating Requirements: Results from Three Companies.* Proceedings of the 23rd ISPE Inc. International Conference on Transdisciplinary Engineering, 3-7 October, Parana, Curitiba, Brazil.
- Roland Stolt, Samuel André, Fredrik Elgh, Petter Andersson (2016). *Early Stage Assessment of the Inspectability of Welded Components: A Case from the Aerospace Industry.* SPS16, 26-27 October, Lund, Sweden.
- Fredrik Elgh, Samuel André, Joel Johansson, Roland Stolt (2017). Design Platform – A Coherent Model for Management and Use of Mixed Design Assets. 24th ISPE Inc. International Conference on Transdisciplinary Engineering, 10-14 July, Singapore.
- Dag Raudberget, Christoffer Levandowski, Samuel André, Ola Isaksson, Fredrik Elgh, Jacob Müller, Joel Johansson, Roland Stolt (2017). *Supporting Design Platforms by Identifying Flexible Modules*. International Conference on Engineering Design (ICED17), 21-25 August, Vancouver, Canada.
- Samuel André, Fredrik Elgh (2017). *Creating an Ability to Respond to Changing Requirements by Systematic Modelling of Design Assets and Processes.* 2017 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM), 10-13 December, Singapore.
- Roland Stolt, Samuel André, Fredrik Elgh, Petter Andersson (2017). *Introducing Welding Manufacturability in a Multidisciplinary Platform for the Evaluation of Conceptual Aircraft Engine Components.* International Journal of Product Lifecycle Management, 10(2), 107-123.

- Roland Stolt, Samuel André, Fredrik Elgh (2018). *Introducing Inserts for Die Casting Manufactured by Selective Laser Sintering.* 28th International Conference on Flexible Automation and Intelligent Manufacturing, FAIM 2018, 11-14 June, Columbus, Ohio, USA.

- Dag Raudberget, Samuel André, Fredrik Elgh (2018). *Modularisation in Two Global Product Developing Companies: Current State and Future Outlook*. NordDesign, August 14-17, Linköping, Sweden.

- Samuel André, Martin Lennartsson, Fredrik Elgh (2019) *Exploring the Design Platform in Industrialized Housing for Efficient Design and Production of Customized Houses*. 26th ISTE International Conference on Transdisciplinary Engineering, 30 July-1 August, Tokyo, Japan.

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

DPA – Design Platform Approach

DP – Design Platform

DPM – Design Platform Manager

DS – Descriptive Study

DSM – Design Structure Matrix

ETO – Engineer to Order

GPI – Generic Product Item

HPDC – High Pressure Die Casting

IHB – Industrialized House Building

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

UML – Unified Modelling Language

CONTENTS

1

INTRODUCTION

This chapter explains and argues for the need of this work with respect to challenges existing in the industry and research gaps in the literature. A broad description of industry practices and challenges is given, followed by the specific focus, aims, goals, and questions to be answered within the frame of this thesis.

1.1 Background

Manufacturing companies are continuously faced with the challenge of improving technological novelty, time to market, levels of functionality, product prices, and product lifecycles. This applies particularly to companies developing and manufacturing highly customized products where a significant amount of engineering is needed for each individual customer order. The traditional view of the product lifecycle introduces the customer during the sale and distribution stage—that is, when the product has been developed and produced. This type of business is often concerned with identifying and transferring customer needs into fixed specifications that guide the product development (PD) of end-consumer products. However, engineer-to-order (ETO) companies differ in that the customer is involved from the scoping and quotation stage (Elgh, 2012). Moreover, the ETO company is often part of a large supply chain that includes several intermediate customers, separating the ETO company from the end consumer.

Customization refers to the ability to design and manufacture tailored products for individual customers. Four different business models can be identified depending on where the actual customization starts: ETO, modify-to-order, configure-to-order, and select variant (Hansen, 2003). In the latter two, product platforms, which are defined by standard modules, components, and interfaces, have been successful enablers of efficient customization. For some industries, especially the ones directly supplying products to the end consumer, product platforms have been the single most important factor to stay competitive (Hvam, Pape, & Nielsen, 2006). One example is the car industry where configuration systems have enabled customers to configure individual car variants; these systems allow a relatively high level of customization and production standardization, while also managing complexity, which in turn provides the company with a competitive edge.

However, ETO companies cannot work this way because the customer directly sets the requirements, which may lack in specificity. It is not uncommon that products are developed in cooperation with the customer, often an original equipment manufacturer (OEM) or another supplier. These projects may run for several years, and the requirements may not be fixed at the outset. During the cooperative development stage, requirements often change. This has been investigated in the automotive industry where it was found to be a natural process since knowledge is gained and prerequisites change throughout the project (Almefelt, Berglund, Nilsson, & Malmqvist, 2006). These changes have many sources (Fernandes, Henriques, Silva, & Moss, 2015), but they often stem from the complex interplay between the involved suppliers, who use the same interfaces as references for their own development process. When a design requires change to shared interfaces, other suppliers' designs are also affected. This requires change from the affected sub-systems or changes to the requirements themselves. On these occasions, it is crucial to manage these requirement fluctuations and have a way to adapt to such ever-evolving situations.

Product platform approaches as enablers for customization have been widely accepted in the industry; they facilitate a large range of products while keeping internal variety low (M. H. Meyer & Lehnerd, 1997). Product platforms have also served effectively to reach different customer segments while maintaining commonality in product components and interfaces. Here, balancing the trade-off between commonality and distinctiveness is key to success (Halman, Hofer, & Vuuren, 2003). Recent research has focused on product platforms with a broader scope regarding definitions, which aim to reuse more of the skills and knowledge (i.e. assets) created in a company compared to the component-based product platform. From this perspective, it has been questioned whether companies could afford not to apply a product platform (H Johannesson, 2014).

However, less investigation has been conducted on how to apply a product platform approach to ETO companies, which do not have the advantage of interface standardization and component commonality. Component-based product platforms tend to require proactive and focused platform development including late-stage customer involvement, which in turn requires some knowledge about which future variants are to be derived from the platform. This kind of forecast is hard or impossible for ETO companies, whose main competitive edge lies in a high level of customization and for which the interfaces with the customer product are unknown beforehand. One factor that amplifies the challenge for ETO companies is the separation of technology development (TD) and PD. It has been suggested that splitting TD and PD decreases risk in customer-focused projects (Säfsten, Johansson, Lakemond, & Magnusson, 2014). TD often has a long-term goal of supplying a relatively uncertain future market with new technology, whereas PD has a more short-term character and fulfils specific customer requirements. Conducting an efficient TD requires that new initiatives are proactively planned to fit a future market situation. This is a challenge for ETO companies since the future customer requirements are unknown and product platform development is closely related to the forecast driven development represented by TD.

There is increasing demand for complex products that require mechanics, electronics, and embedded code and that must be developed through cooperation between design and analysis, as well as be evaluated for producibility. This complexity ultimately places high demands on companies' abilities to work in a transdisciplinary fashion. As disciplines within companies, such as design, purchase, analysis, and aftermarket, are centralized to specific departments, it becomes more crucial and complex to obtain an overview of the engineering assets. These assets are used for TD and PD, can be reused in a range of products, and allow a common information model to be used for communication.

Thus, ETO companies are developing products in a challenging environment unlike traditional schoolbook examples. At the same time, for many companies, product platforms have been an enabler and a necessity to remain competitive in an increasingly challenging industry. In light of this, there is a blank spot regarding if and how ETO companies can take advantage of the product platform concept. This question forms the focus of this thesis.

1.2 Aim, goal, and intended contribution

This section describes the aim and goal of this thesis. An aim is a desired future state for which you strive. Therefore, it is not reasonable to expect that this aim will be reached during the time in which this thesis was written. A goal, on the other hand, can be expressed in a way that it is obtainable and maybe also measurable. By fulfilling goals, you should, therefore, get closer to and move in the direction of your aim.

The aim of the research is to enable ETO companies developing highly customized products to be more efficient during product development. In more specific terms, this efficiency includes providing means to identify, develop, manage, and maintain the engineering assets which reside in companies and embodies their know-how. This can, in turn, improve a company's ability in several situations, such as responding to fluctuating requirements during the scoping, quotation, and subsequent development processes. Other improvements include the possibility of reusing engineering assets and of assessing the implications of change.

The scientific goal is to contribute to knowledge on product platform approaches, in terms of models and methods, in settings where product platforms have traditionally been difficult to implement. This thesis further aims to exemplify the application of an alternative approach to the component-based product platform approaches found in the literature. The research area targeted for the contribution is platform-based development. The intended contribution, therefore, aspires to be a product platform approach, which includes a coherent methodology and model, builds on the theories of platform-based development, and supports ETO companies. The industrial goal of this thesis is to propose an approach to identify, develop, and manage engineering assets within a product platform context. The ETO industry can thus be provided with means to work in a structured way that allows it to become more efficient by increasing the utilization of different assets continuously developed in a company. This study intends to provide the industry with such an approach, along with demonstrators that make it possible for companies to conduct platform-based, as opposed to single product, development.

1.3 Research focus

Product platforms are well established in the literature (Pirmoradi, Wang, & Simpson, 2014; Timothy W Simpson, Jiao, Siddique, & Hölttä-Otto, 2014). More recently, technology platforms have also been of interest (H Johannesson, 2014). However, there are few studies that describe a coherent product platform approach for and its application in ETO companies. This thesis focuses on how ETO companies can benefit from a product platform approach and thus work platform-based when a solely component-based product platform is not a viable option. Component-based refers to when a product platform is predefined in terms of variants or configurable modules and no direct engineering work is needed to deliver an order. When a product is ordered based on a component-based product platform, the production specifications (such as drawings) are readily prepared. This is hard to achieve for ETO companies. Further, the research focus is to propose an approach that is grounded in empirical data and previous research and that consists of a product platform model and a method prescribing how to set up and execute it.

1.4 Research questions

The research focus regarding the challenge of product platform application in ETO companies described above is summarized in the following question: *How can ETO companies be supported in using a product platform approach?* The word "approach" refers to one or several models and methods that are supported by suitable tools.

In order to elaborate on this question, it has been broken down into the following three research questions, which are the focus of the complete thesis:

- **RQ1:** *What is the current state of the utilization of product platforms for ETO companies?*

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

- **RQ2**: *How can a product platform approach be conceptualized to support customization for ETO companies?*

A key assumption, which is supported by the literature (Ulf Högman, Bergsjö, Anemo, & Persson, 2009), is that ETO companies cannot fully apply a traditional component-based product platform. Building on the result from RQ1, this question aims to explore and develop a suitable model and method for such a company and to report on what elements the model could contain.

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

To make use of the approach resulting from RQ2, the approach must be supported. This question concerns how such support could be created and evaluated.

1.5 Scope and limitations

This research, like most PhD theses, has been subjected to a time limit of 4 years and a specific set of research projects, as well as companies and research colleagues. These parameters create the main frame of the research work.

The research focuses on a specific group of industrial companies that develop highly customized products, i.e. ETO companies, often in a business to business relation. The results are expected to be generalizable across a broad range of companies, but this element is not specifically evaluated in this thesis.

PD concerns many artefacts, processes, and people. Organizational and management issues coupled with the forms of 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. Because the real effects of change and attempts to improve the industry situation can take longer than the time available for completion of this thesis, it is not reasonable to expect a comprehensive validation of the approach presented in this thesis. The evaluations performed and presented in this study should, therefore, be viewed as initial steps towards a fully evaluated approach.

Though an efficient product platform spans several company functions, PD, including design, is the focus of this thesis. This means that issues connected to the production system are out of its scope.

1.6 Outline of the thesis

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

Chapter 2 outlines the research methodology applied in this work. It presents methods and models in a generic manner and their application in this study.

Chapter 3 presents the frame of reference, which is an assortment of literature that identifies best practices, the current state of the industry, and fundamental theories. The chapter concludes with a summary identifying the research gap that this thesis strives to fill.

Chapter 4 outlines a novel product platform approach and the main result, as synthesized from the appended papers.

Chapter 5 summarizes the results from the appended papers and points to the progression and evolution of the research across the papers.

Chapter 6 outlines the discussion. It focuses on the results while taking into consideration the literature, the validity of the research, and the research questions.

Chapter 7 briefly summarizes the main conclusions and takeaways from the thesis. It also proposes future work to be conducted.

2

RESEARCH METHODOLOGY

The research methodology aims to describe how a study has been conducted. The methodology used, including its reliability and validity, determines the quality of the research result. This section first outlines the basics of the methodologies used. It then describes how these methods have been applied in this specific research.

2.1 Design research

Design has been defined in various ways, many of which depend on the culture and background of the author (L. Blessing, 2003). However, many agree that design brings together artefacts, people, tools, processes, and organizations, making it a highly complex and even chaotic area (Horvath, 2004). This may partly explain why the validity of design as a research field has been questioned. Design research is still quite a young field, which has yet to be fully explored. 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 (L. T. Blessing & Chakrabarti, 2009; Duffy & Andreasen, 1995; Hubka & Eder, 2012). A descriptive study refers to the objective examination of phenomena using techniques that make it possible not to interfere with the object of study. In contrast, a prescriptive study requires the researcher to take part actively in the studied situation with the aim of improving it. This is different from many other research branches that focus mainly on understanding a phenomenon without affecting it. It has been stated that design research, like several other research fields within the social sciences, includes phenomena that, from a constructivist view, do not await discovery in the same way as physical laws (Checkland & Holwell, 1998). Therefore, design research challenges the traditional research approach for testing hypotheses by requiring the researcher to partake in the studied situation in order to spread the knowledge to practitioners and to understand the phenomena to a higher degree. The action and change involved in design research often concern the introduction of novel methods and models, which are frequently embodied in computer-based tools and created to support the design process. The aims are to support designers in their practice, to create knowledge of the domain, and to generalize the knowledge to similar settings.

2.2 Design support models

Regarding the development of computer-based models, Duffy and Andreasen (1995) propose the approach shown in Figure 1. This approach supposes 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 (as-is). These models are refined into information models that act as blueprints for the desired state (to-be). The information models are 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 reality, insight can be gained into the process, which, in turn, can be used to improve the prescriptive models. In essence, this model is not directly operational, but it can describe how a design researcher relates to these different domains.

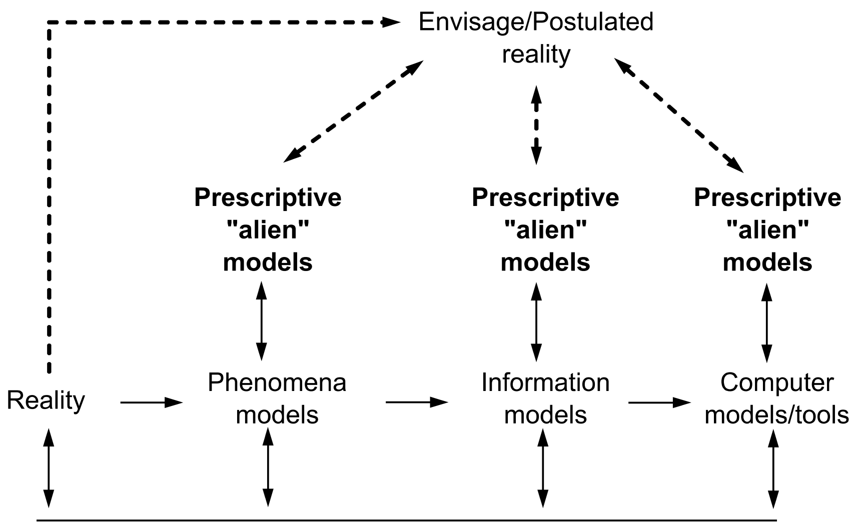

Figure 1. Design modelling research approach, according to Duffy and Andreasen (1995)

2.3 Research approach

The overall research framework used for this study was proposed by Blessing and Chakrabarti (2009) and is called a design research methodology (DRM). The outline of the approach is shown in Figure 2. The framework is partly based on the fact that design science not only strives to create knowledge about a phenomenon but also tries to improve the design process itself. Assumptions based on both understanding and beliefs are made by the researcher regarding how to accomplish this improvement.

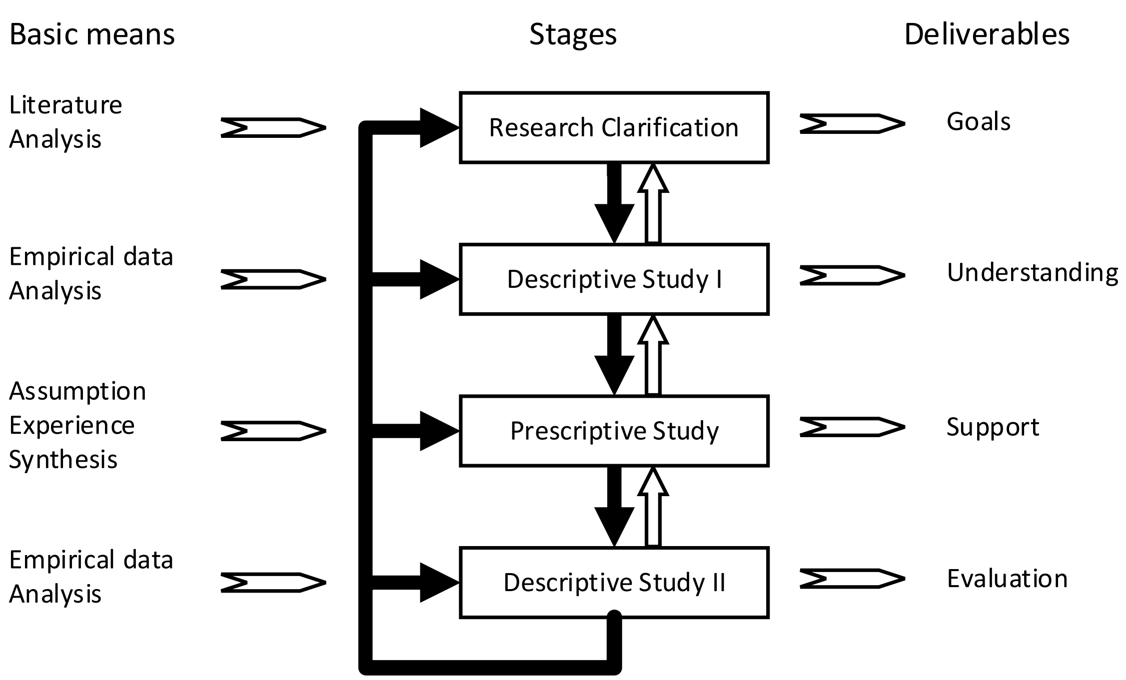

Figure 2. The generic design research methodology framework according to Blessing and Chakrabarti (2009)

It is important to note that this framework is not sufficient to perform rigorous research; indeed, each step needs to be filled with relevant methods and techniques in order to produce credible results.

The case study is one example of an approach that can be employed in the descriptive phases. Eisenhardt and Graebner (2007) reason that case research can help to create an understanding of a phenomenon by demonstrating its nature and complexity in its real setting. Sousa and Voss (2001) recommend the case study as a methodology when the research question is exploratory or embodies an exploratory component, which is often the case in engineering design research. Other approaches relevant to the prescriptive phases of DRM are action research and systems development, which are further outlined in a subsequent section. The DRM framework is created to support research within engineering design, which makes it suitable for this thesis. Furthermore, it is vital to have a system perspective on design, given its complex nature. The system perspective implies that no phenomenon can be studied or affected without a greater understanding of the whole system. A lack of system perspective might result in sub-optimal solutions and an insufficient understanding of the research object's relationships to other, directly related phenomena.

A summary of the main phases follows:

- *Research Clarification (RC)* refers to the activity in which researchers search for evidence to support their assumptions. A general understanding of the research field is sought, mainly by studying and analysing literature.
- *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 the practice). For evaluation purposes, for example, success criteria can be created as a datum.

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. At this stage, the support is verified by investigating how well it functions by itself.
- *Descriptive Study II (DS-II)* describes the phase in which the support is validated through testing in the intended environment. At this stage, the support's contribution to the planned success of the research is assessed through comparison with the initial success criteria.

DRM proposes seven ways of applying the previously described methodology (L. T. Blessing & Chakrabarti, 2009). Depending on the characteristics of the research project, each DRM stage (Figure 2) can be applied to different levels of depth. *Review-based* refers to a result produced by analysing existing literature. *Comprehensive*, on the other hand, refers to a result and method that requires that new knowledge to be 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 and in which the focus instead is on preparing a result to be used by other researchers. These types are visualized in Table 1.

Table 1. Research types adapted from (L. T. Blessing & Chakrabarti, 2009)

	Research Clarification	**Descriptive Study I**	**Prescriptive Study**	**Descriptive Study II**
1.	**Review-based**	Comprehensive		
2.	**Review-based**	Comprehensive	Initial	
3.	**Review-based**	Review-based	Comprehensive	Initial
4.	**Review-based**	Review-based	Review-based Initial/Comprehensive	Comprehensive
5.	**Review-based**	Comprehensive	Comprehensive	Initial
6.	**Review-based**	Review-based	Comprehensive	Comprehensive
7.	**Review-based**	Comprehensive	Comprehensive	Comprehensive

2.4 Applied methods within the DRM framework

As previously described, DRM constitutes the research framework used in this work. To make it operational, however, additional methods have been utilized that either focus on an area resembling one in the research setting or give detailed practical support for conducting this research.

2.4.1 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. Action research stems from a constructive viewpoint, which holds that social phenomena do not await discovery in the same way as physical laws (Kock, 2007), and thus it challenges the claims of a positivistic world view (Brydon-Miller, Greenwood, & Maguire, 2003). A key characteristic that differentiates action research from other research designs is the fact that the researcher interacts with the studied situation and intends to improve it.

Design research is similar to action research in that it aims to bring about improvements while also creating knowledge (Williamson, 2002). Action research is usually carried out in cycles of action and reflection, as shown in Figure 3. Each action generates a result that leads to reflection, and the next action stage is based on the result of the reflection. The pattern continues for as many iterations as 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. It should be noted that both formal and informal reflection and action cycles occur throughout a research project.

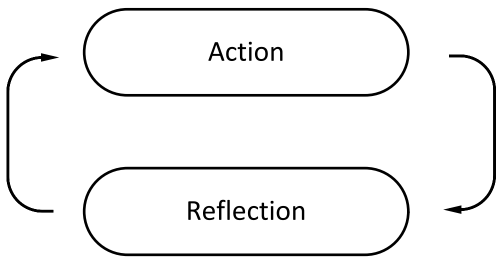

Figure 3. The most basic action research cycle adapted from (Williamson, 2002)

Action research has been accused of being less scientific than other research approaches. There are three components to this perception (Avison, Davison, & Malaurent, 2017).

- Action research is perceived to be less rigorous than other methods.
- It is difficult to make theoretical contributions from investigations based on this method.
- Action research is very similar to consulting.

Other criticisms concern the subjectivity of the researcher and the difficulties of generalization. However, Avison, Davison, and Malaurent (2017) claim that the criticism regarding rigor lies more in a faulty understanding of the word than in the approach of action research itself. Avison, Davison, and Malaurent (2017) discuss theory building as a strength of action research due to its close connection with practice and thus reality. They claim that action researchers are not consultants because the researcher is often financed by a source other than the client and has a more holistic perspective. Also, for a consultant, the client is typically the company or the organization which finances him or her, while, for a researcher, a client consists of all stakeholders and the research community at large.

2.4.2 Systems development

Systems development is an approach within applied research. It is based on the belief that development is always associated with the exploration, advanced application, and operationalization of theory. This research approach arose in the field of information science as a way to manage the multi-disciplinary characteristics of, and to bridge the gap between, the field's technological and social sides (Williamson, 2002). Systems development 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 developing support for, realizing, and prototyping models in engineering design. Nunamaker Jr, Chen, and Purdin (1990) have argued that it is a central part of a multi-methodological information systems research cycle, as seen in Figure 4. Systems development becomes an intermediate step that links basic and applied science and supports the connection of theory, descriptive studies, and experimentation. However, though a prototype can be used as a proof-of-concept, it should not be viewed as a research contribution in and of itself (Nunamaker Jr et al., 1990); rather, a generic method or model should be developed concurrently. As the conceptual method or model is formalized in a software, one can quickly identify strengths and weaknesses of the concept. As the development proceeds, qualitative and quantitative techniques can be utilized to evaluate the performance and impact of the prototype by integrating practitioners in the research.

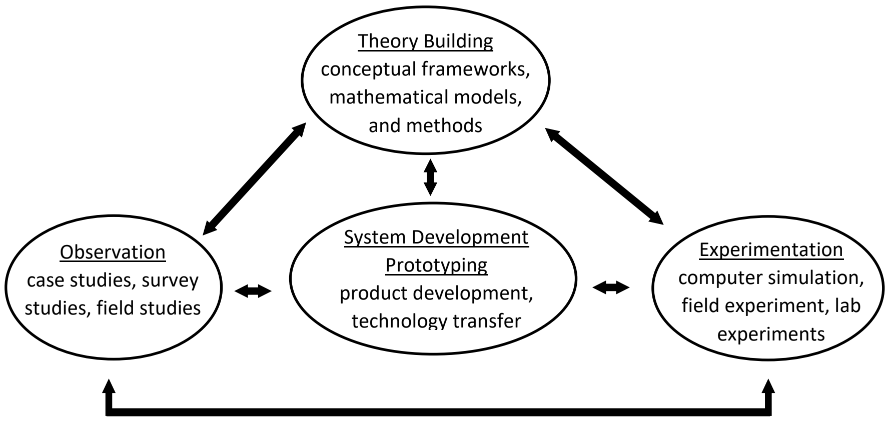

Figure 4. A multi-methodological approach to information system research, adapted from Nunamaker Jr, Chen, and Purdin (1990, p. 94).

2.4.3 Connection and relevance of the research methods and models

A clarification needs to be made regarding how the four methods presented—DRM (L. T. Blessing & Chakrabarti, 2009), action research (Kock, 2007), systems development (Nunamaker Jr et al., 1990), and a model of computer support within engineering design (Duffy & Andreasen, 1995)—are connected on a general level. The connection is visualized in Figure 5 and shows three domains. The DRM framework is the main framework, which has been used to create and understanding to affect the reality. The descriptive study utilizes traditional qualitative techniques and produces a model of the phenomenon of study, a so-called as-is description. At this stage, a research gap might already have been identified. An industrial problem can also be the starting point of the research, which is discovered in the descriptive phase and must be supported by literature by returning to the clarification stage. This as-is description is used to create assessable criteria to be use later in the process. Another outcome of the descriptive phase is a description of the outlook, a so-called to-be description. This description focuses on the desired state, which is also tied to the assessable success criteria. Thus far, the action in the research is kept to a minimum, which makes the step suitable for systematic literature reviews and case studies.

In the prescriptive phase, the systems development process is utilized. A conceptual model or method to support the identified challenge is developed in the conceptual domain and realized using the proof of concept domain. The concept and its realization are iteratively developed, presented, and evaluated in a real industrial setting, which potentially impacts the objects of study. Hence, the prescriptive study resembles action research. A strength of systems development is that it allows concurrent conceptualization and creation of a proof of concept. The developed prototype is used to verify the information model and to validate the model in its intended setting.

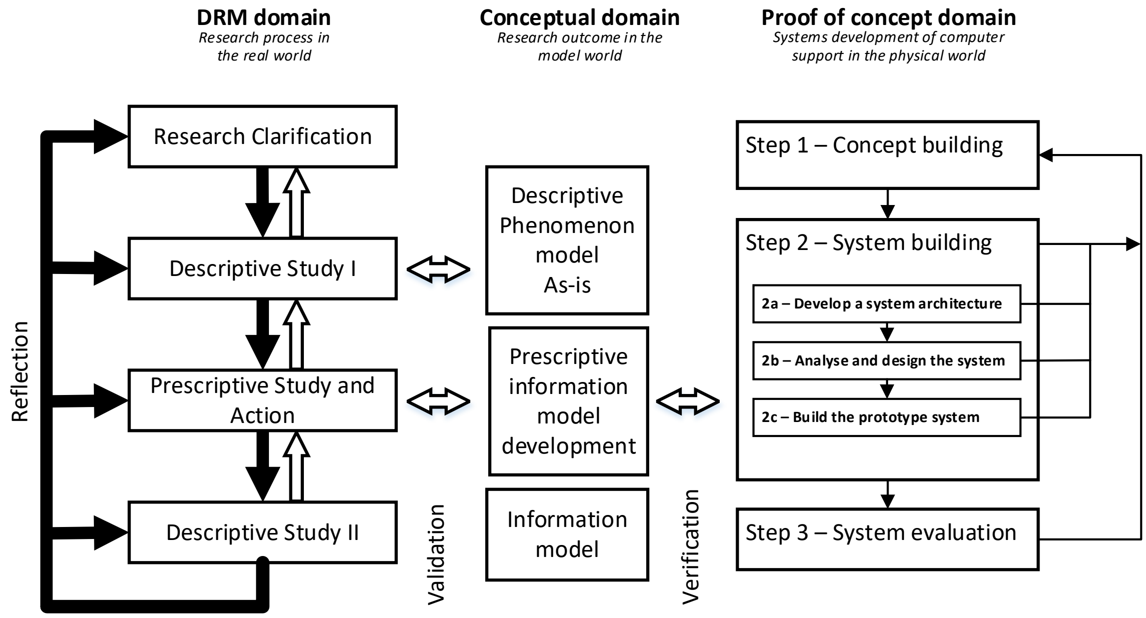

Figure 5. The connection between applied research models and method

2.5 Data collection techniques

Due to its nature, this work relies on qualitative data collection methods. Interviews serve as a way to obtain firsthand qualitative responses from design engineers who possess vital information. Interviews can be classified as follows (Williamson, 2002):

- 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 allow the interviewer to 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 interviewee. It can be used as a preliminary step before creating a structured interview or a self-administered questionnaire.

Workshops are closely related to interviews. They can be more or less structured, focusing on one or more predetermined issues, questions, or subjects. First, questions are posed in an open manner; then, initial discussion is held in a structured fashion. The workshop can have either a descriptive or prescriptive focus; the former aims to find out, for example, a company's current practice, and the latter aims to identify or create new methods and models that would improve a certain situation.

An alternative data collection method is the self-administered questionnaire. This method can be useful for gathering information from large groups in order to build quantitative data and allows open-ended questions. It provides a way for respondents to remain anonymous, which could be paramount when handling sensitive issues. On the other hand, it does not allow for follow-up questions in the same way as an interview.

In studying the knowledge-intensive PD process, it is important 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. In some cases, documents say more about the actual process in a company than information coming from a practitioner.

2.6 Quality of the research

Reproducibility—the ability of independent researchers to obtain the same (or similar) results when repeating an experiment or test—is one of the hallmarks of good science, according to (Popper, 2005). This is, however, a tough requirement in several research branches, such as social sciences, where one does not assume that "social laws" await discovery in the same way as physical laws. Engineering design is similar in this regard since its settings include complex interactions between people, artefacts, and processes (Horvath, 2004). Action research, along with other forms of qualitative research, is unable to match the complete replicability of experimental results. Researchers using this method must, therefore, at least achieve a situation in which their research process is recoverable and transparent (Kock, 2007). One way of improving the reliability of qualitative research is to increase the number of data points that are considered. According to Yin (2014), multiple case research is beneficial when a phenomenon is investigated from different perspectives as it helps to detect patterns and gain a deeper understanding. Multiple case study can contribute to more reliable findings and enhance research transferability.

Evaluation is crucial to guaranteeing the quality of any research. Evaluation involves comparing a number of criteria to data of some kind, such as requirements or success criteria (L. T. Blessing & Chakrabarti, 2009). Evaluation can be divided in two components: *verification* and *validation.* However, the definition of these two components varies in the literature. 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 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 (Kossiakoff et al., 2011, p. 393). It 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. Parts of this thesis have utilized success criteria as a validation method, the details of which are outlined in the summarization of the respective paper. However, the characteristics of this research are mainly qualitative. Olesen (1992) provides five criteria that can be used to assure the validity of qualitative research:

- *Internal logic* – known and accepted theories are the basis of the research, and the work is stringent from the 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.

2.7 Research projects

The research conducted within the scope of this thesis emerged from participation in three different research projects, which are described in detail in the following.

2.7.1 ChaSE

The first project is called ChaSE (Challenge Fluctuating and Conflicting Requirements by Set-Based Engineering). The three-year project ran from Q4 2013 to Q1 2017, gathering four companies. The aim of the project was to determine how companies can develop adaptable solutions to respond efficiently to fluctuating and conflicting requirements. The project was a joint effort between Jönköping University, The Swedish Agency of Innovation Systems (VINNOVA), and four companies developing customized products.

The project was used to gather empirical data from the involved companies, to develop an initial product platform approach, and to use company representatives to evaluate the approach. The involved companies provided empirical data regarding state of practice and to-be states. The product platform approach and introduced support tools were synthesized by the research team and developed based on the literature and gathered empirical data.

2.7.2 ProAct

The third project is called ProAct (Platform models for agile product development— building an ability to adapt). The three-year project runs from Q1 2017 to Q4 2019, gathering four companies. The project focuses on ETO companies and how they can model and apply product platform approaches to fit their specific business model. The project is funded by Region Jönköping County.

The project was used to gather empirical data from the involved companies regarding state of practice and to-be states. The product platform approach that was developed and evaluated in the ChaSE project was further applied to one of the companies in the project. The project was used principally to validate the product platform approach by applying it in an industrialized house-building setting.

2.7.3 Distinct

The second project is called Distinct (Design methods for customized products when introducing additive and cyclic manufacturing). The three-year project runs from Q2 2017 to Q1 2020, gathering five companies. This project focuses on product platform architectures that can cope with requirements regarding circular manufacturing and the new technology of additive manufacturing. It investigates business models and how a product platform architecture can support and take advantage of this environment. The case application of the project is the high-pressure die cast industry, which includes the complete production change from design to manufacturing. The project is funded by the Knowledge Foundation (KK).

The project was used to gather empirical data from the involved companies regarding the specific business environment in which they were active. The involved companies provided empirical data regarding state of practice and to-be states. The product platform approach and introduced support tool that had been developed and evaluated in the ChaSE project were further refined and applied to one of the companies in the project. The project was used both to refine and validate the product platform approach by applying it in a tool design and manufacturing setting.

2.8 Application of the research methodology

The research presented in this thesis has been executed using a combination of research methods. However, DRM, as proposed by Blessing and Chakrabarti (2009), frames the research work as a whole. The application of DRM includes four loopbacks of the last two stages, meaning that both PS and DS-II are initiated four times. It is also natural, when applying DRM, to do smaller iterations back to the RC and DS-I stages in order to clarify research questions, keep up-to-date, broaden the literature base in a certain area, and to fine-tune the path of argumentation according to the needs of the research project. The following section focuses on how the research design was established, resulting in each paper included in this thesis. Table 2 provides a summary of the connections between papers, research question, DRM stages, and characteristics of the paper content. As seen, Papers A and D report on studies conducted within the ChaSE project and thus use the same companies in the research setting. Similarly, Papers B, C and E are outcomes of the ChaSE project but with a more detailed focus on one of the companies. In this project, success criteria were used as a validation method in order to develop an approach that supported ETO companies. Papers F and G are outcomes from the projects Distinct and ProAct and are concerned with the application of the approach in other industries to strengthen the thesis´s validity. Each paper has been peer-reviewed, both within the research group and as part of the publication process of the relevant conference or journal.

- *Paper A* is mainly concerned with the RC stage and the interface between the RC and DS-I stages. The paper relies on a thorough systematic literature review that pointed to a research gap. The outcome of the 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

Table 2. Summarizing the connection between papers, research questions, DRM stages, and paper content characteristics; inspired by Levandowski (2014, p. 39)

Paper	RQ	DRM step	Multi/single case	Research project	Emperical study	Theory	Tools	Application of tools
A	1	RC — DS I — PS DS II	Multi	ChaSE	⬤	●		
B	2	RC DS I — PS — DS II	Single	ChaSE	●	⬤	●	
C	2, 3	RC DS I — PS — DS II —	Single	ChaSE		●	●	●
D	1, 2, 3	RC DS I — PS (2nd) — DS II (2nd) —	Multi	ChaSE	●	⬤	●	●
E	2, 3	RC DS I — PS (2nd) — DS II (2nd) —	Single	ChaSE		●	⬤	⬤
F	1, 2	RC DS I - PS (3rd) — DS II (3rd) -	Single	Distinct	●	●	⬤	⬤
G	2, 3	RC DS I - PS (4rd) — DS II (4th) -	Single	ProAct	●	●	⬤	⬤

- planned in collaboration with the research group. The interview study was conducted at four companies, with two to four people from different levels of the organization representing each company.
- *Paper B* 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 A, which was partly based on the 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 to sketch the emerging concepts 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, in a similar way to the action and reflection steps in action research.
- *Paper C* used both unstructured in-depth interviews and review of documents. The model that was proposed in Paper A was conceptualized using unified modelling language (UML). Both the information model and the computer model were iteratively refined using the mind-set developed by (Duffy & 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 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 stages.
- *Paper D* includes results from semi-structured interviews and workshops from four companies. 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 resolve the questions posed by the overarching ChaSE project description. The final evaluation used a self-administered questionnaire and enabled the respondents to grade the level to which the success criteria were fulfilled. The overarching model was developed as a joint effort within the research group, using several documented workshops. This paper describes DS-I and the second loop of PS and DS-II, including a refined support tool and application in four cases and the second evaluation.

- *Paper E* includes results from semi-structured interviews and workshops from a single company and is an in-detail description of one of the cases in Paper C. 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 resolve the questions posed by the overarching project description. The final evaluation used a self-administered questionnaire and enabled the respondents to grade the level to which the success criteria were fulfilled. The overarching model was developed as a joint effort within the research group, using several documented workshops. This paper describes the second loop of PS and DS-II, including a refined support tool, detailed application in one case and the second evaluation.
- *Paper F* aims principally to validate the approach developed in the previous papers by application in a new setting. It includes results from semi-structured interviews and workshops from a single company. The model developed in the previous papers is extended from a product focus to a process focus, and a tool is developed and presented. The tool was applied to company specific data and validated through demonstrations and discussions with the company.
- *Paper G* similarly aims to validate the approach developed in the previous papers by application in a new setting. It includes the results from semi-structured interviews and workshops from a single company active within the house-building industry. The paper presents the identification of engineering assets and connects them to a product platform approach. Further, a conceptual Product Lifecycle Management (PLM) concept is proposed to support the described company situation. The tool was applied to company specific data and validated through demonstrations and discussions with the company.

Bearing in mind results and methodological work in these papers, the most appropriate DRM research type (Table 1) is a combination of *Type 5* and *Type 7*. This combination is due to the loopbacks that have been performed between DSII and PS, which are part of *Type 7.* However, the loopbacks shall be seen as iterations that are natural in action research and not as a complete evaluation of the results. The evaluation is, therefore, to see as *initial*. *Type 7* is often considered as the most extensive research project according to the DRM methodology. Therefore, it is important to clarify that some of the research results have been made possible because a complete research team has been contributing in the research projects.

3

FRAME OF REFERENCE

This chapter presents a selection of research that has been of interest for this work. This includes adjacent fields of research, as well as the specific area to which this research contributes. The chapter ends with a summary and by stating the scientific motivation and knowledge gap identified from this literature review.

The focus of this thesis, and the area to which it contributes, is platform-based development. This concept has many connections to other areas, which makes it useful to explain how the different areas included in the frame of reference relate. Product platform development spans a large space, covering the early stages of development that are often conducted in technology development projects (Cooper, 2006). A product platform is a significant investment and is, therefore, preceded by technology development and thorough investigations of the customer segments to be covered by the variants derived from the product platform (Hvam et al., 2008). For this reason, requirements management becomes crucial to identifying needed customer features, transferring these into requirement specifications, and tracing the requirements throughout development, as they tend to change. In ETO-oriented industry, however, these processes are not always as predefined, and, therefore, this chapter also investigates other approaches. Product development often follows technology development through technology transfer, and concepts like customization become strategically important and a challenge for design departments. The relevance of product platforms for engineering design often has to do with design knowledge reuse, which makes it a subject worth exploring. This section also considers product platforms themselves through the vast body of research that describes models and methods to develop and apply them in different settings.

3.1 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). Holmdahl (Holmdahl, 2010, p. 51) offers the following definition of PD's aim: "The aim of product development is to increase the quality of life for one human being without decreasing it for another". However, primarily PD should serve to move 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 (K. T. Ulrich & Eppinger, 2012), as shown in Figure 6.

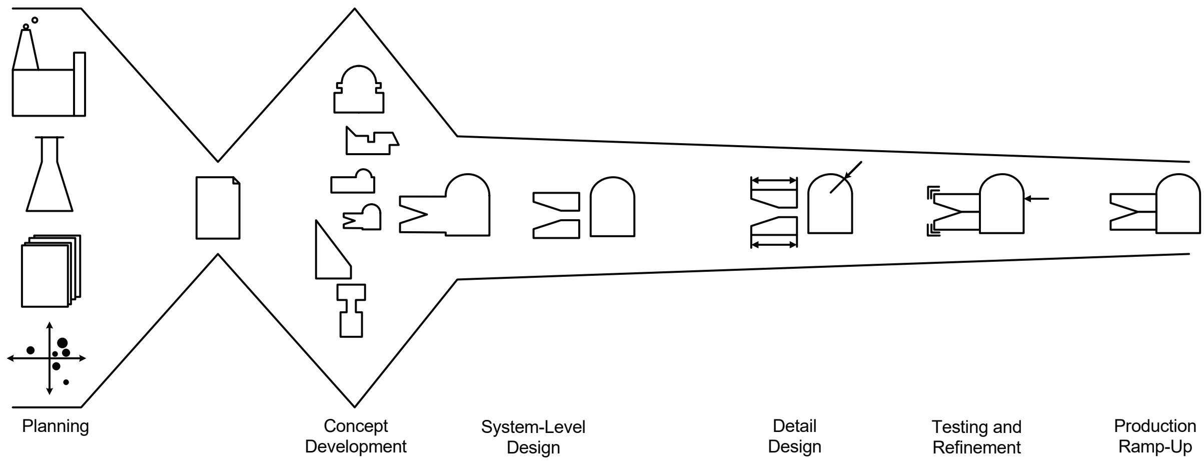

Figure 6. A generic product development process, according to (K. T. Ulrich & Eppinger, 2012)

3.2 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 that allow customization of products at a low cost. Customization refers to abilities and strategies that aid in designing and manufacturing products tailored to 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. Simpson (2004) further proposes product families as a primary enabler, describing two basic approaches to the design of product families to achieve efficient customization: (1) Top down, which describes 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 differs. Suppliers with an ETO business approach often find themselves in an environment in which several intermediate steps, which involve different stakeholders, other suppliers, and an OEM, differentiate them from companies focused directly on the end customer. This situation introduces several interfaces and stakeholder interests that the ETO supplier must manage. Yet holistic research in this area, taking all or at least several of these perspectives into consideration, remains scarce. Tuli and Shankar (2015) describe lean-in collaboration between the supplier and OEM and review the existing literature on supplier, OEM, and customer integration. One interface towards the customer that is central to customization is the customer order decoupling point (CODP). The CODP 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). It 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 levels of CODP categorization , which can also be seen in Figure 7.

- ETO is the highest level of customization and is characterized by the fact that engineering work needs to be conducted for each individual customer order. The product offer is usually broad in terms of customer choices.
- Modify to order (MTO) is similar to ETO with the difference that the product specification can be supported by predefined modules and sets of rules that govern the design of product variants.
- Configure to order (CTO) businesses aim to provide automatically configured product vagrants based on fully defined product parts, modules, and configuration rules. A configuration system is commonly used to support this business approach.
- “Select variant” is the lowest level of customization and is best exemplified by a store-bought product that can be found on a shelf. Variants of the same product type may exist to target different customer groups willing to pay varying prices based on the product’s appearance and functionality. These types of products are often packaged, and further adaptation is typically not possible.

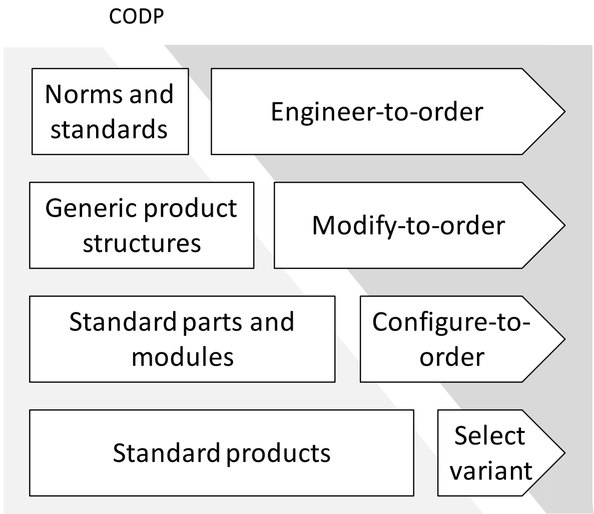

Figure 7. Customer introduction in the order specification process, adapted from (Hansen, 2003)

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 occurring before production. These authors propose a two-dimensional approach that takes into consideration the engineering adaptation that occurs for each customer order.

When it comes to customization, a challenging trade-off occurs between flexibility and rigidity. Fredriksson and Gadde (2005) have investigated the implementation of module assembly units (MAUs), a combination of build-to-order and mass production that offers a feasible strategy to balance the trade-off. Fredriksson and Gadde (2005) also emphasize that the range of options for the customer must be constrained to control the number of variants. Michael et al. (2007) likewise argue for the necessity of constraints in customization offers to decrease problems with customers' insisting on custom modifications, which, in turn, negatively influence the company's financial position. Other trade-offs are investigated by Squire et al. (2006), who use 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.

3.2.1 Mass customization

The concept of mass customization is found in automotive, clothing, and computer manufacturing, as well as in the food, electronic, and mobile phone industries (Fogliatto, da Silveira, & Borenstein, 2012). It has two definitions. 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" (Hart, 1995). 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). Hart (1995) continues by defining and describing the four pillars involved in answering the question, "Should we pursue an explicit mass customization strategy?" The pillars are as follows: customer sensitivity, process amenability, competitive environment, and organizational readiness. According to Hvam et al. (2008), there are two main enablers for mass customization: (1) product ranges should be developed on the basis of modules, and (2) configuration systems should help to relate the tasks involved in custom-oriented business to the specification of customer-specific products. The authors continue by describing the three kinds of companies involved in mass customization. The first is a firm that has previously produced and sold identical products in large runs; these companies tend to decline and evolve into companies that strive to adapt their products to the individual customer. They are best situated to pursue a mass customization strategy (Figure 8). The second type of company is one that develops and manufactures large complex products; here, the excessive development is initiated directly by the customer. In order to make the process more efficient, Hvam et al. (2008) suggests separating development from detailed design. The

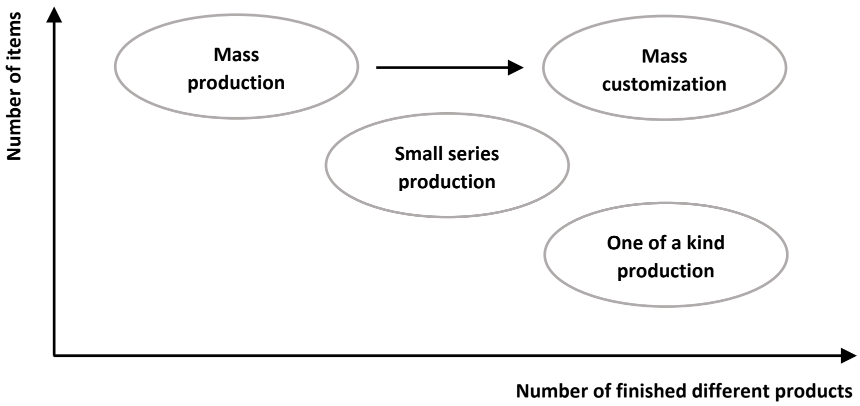

Figure 8. Different types of companies pursuing mass customisation; adapted from (Hvam, Mortensen, & Riis, 2008)

third type of company manufactures customized products in small series. Companies engaged in one-of-a-kind or small series production are challenged with separating the task of specifying customized products from the task of development; they also need to formalize that process while introducing IT support to make the process more effective. The three kinds of companies described by Hvam et al. (2008) have different customer-initiated specification processes. Forza and Salvador (2002) describe a case in which a configuration system was implemented. The authors state that the multiplication of product features induces exponential growth in the volume of information exchanged between the firm's sales organization and its customer base. By using this software system, the information can be stored in organizational memory instead of retained in (or forgotten by) individual employees. However, the introduction of such a system may require significant and potentially painful changes in the way the order acquisition and fulfilment activities are organized. Knowledge-based engineering (KBE) has been recognized as a key enabler for mass customization in managing large ranges of variant designs and responding quickly to customer requirements (Verhagen, Bermell-Garcia, van Dijk, & Curran, 2012). The authors, however, state that KBE requires further research in order, for example, to develop methodological support, improve transparency, and source and reuse knowledge efficiently.

A detailed framework for designing mass customization is presented in Tseng et al. (1996). The authors outline a process that emphasizes the creation of product family architectures in order to conduct family based-design and to integrate departments within the company. Swaminathan (2001) presents a framework for standardization strategies that enables mass customization by emphasizing the standardization of parts, process, product, and procurement; the author also describes the concepts of modular designs and processes, as well as how they interact and can be combined. Considerations that must be taken into account before adopting standardization strategies are presented in Figure 9.

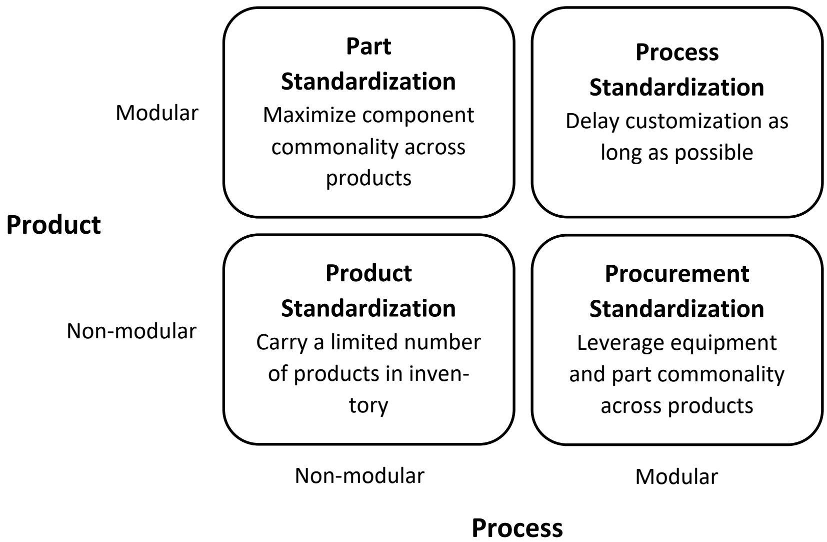

Figure 9. Choosing standardization strategies; adapted from Swaminathan (2001)

3.2.2 Product platforms

A product platform approach can be defined as the development and implementation of technology, components, or subsystems that are shared across multiple products (Marc H. Meyer, Osiyevskyy, Libaers, & van Hugten, 2017). Product platforms have also been shown to prolong the average product lifecycle due to shared components between product variants and the flexibility of introducing new components over time within that same architecture. Product variants derived from a product platform, according to the definition above, have both higher aggregate sales and aggregate gross profit margins over the product lifecycle than products not derived from a platform (Marc H. Meyer et al., 2017). However, this is not the only definition that presented in the literature. Depending on author and business context, for example, several other definitions can be identified. The four most commonly used definitions of product platform are as follows:

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

Based on these definitions, it is clear that a product platform can be described on many levels of concretization or abstraction. This is also reflected among suppliers who describe the company product platform using levels of abstraction, as shown by Högman et al. (2009). The authors investigate whether a product platform strategy is applicable for reuse purposes for a supplier in the aerospace industry. They conclude that a modular product platform is not feasible in such a company because most of the possibility of reuse is found on a higher level of abstraction. Halman et al. (2003), however, state that companies in the manufacturing industry have not been keeping pace with research on product platforms due to a lack of tools. The authors also identified a disjointed view of product platforms in the industry, which is exemplified by the four definitions presented earlier. According to Sawhney (1998), the products within the product platform share a common gene pool. The author also emphasizes some advantages of a product platform approach, including improved design quality, coherence, and option value. Other advantages have been identified as:

- Increased efficiency in developing differentiated products, increased flexibility, and responsiveness of manufacturing processes (Robertson & Ulrich, 1998).
- Reduced cost, reduced time of development, and improved ability to upgrade products (T. W. Simpson, Siddique, & Jiao, 2006).
- Promotion of learning across products, reduced testing, and certification of complex products, such as aircraft engines (Rothwell & Gardiner, 1990).

M. H. Meyer and Lehnerd (1997) have proposed the concept of the Power Tower, which sheds light on the strategy and business opportunity that comes with product platforms. The Power Tower is visualized in Figure 10, which shows how certain product variants are mapped on to specific customer groups and price ranges and how generic building blocks are the base of product platforms.

One risk with using a product platform approach is the trade-off 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 sufficiently pronounced. 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 product platform strategy gained

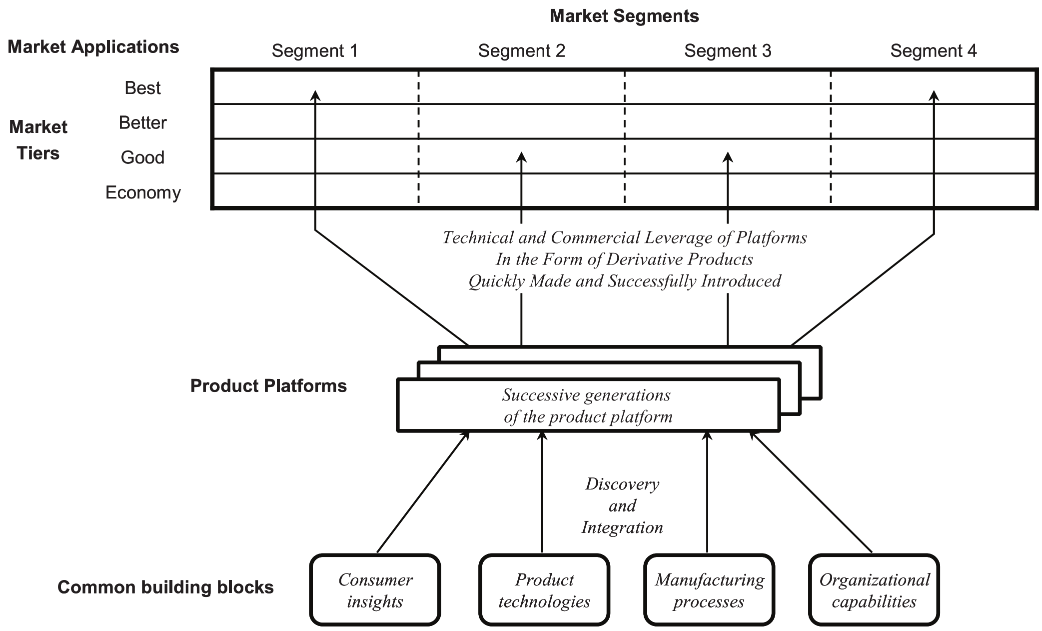

Figure 10. The Power Tower as proposed by Meyer and Lehnerd (1997) and adapted from Högman (2011)

a 5.1% market share, while those that did not saw a 2.2% loss of share. Another trade-off occurs between increased development efforts for the initial product platform and uncertainty as to whether the chosen product platform will allow a sufficient number of derivatives to recover the added expenses (Halman et al., 2003). Cabigiosu et al. (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 result of product architecture choices alone. The study (Cabigiosu et al., (2013) also points to other factors, such as knowledge scope and capabilities, that affect the interfaces more than architectural choices do.

Two approaches for creating product platforms are found in the literature (Timothy W Simpson et al., 2014):

- *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 product platform approach. The module-based product platform can be of two kinds:
    - Integral: Several functions share a physical part.
    - Modular: Each function is delivered by a separate physical part.
- *Scale-based (parametric):* designed so that a number of design variables can be varied and can allow the design to stretch or shrink. This is suitable for optimization due to its continuous nature.

The product architecture is a highly related concept that seems to have preceded the product platform (K. Ulrich, 1995). It is defined by Pirmoradi et al. (2014) as "a concept for describing relations among components and connecting the functions to the components in a product". According to the authors, a product platform architecture then becomes "the logical relations between common and unique elements for enabling highly customized products based on customer preference". According to

Mortensen and Lřkkegaard (2017), product architectures have the following three characteristics: (1) shared core interfaces; (2) core modules/systems that exist in balanced performance steps; and (3) architecture(s) that are prepared for several future development projects. Mortensen and Lřkkegaard (2017) identify ten central principles for the design of product architectures to cover a particular market. These include isolation of low-selling options from the product architecture, decomposition of the product architecture into modules, and identification of stable interfaces, among others.

Modularization is a critical enabler for mass customization and platform thinking that has emerged from the theories of product architectures (Hsuan Mikkola & Skjøtt-Larsen, 2004; Hundal, 2012; Hvam et al., 2008; Swaminathan, 2001). A module can be defined as “a functional building block with specified interfaces, driven by company-specific reasons” (Erixon, 1998) and is a central concept in early publications on product architectures. The reasons Erixon refers to can include aspects usually connected to the voice of the customer, of engineering or of business (Lange & Imsdahl, 2014). Different modularity strategies can be found in the literature. Hvam et al. (2008) outlines “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, such as dimensions, can be adapted in modest ways. “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 upon which components can be mounted. However, these definitions of modularization types intersect with the definitions of product 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 within the literature. Krause et al. (2014) propose a design toolkit as a means of keeping the external variety high while keeping the internal variety low in order to manage design process complexity efficiently. Four principles outline the basis of the proposed toolkit, in which modularization plays an important role: (1) Clear differentiation between standard components and variant components; (2) reduction of the variant components to the carrier of differentiating properties; (3) a one-to-one mapping between differentiating properties and variant components; and (4) a minimal degree of coupling of variant components to other components (Krause et al., (2014). Stjepandić et al. (2015) outline several developments and tool implementations of modularity in the context of concurrent engineering. They state that modularity is an essential property of product design as a multidisciplinary concept. They conclude that a current trend is to combine and integrate different technologies, such as advanced computer-aided design (CAD) systems, product configurators, agent-based systems, and product data management (PDM) systems. Li et al. (2013) present an approach for flexible product platform development using flow analysis, design structure matrix (DSM), multidimensional scaling, and fuzzy clustering, which is used to identify modules. (Park, Shin, Insun, & Hyemi, 2008) present another approach that integrates quality function deployment as a means not only to develop a modular product platform but also to understand the product family. Their method includes categorizing technical requirements and calculating the degree of variety among components.

A successful product platform must be flexible, according to Suh et al. (2007). The authors outline a process for designing a flexible product platform that uses a combination of quantitative analysis and engineering knowledge. Product platform flexibility by abstracting the platform constructs has proven to be one successful way of managing changing requirements. Pakkanen (2016) describes a process that emphasizes the reuse of assets due to limitations in reusing existing designs. Pakkanen's method is directed at companies in the project business that deal with a high level of customization.

3.2.2.1 Models for product platforms

For a product platform to be described in the context of engineering design, a model of the platform is crucial. The term "model" is usually used to describe a simplification of reality. Therefore, a product platform model, as defined in this thesis, is a simplified description of some aspects of a product platform. Some approaches to creating a product 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; to some extent, it can be used to model a product platform (Hvam et al., 2008). The main aim of the PVM is to map the product variants in a company and to couple them with a generic product architecture in order 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 with the "part-of" structure is the "kind-of" structure, which 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, & Haug, 2010). The PVM and CRC cards are said to bridge the gap between domain experts and IT developers.

The research in the field of product platforms has generally adopted a component-based approach, supported by the evolution in product lifecycle management (PLM) and configuration systems; that is, 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 et al. (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 that are uploaded to a PLM system. 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 product platform development methods used in several industrial companies.

Another methodology that can be termed a product platform model is the configurable component (CC) concept (Claesson, Rosvall, & Johannesson, 2005). Instead of modelling the connections between physical parts and modules, as in the PVM, the connections between functional requirements and design solutions are mapped. This modelling 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 product platform in the early phases of development using the CC concept and set-based concurrent engineering (SBCE).

The design (also dependency or decision) structure matrix (DSM) has proven to be a useful tool in many different areas (Malmqvist, 2002), especially when many relations exist between several entities and thus create a complex system. This fact makes the DSM highly suitable for complex product platform modelling that includes many objects and relations of different kinds. The literature describes several cases where the DSM has been used for product modelling (Malmqvist, 2002), product family design (Hölttä-Otto, Chiriac, Lysy, & Suh, 2014), knowledge management (Tang, Zhu, Tang, Xu, & He, 2010), knowledge modelling (Elgh, 2010), process analysis (Helo, 2006), and product configuration (Lee et al., 2010). The DSM is a matrix representation of the connections between a set of elements, such as the components of a product, people, or activities in a process. The off-diagonal relationships describe dependencies, interfaces, interactions, input/output relations, etc. The type of relation in the off-diagonal cells either can be binary or contain multiple values or types. There are two ways of setting up a DSM: “inputs in rows” or “inputs in columns”. Both ways are the transpose of the other, and which is used depends on the tradition and context. The regions above and below the diagonal distinguish the directionality of the represented relationships in the matrix. The ultimate goal of the DSM is to expose the structure of a system’s architecture or design (Browning, 2015). In the case of product DSMs, Malmqvist (2002) lists different features that can be modelled using the DSM.

- Properties: requirements (desired properties) and behaviour (actual properties).
- Functions: what the system should do in terms of transforming an input to an output or creating an effect.
- Subsystems/organs/design parameters/features: the entities of the product that realize the properties and functions. These can be subsystems, parameters/dimensions, functional surfaces, and so on.
- Components: the physical parts of the product as an assembly-oriented decomposition. Examples: for modelling assembly relationships and for analysing change propagation (Clarkson, Simons, & Eckert, 2004).
- Life-cycle systems/processes: the systems that the product interacts with during its life-cycle, including parts manufacturing, assembly, distribution, and so on.
- Product-level alternatives or variants.

Due to the simple format of the DSM, it becomes highly suitable for automatic computational analysis. Malmqvist (2002) continues by summarizing some of the different analysis methods existing in the literature:

- Clustering methods can be utilized to organize the matrix in such a way that elements with strong relationships are gathered in blocks or “chunks” that end up in the matrix, having a block-diagonal form. Clustering can be used to identify modules in a system of components.
- Partitioning methods sort the matrix in such a way that feedback/iterations in a process are minimized. This can be applied to functions or to design parameters.
- Coverage methods are used to analyse the matrix in order to, for example, detect requirements that are not allocated, functions that are not realized, components that lack an assigned function, and so on. Potential also exists for identifying conflict areas, for example, when a particular function is realized by multiple components.
- Index computation methods compute some aggregate value based on the contents of the cells. An example is the strength of a particular module driver in Modular Function Deployment (Ericsson & Erixon, 1999).
- Interaction analysis methods take into account the contents of individual relations and can propose strategies for re-design in order to eliminate or, at least, manage harmful effects.
- Change propagation analysis methods follow the relations from a particular element to its closest related elements and then to other related elements. In this way, the impact of a change proposal can be identified. Aspects such as probability and amount of re-work can be used as factors in the analysis (Clarkson et al., 2004).
- Alignment methods compare the contents of two related matrices, such as a product and the organization structure. Differences are highlighted to identify areas where the organization may encounter difficulties in managing interfaces existing in the product.

3.2.2.2 Technology platforms

Component- and module-based product platforms have been found not to fit every business model (Ulf Högman et al., 2009). However, companies in ETO industries need ways to harvest the fruits of a product platform definition that can give them advantages similar to a component-based product platform. Johannesson (2014) asks whether companies even have a choice about implementing product platforms, since they can exist on all levels, ranging from standard components to knowledge and relationships (Robertson & Ulrich, 1998), which makes 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. (U. Högman, H. Johannesson, 2013) investigate the application of a stage-gate process to aid in managing technology development. Levandowski et al. (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 that there is a need to access knowledge and establish ways of sharing 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 ensure the maturity and product platform compatibility of technologies. Guðlaugsson et al. (2014) describe a tool, which is based on the technology platform concept, called the Conceptual Product Platform. Its aim is to communicate the product portfolio by mapping application requirements through concepts onto the product organs, which are the physical features that realize the functions.

3.3 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 links from customer need to a product's functional attributes. Halbleib (2004) defines requirements as "the agreed-upon facts about what an application or system must accomplish for its users". A requirements statement answers the question of what the system or product should do. However, it should never state *how* the product should fulfil 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 the product must fulfil, 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 documentation, (5) the decision-making process that led to the requirement, and (6) the status of the requirement (Sutinen, Almefelt, & Malmqvist, 2000).

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

- Requirement and concept modelling.
- Incorporating customer needs through quality function deployment (QFD). He links QFD to the hierarchical breakdown of requirements and proposes dividing a master matrix QFD into smaller ones to improve manageability.
- Stakeholder integration. He proposes involving the needs of all stakeholders, not just those of the customer.
- Product and process modelling. He describes how to link product and process development by a model.
- Structuring manufacturing requirements.

3.3.1 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 a 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 requirement changes are 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 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 during a development project due to competitor, product, market, and project evolution.
- Requirements are consciously and unconsciously reprioritized throughout the development project because of knowledge gained, approaching toll gates, and responsible actors, among other things.
- 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 managed formally in some fashion. Changes in requirements also needs 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, enabling changes to be traced 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 are used by Thomke and Reinertsen (1998) as a means of handling changing requirements and variations. The authors propose keeping requirements simultaneously frozen and liquid and make a comparison with the way a newspaper is structured so as to allow different time horizons for completion. Some parts of the newspaper are planned and written weeks in advance, while others are not finalized until the last minutes before printing. This comparison 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, (Land, 1982) proposes trying to gauge the future in order to predict which requirements will be changed. Elgh and Cederfeldt (2008) use a process approach for assessing producibility under the effect of changing requirements.

3.4 Reuse of design knowledge

Many researchers have studied the reuse of design knowledge over the years. Tools and methods have been developed to support the reuse of some aspects of both the design process and the artefact. 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 domain experts' knowledge is gathered and formalized, and knowledge representation, described as a simulation of the relevant knowledge. They discuss these aspects in light of 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, to create an informal and a 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 constructively criticize MOKA, proposing that a software tool would be helpful to create an informal model for reasons such as checking for validity. They propose that MOKA should support an iterative process in developing the formal model, while they suggest a more concurrent process for creating the code in the final system. This suggestion is an effort to mitigate the risk of programming and the system integration problems that often occur if the coding is left for the very last stages. Preston et al. (2005) further state that creating the system at the same time as collecting the knowledge could create a feedback loop that enables the experts to check whether the software model aligns with their own understanding of the area.

Design rationale is an important aspect of design reuse research. Regli, Hu, Atwood, and Sun (2000) define design rationale as the explanation of why an artefact or some part of an artefact is designed in the way that it is. A complete design rationale includes all background knowledge, reasoning, trade-offs, and decisions taken throughout the design process. 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 (Elgh & Poorkiany, 2012). 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, is 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, which is carried out by filtering product candidates based on their alignment with desired characteristics and their performance efficiency.

There is a pressing need in the industry for tools to practice reuse of knowledge. This is, however, a highly complex area, since knowledge can appear in different kinds, formats, or locations, such as in PDM systems or people. This complexity has hindered research to find a generic solution for knowledge reuse that integrates systems used by a company and is sufficiently user-friendly 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) consider knowledge to be actionable information, and they raise the problem that many previous design knowledge reuse systems focus exclusively on geometrical data, which is often not applicable in the early stages. They propose that future reuse models should 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 accessible, several additional factors must be met if it is to be reused, which are 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. They emphasize 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) highlight a lack of user-friendly classification and structuring of engineering knowledge in today's CAD software, which makes the retrieval of existing templates (used for knowledge capture) a challenge and often results in the total recreation of designs. Their paper proposes a structured way of reusing feature templates, based on the generic product structure of a product, to make up for the inability to do so in a systematic way.

3.5 Technology development

The way that research (TD) and development (PD) are managed has evolved in recent years due to changes in the structure and demands of the economy. In the early days, research and design (R&D) was seen as an overhead cost focused on pushing technology towards the market. In the 1980s, the focus shifted towards developing a complete product concept, consisting of service, distribution, and product platforms. Most recently, it has become increasingly 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 artefacts in order to enable PD (U. Högman, 2011). Deliverables can also appear in the form of demonstrated feasibility (Nobelius, 2002) or a technological platform (Cooper, 2006). Cooper (2006) further notes that TD is important for a company's long-term growth but often is assigned low priority and represents a small portion of the company's total effort. 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" (D. Corin Stig, 2015). Technology has also 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 in line with its overall strategy (Clausing, 1996). Figure 11 visualizes a PD project (with a certain time frame and targeted customer) that tries to "fish out", or identify and select, a relatively pre-developed technology present in the company that can form the basis of a new product. However, what is the need to separate the two development processes of TD and PD? 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, 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, 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).

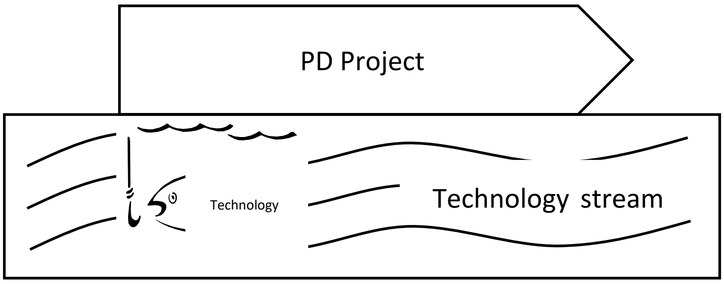

Figure 11. The technology stream; adapted from (Clausing, 1996)

Even though a separate process for TD has been proposed by several authors, challenges 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' advocation of the incorporation and development of new technologies.
- Insufficient understanding of TD as primarily a process of understanding.

The splitting of TD and PD creates interfaces between the two. Lakemond, Magnusson, Johansson, and Säfsten, (2013) offer a categorization of these interfaces as follows: (1) *contextual interfaces* are interfaces at the edges of a new product development (NPD) project, 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, the degree of change in the product, the complexity and degree of change in the production process, the organizational separation of TD and PD, and the 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 when new technology is introduced into PD.
- Transfer scope. This refers to decisions regarding what to transfer; it deals with concept, test results, and recommendations.
- Transfer management. This concerns 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. Cohen, Keller, and Streeter (1979) likewise 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 during this transfer to offer support with tacit knowledge (D. Corin Stig, U. Högman, D. 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 contribute to transferring knowledge between TD and PD. Ravn, Gudlaugsson, and Mortensen (2015) propose 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. Their model can be seen as a description that connects a core technology with, for example, those people who possess tacit knowledge about the technology, a prototype, trade-off curves, technology readiness levels (TRLs), and technology implementation. The TRL scale is used to judge a technology's level of maturity (Mankins, 1995).

3.6 Scientific motivation and knowledge gap

The concept "product platform" seems to have been coined sometime in the eighties. However, platform thinking (Sawhney, 1998), including reuse, component standardization, and commonality (Guerrero, 1985), was probably practiced by design engineers long before its obvious business possibilities were discovered and systematized. (Sawhney, 1998) and (McGrath, 1995) identify the business opportunities of platform thinking and include the whole firm in the mind-set. These publications, however, lack the detailed support for companies to gain the direct benefits of platform thinking. The response from design departments in later publications focuses on modularization and standardization of components and subsystems in products (Erixon, 1998; Hvam et al., 2006). Researchers soon discovered that certain types of businesses, like aerospace supplier companies, have difficulties in applying such approaches (Claesson et al., 2005). The product platform concept has further developed with this in mind, and reuse possibilities have been sought in other areas than physical components (H Johannesson, 2014). Today, platform-based approaches that focus on sharing and reusing are used on several levels in companies from the early phases of design (C. Levandowski et al., 2014), as an architecture tool, to business tools used for branding in the consumer market (Zhang, 2015). However, much regarding product platforms still needs to be researched, including requirements management in the presence of uncertainty, the integration of new technology in product platforms, and the product platform strategy in regard to the business model.

This literature overview has shown that customization has been identified as important to fulfilling customer needs more completely. A product platform approach has been shown to be an enabler for efficient customization, reuse, production standardization, novel technology integration, and changing requirements. There is, however, a lack of research detailing how a product platform definition can be utilized by companies with an ETO customization strategy to gain the benefits of platform thinking. Company and customer integration and collaboration have previously been in focus in research (Tuli & Shankar, 2015), often involving a single business interface (Siddique & Boddu, 2004). However, this focus is often a simplification of reality since ETO companies are frequently part of a supply chain in which other suppliers and an OEM act between them and the end customer. This complexity introduces several interfaces and stakeholder interests that the company and thus the product platform must manage. Holistic research in this area, which takes all or several of these perspectives into consideration, remains scarce and significantly complicates the use of product platforms. In previous research focusing on product platforms (Zhang, 2015) and design repositories (Lyu, Chu, & Xue, 2017), an emphasis has been placed on the artefact, excluding the design knowledge required for the process that leads up to the artefact. This omission 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 (Hvam et al., 2006). However, there is little research that describes how knowledge of different kinds and in different formats can be integrated into a product platform description to achieve advantages like those found in component-based product platforms.

4

THE DESIGN PLATFORM APPROACH

This chapter outlines the main result of this thesis, as well as the principles and overview of the generic Design Platform Approach (DPA) that has been developed and applied throughout the appended papers. The DPA, along with the terms used to describe it, has evolved and changed throughout the appended papers. Thus, the approach presented in this chapter is a synthesis of the appended papers and the final result.

Manufacturing equipment, like machines and tools, standardized manufacturing processes, and the people working in them are a necessity in manufacturing companies. In an efficient production system, the raw material flows are streamlined, and all involved staff know their function. It is common practice to keep track of existing machines, their capabilities, what activities they can perform, their service interval, and the cost of using them. The production system is a crucial asset in which investments are made. It can be referred to as a production platform that enables the production of certain product variants. Development resources can be thought of and similarly described as important assets. They are investments made in the company that can leverage efficiency over ranges of product variants. However, the capabilities of the development team, the existing development solutions and tools are less often described in a coherent way than that of a production systems.

An important issue in the industry is the loss of knowledge created by focusing on specific product and project 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 is described in a way that enables easy adaptation during customization and that can be generalized for use in future projects. If this approach is successful, it will expand the company's overall knowledge.

The approach must also be able to support different knowledge domains, spanning various disciplines and world locations. The product platform approach, as will be outlined, aims to offer a coherent model that brings together elements of product platform theory and other resources residing in companies. The approach thus must be able to manage the diversity of knowledge that is continuously created and described in companies. Further, it must prepare the knowledge in a way that makes it adaptable and enables efficient customization at a later stage. In order for companies to build an ability to create such product platforms, it is important to acknowledge that changes will occur throughout the development.

The DPA is inspired by Robertson's (1998) product platform definition which indicates that a product platform is constituted by assets residing in the company. What needs to be added to this definition is a coherent model and a structured way of identifying, developing, and managing DPA that makes use of these assets in the right way. Without a coherent product platform model and method, and with only a vague definition of a product platform, the development will not be supported. This thesis uses the word *approach* to capture both the method and model needed to successfully create and execute the DPA. Thus, the DP refers to the information model that is the blueprint and information carrier, whilst the DPA is the overarching concept that refers to both the DP model and the methodology of creating and executing it.

4.1 The Design Platform model

A specific Design Platform (DP) model is composed of different objects related to processes, synthesis resources, product constructs, assessments resources, solutions, and projects, as well as the relations between those objects. These domains are the result of appended papers in this thesis. The domains should support pre-existing resources in a company (commonly projects and solution resources), as well as describe generic processes, generic product concepts, tools, and future support models. Figure 12 depicts an example of the information model on a conceptual level.

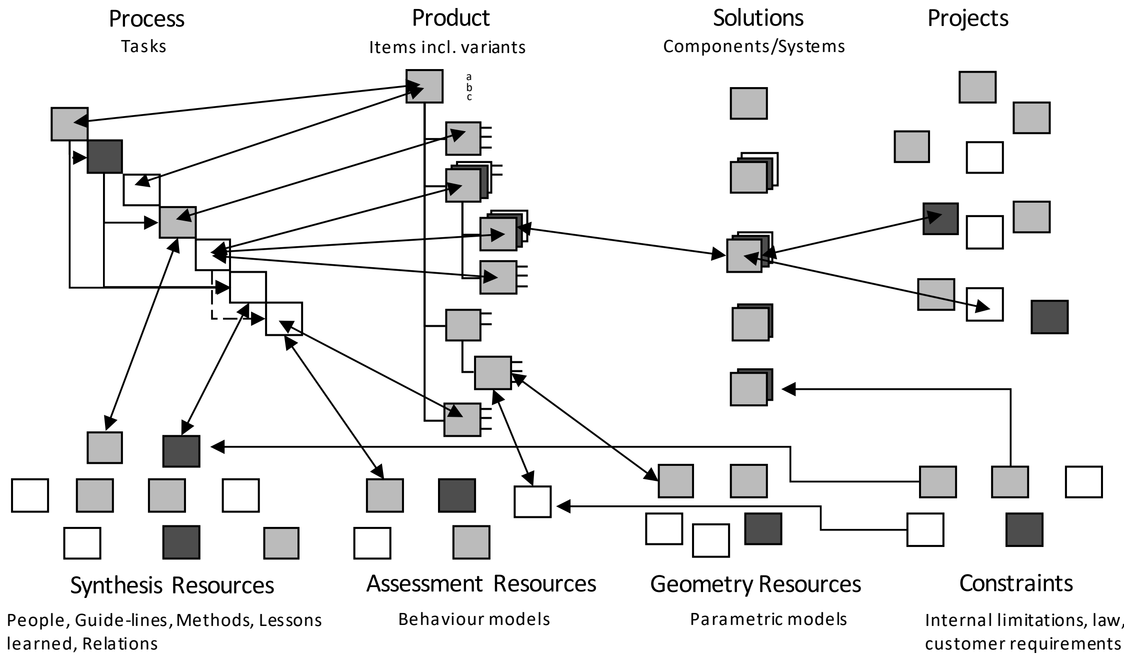

Figure 12. The conceptual DP model and its constituents

The figure illustrates how these different objects are linked to a generic product concept and process. Though a company business model might not allow for a strictly component-based product platform approach, elements of knowledge regarding the product concept exist, which can be a mixture of methods, tools, standard components, etc., and thus constitute the company's competitive edge. Some parts of the product might have the potential to be standardized, while others require that specific processes be completed during the design process, resulting in unique solutions. Therefore, both a product and a process domain are included in the DPA and linked to each other. For example, a component type that often requires a unique solution can be supported by and linked to a formalized process model that is, in turn, linked to design guidelines (synthesis resources); these guidelines support how specific parameters and calculation models (assessment resources) are determined, which supports the assessment of parameters or complete designs. These process and product concept models are linked to relevant constraints that should be considered during the development process. Modelling the processes, product concepts, and resources essentially unifies knowledge and enables reuse, which supports designers by creating a toolbox of designs and means for design. Figure 13 shows a generic UML diagram as a formal representation of Figure 12. The information model is ultimately part of a PLM environment with an application landscape that has clear functions. The PLM system will have a core role with the objects containing metadata that point to the location of specific files containing detailed information on the object. The detailed information can also be encapsulated in the object itself, depending on the specific implementation of the DP.

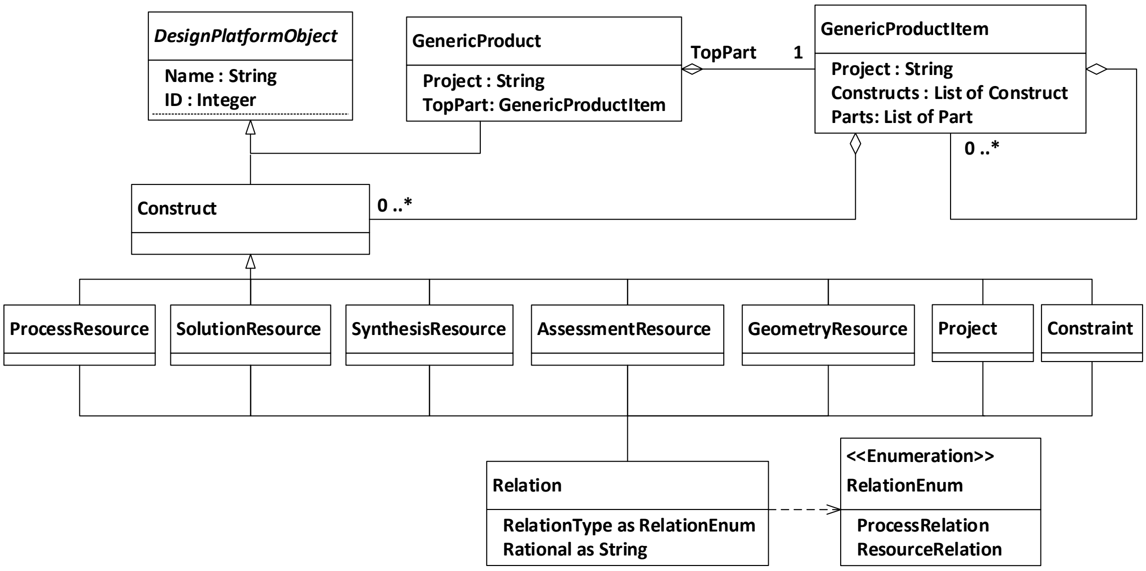

Figure 13. The main classes and relationships of the DP model

The generic representation of the product is represented by the class GenericProduct. Each object that is instantiated from this class becomes a variant and will be unique to a specific project; it consists of its generic item structure, cardinality, and attributes.

The GenericProduct class has the property TopPart, which can represent an assembly or component and is instantiated from the class GenericProductItem (GPI). The GPI has a recursive relationship, allowing it to create a generic part structure. The GPI class can represent generic components and subassemblies to be included in a finished product; these components and assemblies should be seen as containers for intended or needed resources to support their realisation and not as finished solutions themselves. The GPI class has a list of the Construct class. The Construct is the general type of class used to model and point to resources of different kinds. It is a superclass containing base attributes, which are described in Table 3.

Table 3. Attributes of the Construct class

Attribute	Explanation	Type
Description	Description, which is a general explanation of what the Construct is and does	String
Context	Information about when and in what context the construct is valid	String
Technology readiness level (TRL)	TRL denotes the maturity of the construct. This number ranges from 1–9 and originates from the aerospace industry (Mankins, 1995). 1 implies that basic principles have been observed and reported, and 9 refers to a fully developed and validated construct. The TRL is used to judge which constructs need further development and which constructs can be used as they are.	Integer
Management	Management and responsibility information including relevant people, dates, and versions.	String
Raw data	Information regarding the raw data and their location for the detailed information that makes up the construct.	String

The different classes that inherit *Construct* include *ProcessResource, SolutionResource, SynthesisResource, AssessmentResource, GeometryResource, Project,* and *Constraint*. Their attributes are differentiated depending on the needs of the specific company in which the DP model is applied. Objects instantiated from *SolutionResource* relate to finished designs that have formally been designed and have thus been created with some boundaries, as defined by *Constraint* objects. *SynthesisResource* has explicit *Constraints* that are related to the method, guideline, or tool that the object represents. Objects of the type *AssessmentResource* support the evaluation of product variants and embody mathematical models representing behaviour and other properties. They can result in implicit constraints, the implications of which are made visible through, for example, simulation models. However, in general, *Constraint* objects are related to *GPI* or *ProcessResources* that are considered for a specific component, sub-system, or process step. A *ProcessResource* can take the form of tasks or execution orders of activities that are required or intended to support some part of the design process. Objects of the type *GeometryResource* are commonly parametric CAD models that can act as a baseline for new designs. Their parametric nature can, in turn, represent the physical design space of the model. To each of these classes, the *Relation* class can be associated. A *Relation* refers to an object that connects two other objects and holds information about the connection's purpose and nature. The *Relation* class can constitute different types and essentially relates the different construct types to each other. The relation can be of the following types:

- *Process relationships* states the sequence order of objects. The term refers foremost to the order in which a set of activities or process steps should be executed.
- *Resource relationships* states if a certain object is to be used as a resource for another object. For example, an object describing a parametric CAD model can be a resource for an object describing an activity concerning the design of a specific component.
- *Hierarchy relationships* are derived from the tree structure and are not an explicit relationship type in the class diagram (Figure 13) since they exist implicitly in the aggregation relationship of the GPI class.

The DPA utilisation is divided into two stages: expansion and use. The expansion stage refers to when the model is established or modified; here, assets are identified and/or created, and relevant information is linked to the asset objects. The use stage refers to when the information in the model is used and DP instances are created. A DP instances is a description of the finished product design, the assets used during the design process, and the relation between them. The two stages are more elaborately outlined in the following section.

4.2 The Design Platform Approach expansion

The activities that support the creation, expansion and maintenance of the DP model are shown in Figure 14. The first step focuses on organizing the management of the DPA, identifying areas of responsibilities and associated personnel, and setting up a structured database environment. Asset owners need to be appointed who are responsible for a specific set of assets relevant to their expertise. Similarly, structural owners must be appointed who work on a higher level, relating assets to each other. The second step is to formalize and generalize the product concept by identifying generic products, their GPIs, and structure. These are the common bases to which different constructs will be linked. These items might not correspond to a pre-existing design; however, they must be included in the finished design. The third step is to identify the boundaries of the GPIs in terms of what and how the product interfaces the customer. Depending on a company's business model, a product is developed directly for the end user or to be integrated into the customer's product (e.g. when the ETO company is a supplier for an OEM). The fourth step then becomes to isolate the subsystems that only interface the system which is developed within the company. These subsystems can be standardized to a higher degree into modules, which can be reused in several product variants. As these modules are kept isolated, they can be further developed in technology development projects that are not directly focused on customer orders. At this step, if applicable, the subsystems that interface the customer product are identified.

The subsystems, which will most likely need to be designed for each new customer order and are the focus for the continuation of the steps defining the DPA. In the fifth step, generalized trade-off curves associated with the GPIs need to be assessed. These trade-offs concern properties in existing designs and concepts that are related to each other (e.g. weight vs tensile strength) that are known from previous experience or simulations. Continuous or discrete design spaces are defined or identified in relation to the different GPIs in the generic products (6). By identifying and modelling the feasible parameter space of a design, variations of such designs can be reused and cover a larger application area. To support the modelling of these design spaces, geometric models are developed by identifying generic characteristics of existing geometry and trends in geometry adaptation (7). Processes, best practice methods, guidelines, and tools for their completion are retraced, modelled, and published (8,9). These are described using standard classes. Competence teams are established to build and preserve knowledge, skills, and abilities (10). The GPIs are then linked to the identified descriptions, models, and other engineering assets needed for their realization (11). Improvements through the experience and knowledge gained from PD are continually assessed and applied when a sufficient level of maturity

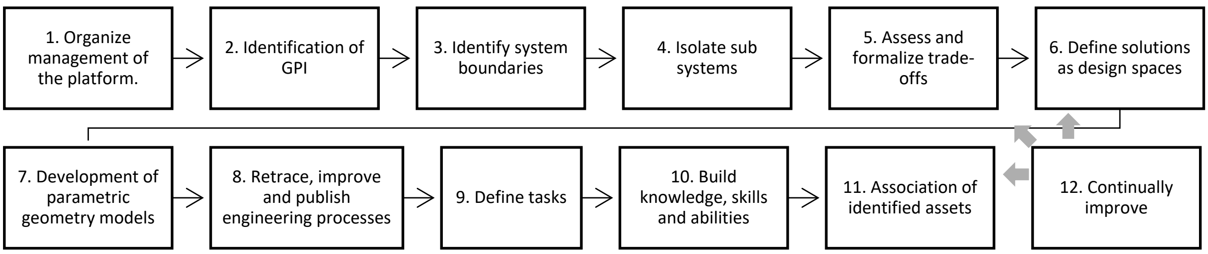

Figure 14. The steps supporting DP model expansion

is reached (12). These improvements can be achieved using lessons learned, which helps users to identify issues in their current way of working or in the assets they have used. Action plans can then be created to address the identified issues and to implement improvements. New formats can also be introduced that capture the specific types of information needed in the PD. It will be the responsibility of the transdisciplinary management team to maintain the DP model over time and continually evaluate the included assets in order to guarantee its usefulness. It is important to manage DP model scalability to prevent the amount of information from growing out of hand. The amount of information targeting a specific use group must be well planned and managed in order to support the users.

4.3 The Design Platform Approach use

A structured use of the DP model requires that some knowledge has already been modelled. Using the DP model focuses on the creation of specific instances that are developed for specific customers. The main steps are illustrated in Figure 15. When a quotation process is initiated or an order project begins, the model is used to obtain a first glimpse of what information exists and what does not. Early in a project, the DP model can be used to find generic process support to guide the design process. Thereafter, the first step is to search among generic products in order to determine if a similar variant has been created in the past or is currently in production. If such a structure or component exists and fits with the specification, the associated variants of that generic product can be searched and matched to the customer's requirements (constraints). However, in most cases in ETO companies, there will be no complete finished variant design. This is common if components, system interfaces with the customer product, or if subjective requirements are used. In these cases, the DP model provides a formalized process that creates associations to other resources describing, for example, a task, a parametric CAD model, or calculation spreadsheets that can aid in designing the desired component or sub-system. When the search has been completed, the model is ready to be instantiated. Upon instantiation, a variant is created that represents an unfinished DP model instance embodying a design template. It consists of resources to support its realisation, which are used in the following design process. In special cases, no product structure will be identified when modelling the DP model, in which case the structure is implicit and is the result of the design process. This is common for manufacturers of highly customized manufacturing equipment where no product concept exists beyond the component and sub-system level. In this case, tasks and guidelines are suitable for governing the work of specifying the final product structure. During the PD, the variant is continually updated, and the different GPI levels are associated with the created designs and resources used in the process. In this way, the DP model evolves as new knowledge is described, added, and associated. When a project is finished, one of the outcomes will be a DP model variant that will act partially as the design rationale. In upcoming projects, these variants will be found during the initial search and include an overview of of the design and its rationale.

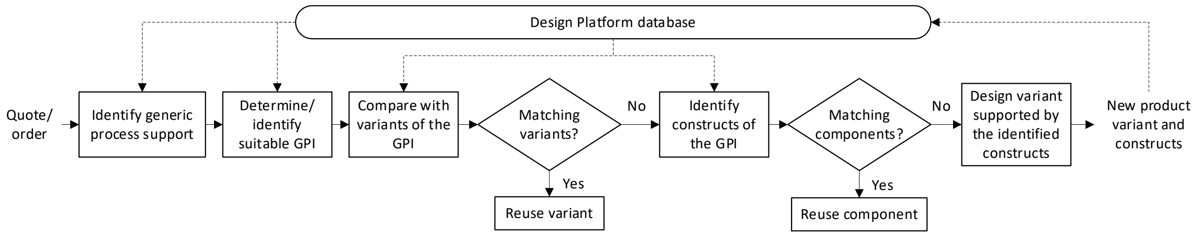

Figure 15. Principles of DP model use

This chapter has outlined the generic DPA, which consists of an information model with associated classes and attributes. Further, it has provided guidelines for how to create and execute the DPA. The detailed examples and case applications of the DPA are found in the summary of papers in the following chapter.

5

SUMMARY OF PAPERS

This chapter presents the main results of the appended papers, which are used to underpin the answers to the research questions. The chapter starts by presenting an overview of how each paper is connected and contributes to the overall result. It continues by summarizing the highlights of each paper.

The main focus of the research was to develop a product platform approach for ETO companies in order to increase their efficiency, which resulted in the DPA as described in the previous chapter. Another goal of the product platform approach is to increase company responsiveness to the requirements dynamics that occur throughout the PD process. This approach has been developed in steps, starting with investigating the needs, possibilities, and environment of such a product platform, passing through the development of a conceptual model, and concluding with an applied, formalized, and generalized model and method. The terminology and names used for different concepts have also evolved between the papers, as will be clarified in the subsequent summarizations. Like the theoretical approach, the realization of the product platform approach in a computer support application has evolved, been developed, and been presented concurrently in the papers. The order of the papers corresponds to growing levels of maturity, formalization, and validation of this approach and tool. Therefore, it is important to note that each section on a different paper in this chapter refers to the model and approach presented in the corresponding paper rather than to the final result presented in Chapter 4. Figure 16 visualizes the connection of the results, the evolution between the papers, and the level of model formalization and refinement of the results obtained. Next follows a short summary of each paper.

Paper A presents prerequisites and challenges at four companies. The paper categorizes and connects the companies to the level of abstraction at which their product platforms are defined. It finishes by proposing a product platform idea definition that could support these types of companies.

Paper B elaborates on how a company can describe the outcome of TD and PD on a conceptual level by introducing descriptions at different levels of concretization. This is discussed in connection with a conceptual product platform model as a means to increase the reuse of design knowledge, which builds on the product platform idea from paper A.

Paper C continues to build on the concept presented in Paper B 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 product platform approach is realized in an early prototype software application called the Design Platform Manager (DPM).

Paper D presents further empirical data from the same companies investigated in Paper A. A generalized product platform approach, the DPA, is presented and used to map the four companies. Furthermore, the DPA is supported by computer applications and evaluated in three cases. A discussion of suitable guidelines to feed the DP model from TD is presented.

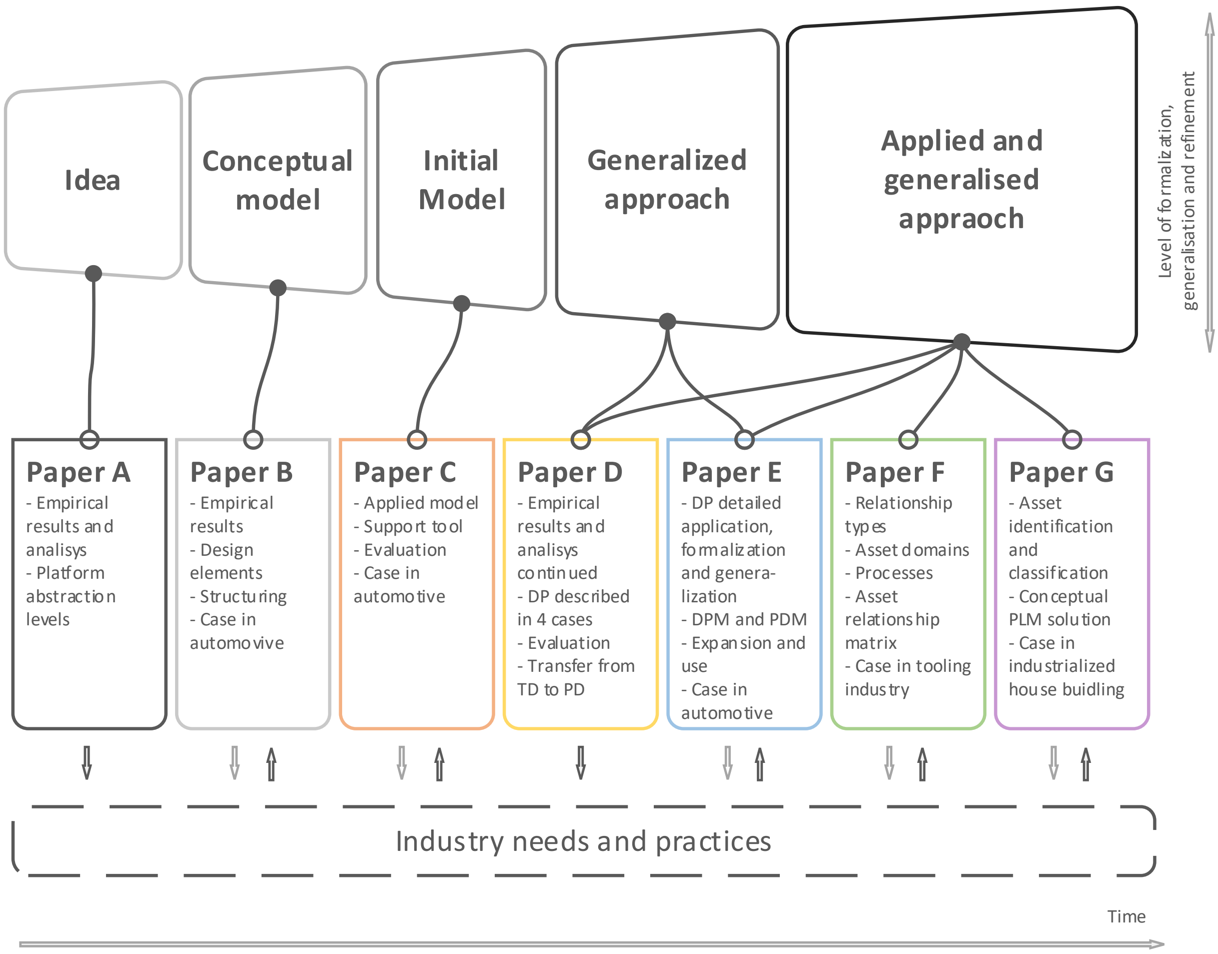

Figure 16. A mapping of paper contribution and progression to the overall DPA

Paper E presents the detailed application of the PD approach in one of the cases presented in Paper D. A more developed prototype system with DP modelling possibilities and PDM integration is described. A process for both the expansion and use phases is proposed.

Paper F describes the expansion of the DP model to include a relationship class that can be of three types. The paper further describes that relationships can be modelled in different asset domains using an asset relationship matrix. The DPA is applied in a case in the high-pressure die-casting tooling industry.

Paper G presents empirical data from the industrial house-building (IHB) industry. Assets are identified and categorized. A conceptual PLM system solution is presented and discussed to support the introduction of the DPA in the company. Changes to the DP model to fit in the IHB industry are discussed.

5.1 Paper A

Companies can gain a competitive edge by continuously and systematically investing in TD in strategic areas. Paper A argues that this is a challenge for suppliers of customized systems due to the large differences 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 preferences. 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 participating 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.

5.1.1 Product platform challenges and prerequisites in ETO industries

The focus in the interviews presented in Paper A was the company product platforms or elements that could be described in a product platform context. One company used the term "product platform" internally. For the others, the product 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 into 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. All the companies had a product in mind when starting a TD project, but they expected different TD deliverables. 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 the companies in the study were faced with the challenge that requirements fluctuate. However, their views of requirements differed; 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 requirement changes.

The interviews made clear that the companies aimed to work in a platform-based fashion and had well-defined and similar processes for PD. The product platform constructs could be defined on different levels of abstraction. In Table 4, the companies are roughly situated according to the abstraction level of their product platforms. Following (Hansen, 2003), the types of variant specifications are also coupled with the abstraction level of the product platform. It should be noted that each company had characteristics that could be coupled with several variant specification types; in the table, however, they are placed on the level that fit them most closely. Company factors that increased the need for a higher product 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 product platform, the more engineering would have to be carried out to deliver a product. Moreover, the higher the abstraction level of the product platform, the more the companies tended to describe it as an explanatory model rather than one composed of physical components.

Table 4. Product platform characteristics categorized by different levels of abstraction

Level of abstraction ->	**C4**	**- Engineer-to-order**
		- The product platform is made out of methods and knowledge.
	C1	- Modify-to-order
		- The product platform is made out of modules and sets of rules.
	C2 and C3	- Configure-to-order
		- The product platform strategy relies on standard modules and parts.
		- Select variant
		- Product variants are pre-defined. No possibility for adaptation.

5.1.2 A product platform idea for suppliers of customized systems

Paper A ends by mapping different engineering assets as a product platform idea for ETO companies, as shown in Figure 17. This model idea is first mentioned in (Elgh, 2013); it differs from the component-based product platform models in that it contains an array of models, methods, and tools that might be possible to integrate into a product platform model. Figure 17 describes how TD is separated from PD and how the deliverables from TD and PD build up a product platform in which the technology is effectively described and can be adapted to fit different PD projects. The PD personnel can then use the product platform for creating customized variants. The product platform also contains the maintained knowledge that is continuously developed with the company's projects. The efficiency of the proposed product platform relates to its adaptability, its ability to handle fluctuating requirements, and how effectively variants can be created from it.

5.2 Paper B

Paper B elaborates on how a company can describe the outcome of TD and PD on a conceptual level by introducing descriptions at different levels of concretization. This is discussed in connection with a product platform approach as a means of increasing the reuse of design knowledge. The conceptual product platform approach in this paper was inspired by the product platform idea presented in Paper A and information gathered from the investigated companies.

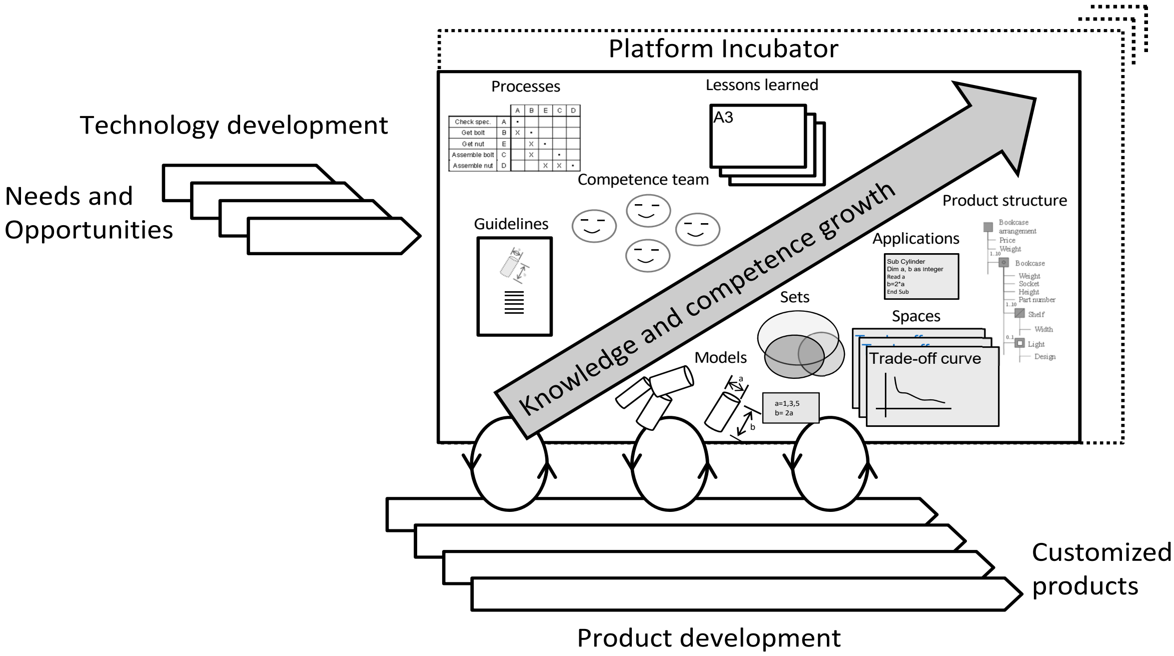

Figure 17. The initial platform model idea

5.2.1 A conceptual product platform description

The aim of the conceptual product platform approach 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 concretized to a high level, which also usually means the design space is narrowed. 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; that is, the possibilities to reuse the instance will be restricted. What is needed is a model which is formed in such a way that the benefits of platform thinking can be gained.

If design knowledge is captured, structured, saved, and is retrievable, it can be reused in future development projects as part of the product platform definition. By being proactive and exploring a design during TD and PD, companies could save and reuse this knowledge through the addition of descriptions at different levels of concretization. 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 product platform, lead to a platform that is more than component- and module-based. The product platform is now composed not only of the product's physical elements but also of elements that support the designing of the product. Therefore, the name "design platform" is more suitable than "product platform" since it refers to the activity as well as the object.

5.2.2 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 resemble a structure with subclasses, metadata model classes, and DE classes (Figure 18). The generic product structure is based on the of the PVM tool (Hvam et al., 2008) but is altered here to fit the ETO context 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 with 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; its 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 formulae, tables, or in text form. They can describe design parameters and how these 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.

5.2.3 Design element structuring and identification

The dependency structure matrix (DSM) tool was used to identify connections between parameters, design variables, and items (Figure 19). Item refers either to a component or a subsystem. By combining the DSMs, domain mapping matrices (DMMs) were created to relate entities between domains. This was done in order to map the given parameters to items and thus to cluster the parameters that

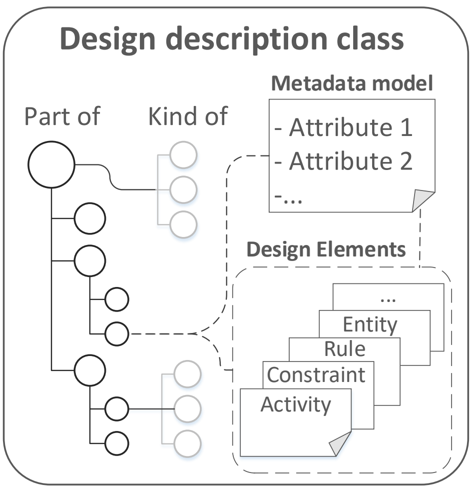

Figure 18. An initial platform concept

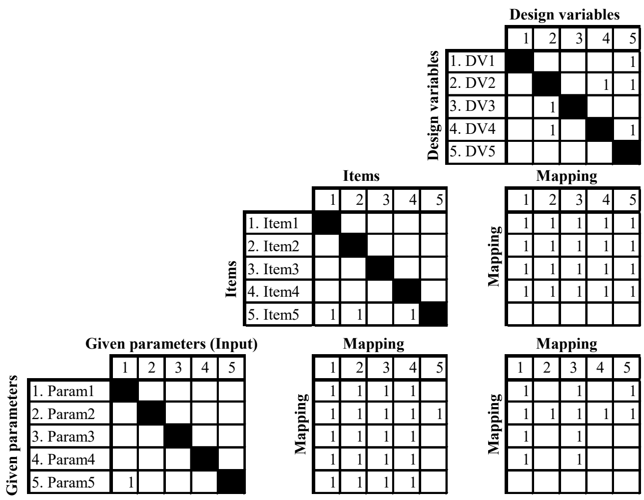

Figure 19. Example of DSM mapping to identify DEs

will be the input for designing an item or configuring a system. 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 exist 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 in MS Excel spreadsheet templates. 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 provide an overview of the DEs present.

5.3 Paper C

This paper builds on the concept presented in Paper B 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 DPA is realized in a prototype software application called the Design Platform Manager (DPM). The DPA and software tool aid the case company in describing not only finished designs and how they relate to a generic structure but also elements like methods, task descriptions, constraints, and design rules. The DP 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 results in terms of increasing the level of reuse, speed, and accuracy during the quotation period and in supporting the design engineer.

5.3.1 A product platform approach to support suppliers of customized systems

The product platform approach is defined by the following characteristics in Paper C:

- Descriptions of product instances and their interrelation with a generic description, which means that the product 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 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 form 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 product 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 its construction 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 model can then contain state-of-the-art descriptions—the latest versions that have been proven to work by experience or evaluation. The model is expanded by setting it up or modifying it, by modelling GPIs, and by creating descriptions that are linked to relevant information. The use phase (which in Paper C is referred to as the “execute” phase) involves instantiations of the generic product items using the information in the model to become product variants.

5.3.2 A DPA support tool prototype

In order to support the use and expansion of the DP model, the conceptual application first presented in Paper B is further developed. The application features a functionality that allows the user to create GPIs and to couple DEs with the different GPI levels 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 20. The application uses the following features:

- Spreadsheet templates for creating DEs.
- XML for saving GPIs, 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 GPIs with DEs.
  - Instantiate GPIs 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.

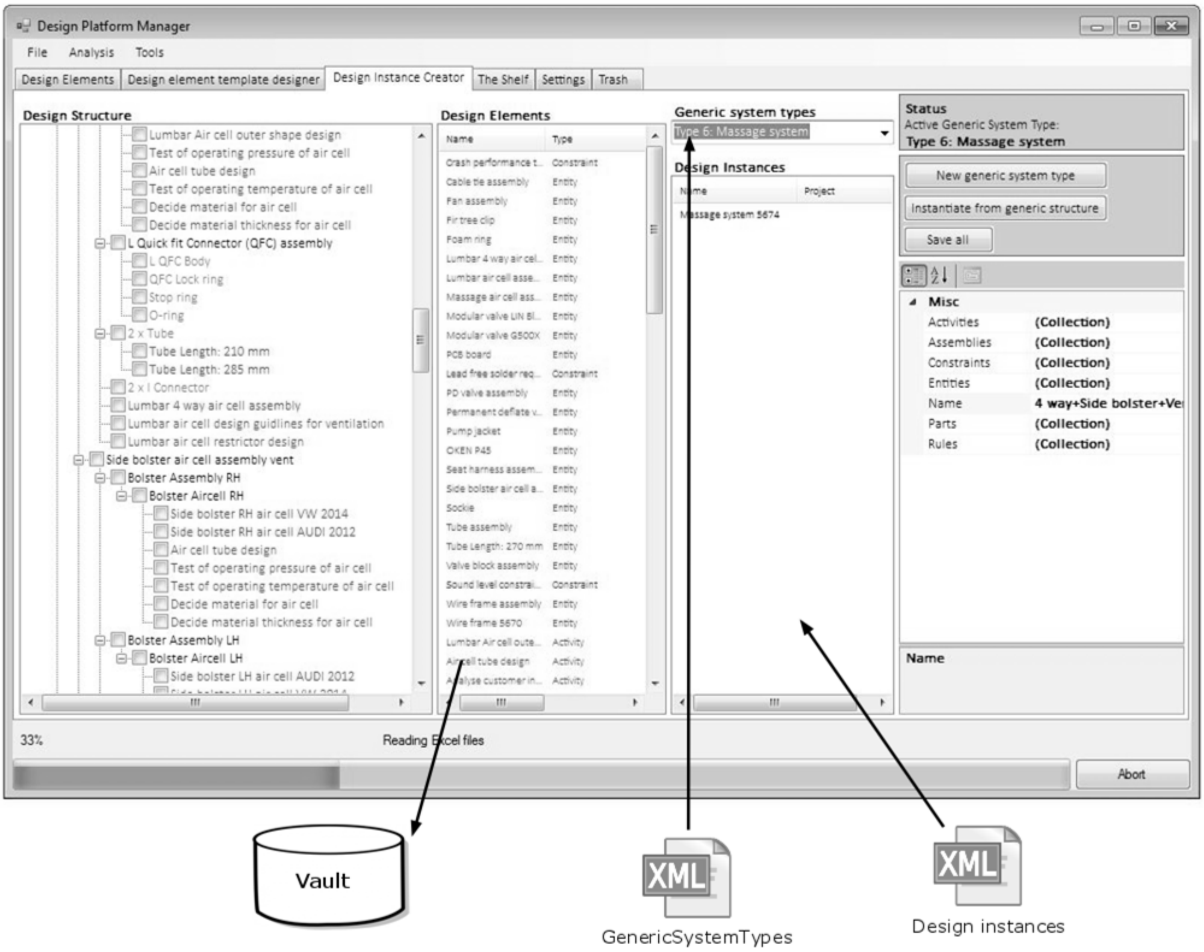

Figure 20. Screenshot of the application user interface with principal explanations

5.3.3 Evaluation

The DPM was presented to, demonstrated to, and discussed with potential users at the case company. The three respondents represented three potential users of the tool at different hierarchical levels in the organization. The employed questionnaire mainly asked respondents to assess how well the presented method and software support met the success criteria identified jointly with the company at the beginning of the project. The questionnaires, which were filled in individually, are summarized in Table 5. The results of the evaluation, along with comments from the engineers who participated in populating the DPM, formed the basis for further development and refinement.

The outcome of the evaluation showed good results and fulfilment 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.
- Support in assuring that requirements are fulfilled.
- Increase the possibility of reusing company assets.

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

5.4 Paper D

This article presents the results of an empirical study regarding the use of product platforms in four companies. Furthermore, it outlines a detailed description of the DPA, the mapping and application of the DP model in four cases, and a two-step evaluation. This summary will focus on the empirical study, the case of applications, and the evaluation.

The focus of the initial study was on descriptions and models that can support the management of changing requirements. The companies itemized reusing assets, using trade-off curves, making solutions adaptable, and gaining the means to estimate cost as important to increasing their ability to continuously manage changing and conflicting requirements when developing customized products. Descriptions that support the work of customizing adaptive technology solutions include cost and decision support, which should be quick and easy to understand. Many descriptions are created, but they are not always saved in a useful format or structured to enable access and reuse.

The paper concludes that engineering assets already existed in these companies (e.g. tools and designs) and that they, if integrated, improved, and combined with additional constructs, could form a coherent product platform description to be used in PD. There are many ways to increase the ability to manage fluctuating requirements, as well as numerous and diverse descriptions that are used and could be used for this purpose. Yet in order to work in a platform-based way, these ways of working and describing solutions need to be supported.

Table 5. Summarization of evaluation results

	Engineering Manager	Design Engineer	Quality Responsible/Project Leader
1. What possible use do you see for a method and supporting software system like this at the company			
	In a pre-study process. Even in a quotation process.	Help to formalize and store existing data and to structure (systems) to make it easier for designers to find relevant information that can be used for new designs.	Good possibilities, but it can be difficult and take time to implement.
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.	50%, especially for new persons involved that do not know the current concepts and techniques.
b. Support in assuring that requirements are fulfilled?	Requirements are always present. It can possibly contribute to our response to the costumer's requirements as a checkpoint/control station/gate.	Yes. Review the customer requirements and use the software and compare these with existing designs/products.	75%, if we don't need to create new development but calculate with the current figures and test reports. Use of available trade-off curves.
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.	Easier and more rapid access to existing solutions and design trade-off curves.
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 the like and help the designer decide whether the components can be reused.	More rapid way to find out what is included in existing designs.
ii. Knowledge?	Some complement as to how we currently work; vizualization of knowledge.	Yes. Allows easy access to stored knowledge.	Better way of finding the technical characteristics of an old product.
3. What do you think the drawbacks of the proposed solution are?			
	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 correct way.
4. What needs to be improved to ensure success criteria fulfilment and usefulness?			
	Visualization 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 ingoing components.	Integration with CAD; visualize CAD models.	Access to CAD models or "black box" that describes the design solution.

5.4.1 The DP model described in four companies

The DP model was initially developed in cooperation with four companies, which are named C1, C2, C3 and C4 in this paper. This section presents how the DP model was conceptually modelled in the companies by identifying existing and future resources that have the potential to be mapped to the model. This is visualized in Figures 21, 22, 23 and 24.

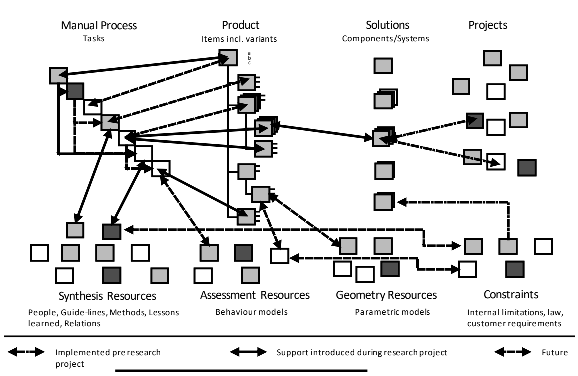

Figure 21. The mapping of company resources in C1

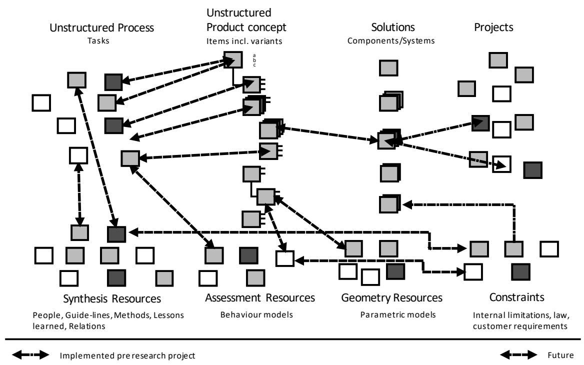

Figure 22. The mapping of company resources in C2

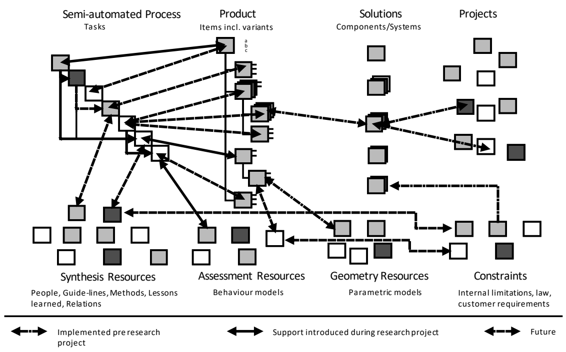

Figure 23. The mapping of company resources in C3

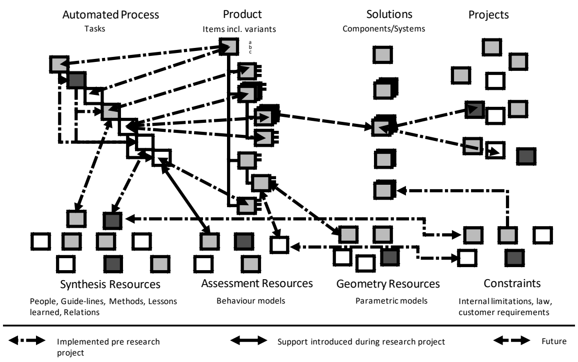

Figure 24. The mapping of company resources in C4

Support was also developed in C1, C3, and C4 with the intention of further aiding the introduction of the DPA and adding to its usefulness. The tools developed to support the application of the DPA in the cases either concerned the addition of a specific type of asset, the creation of a coherent DP model, or both. The focus was placed on C1 to exemplify how the DPA was applied, and this application is further outlined in Paper E. C2, C3, and C4 were not described and exemplified with the same level of detail and were mainly used to strengthen the validity and utility of the DPA.

- C1 is active as a supplier in the automotive industry. Figure 21 visualizes the mapping between existing and future resources in C1. In the beginning of the project, the company had a mapping between projects and solutions using a traditional file structure in its PDM system. It also made attempts to describe some of the know-how regarding process and product knowledge using standard templates. However, these efforts were unstructured and were not employed extensively. No detailed, formalized processes existed, and thus no specific support was used during development.
- Figure 22 shows the conceptual mapping in C2, a company that provides automation services, robotic solutions, and special products to the manufacturing industry. A file structure in the company PDM system provided a mapping between solutions and project. C2 also had a tool

for searching for existing candidate solutions to support reuse. The future objective was to further support PD by developing synthesis, assessment, and geometry resources, constraints, and descriptions of product constructs and tasks; however, the latter two were unstructured because the final products were unique.

- C3 was part of the automotive industry, supplying solutions for attaching equipment to cars. C3 had a synthesis and solution resource to support finding similar existing components and a general product structure. Geometry resources, such as parametric CAD models, existed if new designs needed to be defined. Other resources existed to some extent, which is shown conceptually in Figure 23. The application of the DP model focused on the addition of an assessment resource in the form of a customized software system; its purpose was to streamline the simulation process of an essential product component that was adapted for every new car that entered the market. Constructing the prototype system involved formalizing the calculation engineers' way of working in terms of process and setting up the simulation model. This formalized knowledge was then made available in the CAD environment for use by the design engineers. The introduced support increased the design engineers' abilities to evaluate larger design spaces, investigate trade-offs, and build new knowledge and skills. Organization and responsibilities regarding the product platform, along with associating resources with GPIs, remained to be established.
- C4 supplied components to aircraft engines. C4 had synthesis, assessment, and geometry resources, as well as descriptions of structured product constructs and tasks organized in an automatically executed process, as mapped in Figure 24. In the case studied, the capability of the well-developed company computer aided engineering (CAE) environment was extended via an assessment resource that evaluated manufacturability in terms of robotic welding. C4 had already established management and responsibility of its product platforms. It was progressive regarding its CAE environment, allowing it to assess trade-offs, define, and evaluate design spaces and to automatically use parametric geometry models early on. C4 also retraced, standardized, and published engineering best practice methods and processes, building on the company's knowledge. The company can, however, be supported by the DPA to form a coherent model that ties the different resources to each other. This would also enable the DP model to live on in the PD, which would increase traceability and rationale.

5.4.2 Evaluation of the DPA support applications

As part of the research methodology applied in this work, two iterations were made from the PS-stage to DSII-stage. These two iterations included two evaluations at each of the three case companies where support for the DPA was introduced. The first evaluation can be considered as a "status check" for the three companies. The results of these evaluations can be found in Paper C and in (Heikkinen, Johansson, & Elgh, 2016; Stolt, Johansson, André, Heikkinen, & Elgh, 2016). The second evaluation was conducted using a common questionnaire that focused on the specific support method introduced in that company. Finally, the model and research project were evaluated by all four companies.

In order to evaluate the DPA and the case-specific assets in the form of software applications, success criteria (SC) and associated indicators were identified early in the research project. These were used to judge the success of the project. The SCs and indicators formed a basis of the questionnaire that was given to each company representative participating in the evaluation. For each SC, clarification of the SC's meaning and a statement regarding its fulfilment were given. 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 6, along with the grade of each participating company. 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 was ranked by the case companies in terms of relative importance. In the evaluation, 3-5 persons participated from each company representing positions like designer, calculation engineer, engineering manager, and project manager.

Since the development of the SCs was a joint effort by the participating companies but the ranking of the SCs was conducted by each company individually, not all the SCs were applicable to all companies.

In all cases but two, the averages for each SC are above 3, which indicates an improvement. The standard deviation is below +/- 1 in all cases. Q3 has the highest joint score and refers to the company belief that the ability to reduce the number of formal design loops will increase. L5 has the lowest score, which suggests that the companies believe the return on investment will be low; this is particularly the case for C3 and C4. Suggested improvements and comments concerned the following topics:

- Improved visual appeal of the user interface was suggested to enable use of the support.
- The need for data input in systems was seen as time consuming.
- Several comments concerned the need for a structured method to use the tool and underscored that such a method must be applied by all users. This was seen as hard to achieve.
- Two case companies saw the software prototypes as enabling modelling of knowledge and experience within the company, making it accessible for others.

5.4.3 Evaluation of the DPA

During the final evaluation, company representatives were gathered for a presentation of the final DPA and to answer a self-administered questionnaire. The questionnaire concerned the application of the DPA in their company, what the DP model could contain, what content existed today, and the main challenges and risks. Further, the questionnaire asked about the implications of the DPA in the company.

The companies agreed that the DPA was an applicable and feasible strategy to generalize and reuse processes, methods, and other resources. It was commonly emphasized that the DPA was a great enabler with the possibility of including different ways of storing knowledge.

The main challenge that was emphasized regarded the fact that the DPA might require changed ways of working. Motivating a change in an existing method (which is perceived to work by parts of the organization) was seen as difficult. Other challenges raised included a consistent mind-set, company history, and large variations in the product variants.

A critical implication of implementing the DPA was to communicate the importance of the model to individuals without a comprehensive view in the company. Other critical factors were ease of use, clear value, ease of implementation, accessibility, and the need for education. Identified risks included the DPA’s low number of applications and a too large focus on theory.

Table 6. Success criteria, indicators, associated statements, and ranking.

ID	Success Criteria	Indicator	Statement	Rank	C1	C3	C4
T1	Reuse knowledge	Time to access and understand relevant information	It will be easy to find relevant information using the method/system. The information stored using the method/system can easily be understood. The results or outputs from the method/system can easily be understood.	1	4	3,3	3,7
L2	Time spent to respond to quotation	Time	We will be able to decrease the time currently used to respond to quotation.	2	3,8	4	3,3
L3	Time spent on project	Number of design hours per project	We will decrease the number of labour hours in our projects by implementing such a method/system.		4,3	3,2	3,3
L1	Short start up time	Time spent to introduce new user	The proposed method/system contributes to decreasing the learning time.	3	4	-	3
L5	Time invested to build the support system	Investment/use	The benefits exceed the costs of development and implementation.		-	3,3	2,5
Q1	Assure that requirements are fulfilled	Number of changes after verifying tests	We will be able to reduce the number of changes that must be made after verifying tests (i.e. increase our ability to ensure that requirements are fulfilled at an earlier stage).		3,8	4	3,7
Q3	Number of loops	Number of formal design loops required to achieve series production	The method/system will decrease the number of unplanned changes in series production.		4,3	-	-
P3	Reuse components	Number of carry-over parts	We will be able to increase the number of carry-over components.	4	4	3	-
P2	Support the designer	Assessment by the designer	The method/system will support the designer to a higher degree than the existing solution.		3,8	4,2	3,3
Q4	Keep the project time	Keep the project time	The method/system will contribute to keep the time for start of production (SOP).	5	-	3	3
Q2	Lower number of errors	Number of changes in series production	The method/system contributes to decreasing the number of unplanned changes.		3,8	3,4	2,3
P1	Resource utilization	Number of designs created/design hour	Having a method/system like the presented one will make utilizing our resources more effective.		4	3,8	3,3

The companies agreed that the concept of product platforms in general had expanded from being component-focused to include more engineering assets. The DPA was believed to reduce misunderstandings with customers. The project had also generated discussions and additional development initiatives within the companies. Finally, one participant stated that the DPA had, “Led to a greater understanding for the need to see the complete picture and for different disciplines get a view of each other’s problems and challenges”.

5.5 Paper E

Paper E outlines a refined DPA that is a detailed application of that presented in paper D. To investigate its feasibility, the DPA was introduced and tested at a company acting as a second tier supplier in the automotive industry. The focus was on a business area in which the company develops and manufactures uniquely customized car interior subsystems. The company was, in principle, willing to customize or redesign any part of the system if it generated business, which inhibits them from standardizing components and modules to a high degree. Because of their high level of customization, speed and accuracy in the quotation process is a key issue. Speed is paramount for returning an offer to the customer quickly, while price accuracy is crucial to balance the client's budget realities and the case company's profitability standards. It is, therefore, vital to reuse as much in the way of pre-existing development engineering assets as possible, which places a heavy demand on finding the necessary information and judging its applicability accurately.

5.5.1 A product platform support tool and PDM

The DPM functionality is extended from Paper C to allow associations with content in the company PDM system to be created. Also, the generalized information model presented in Paper D, which supports several types of assets, was implemented. Figure 25 depicts how objects are saved in a joint database. This approach also guarantees that data redundancy is minimized and enables multi-client and concurrent usage. The use of the DPM does not require a change in the methodology or structure employed by the PDM system. However, to execute the DPM, certain changes *are* required to the PDM database; database tables must be added, which make it possible to store the different object types included in the DP model. This involves the addition of tables for all classes in Figure 26. Additional database tables were created to allow many-to-many associations between objects. Design elements can be used in several GPIs and vice versa, which requires specific database tables to delineate the association.

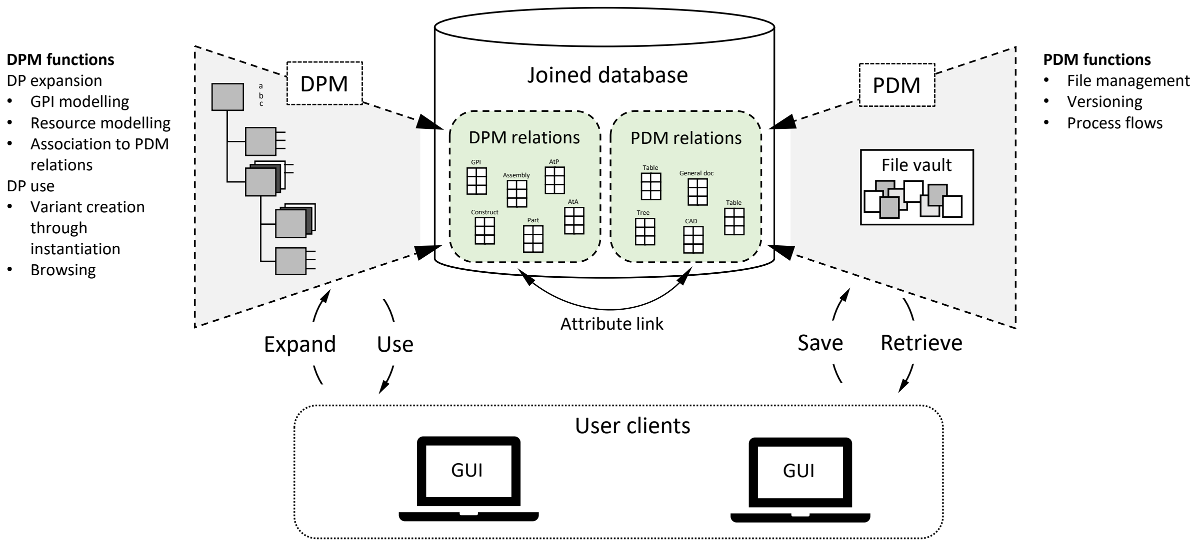

Figure 25. The principles of DPM and PDM

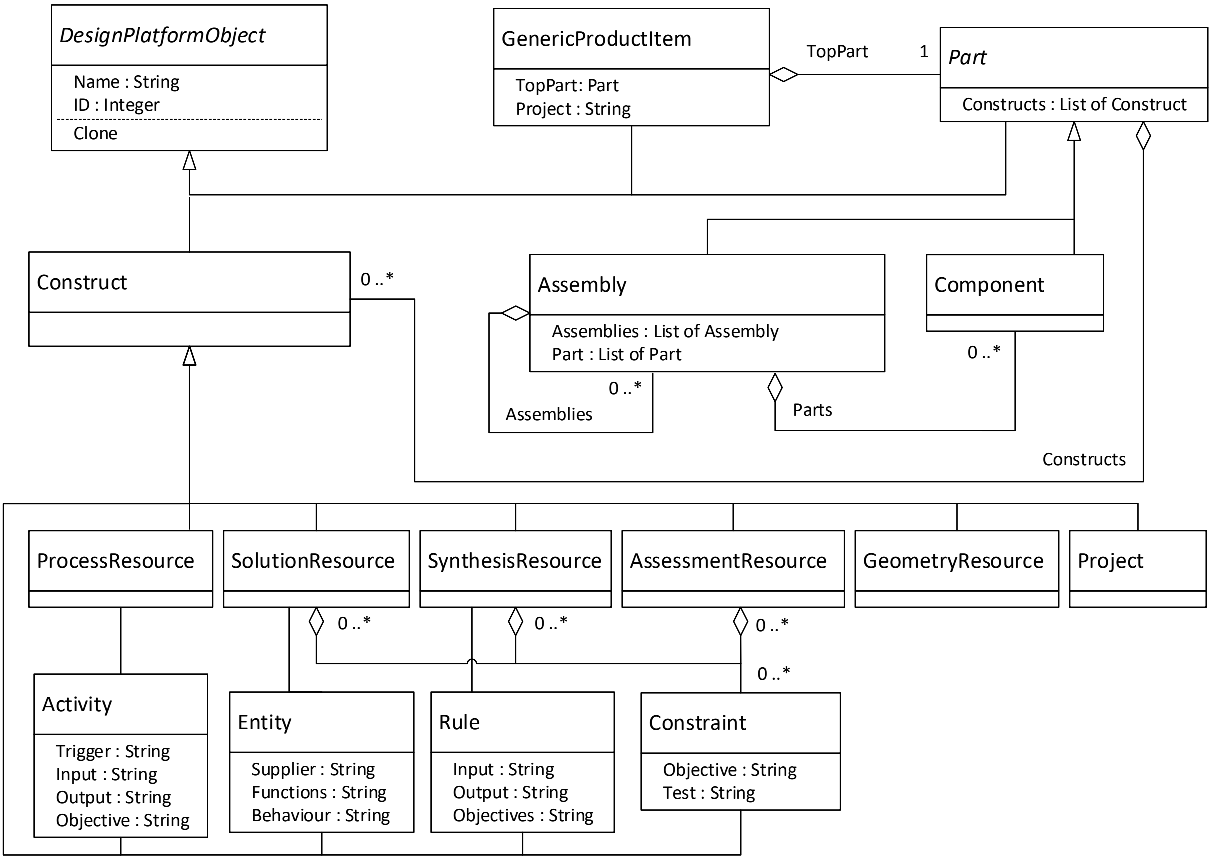

Figure 26. The class relationship diagram used for DPM, and how it relates to the generic DP model

The software tool is based on .NET and uses many standard forms to display trees and lists. The DPM enables the user to model GPIs as trees whose main structures, the holders of engineering assets, are composed of assemblies and components. Each level in the structure can have associated assets that are visible either as nodes in the tree or as items in the Engineering Assets view. Different GPIs can be viewed, added, and changed. The variants belonging to a specific GPI can be listed and searched through the Design Instances view. The properties of objects in the user interface can be viewed and changed. To create a functional prototype, a PDM viewer was integrated to enable easy access to the PDM content. This was possible by reading from a PDM database table that contained all base classes and also pointed to the database table containing all instances of each class. The database tables could then be searched by sending simple string values to the database. The objects displayed in the PDM database viewer could then be associated with the GPI and variant structures. Further, GPIs, variants, assemblies, and components were all saved, retrieved, and updated using specific DPM tables, which were created in the PDM database. Figure 27 shows an instance of the application user interface where a GPI “4 way” has been selected. Consequently, the engineering asset objects associated with that GPI level are displayed, along with some lower structural levels.

The remainder of Paper E covers the DP model expansion and use stages. How the GPIs were identified and modelled in DPM is presented below. Different assets were also identified and modelled, using DE templates, and then attached to the structure.

Approximately 100 design elements were identified and formalized on spreadsheets. Since the generic nature of some of the design elements, they could be reused on several GPIs, producing many more relations between assemblies/components and design elements. Figure 26 further describes the addition of design elements to the generic information model. The ProcessResource was realized through modelling the design element *activity; Entity* design elements formalized and expanded the metadata of SolutionsResources; *Rule* design elements formalized SynthesisResources; and *constraints* were used to model different limitations regarding the previous resource types.

At the company, the DPM was evaluated by potential users, consisting of an engineering manager, a project manager, a lead engineer, and a designer. Using individual questionnaires, they were invited to comment on and grade the performance of the support tool and the associated working approach. The overall judgment was positive. The company representatives anticipated that implementing DPM would decrease project time due to a better overview and better access to the different assets. They believed the need for formal design loops would also be reduced. The company representatives emphasized that DPM would increase the level of support for the designer and the level of knowledge reuse. In identifying areas of the DPM that needed additional work, they focused on the user interface. They viewed the manual input to the system and the need for a structured working approach employed globally across the company as time-consuming.

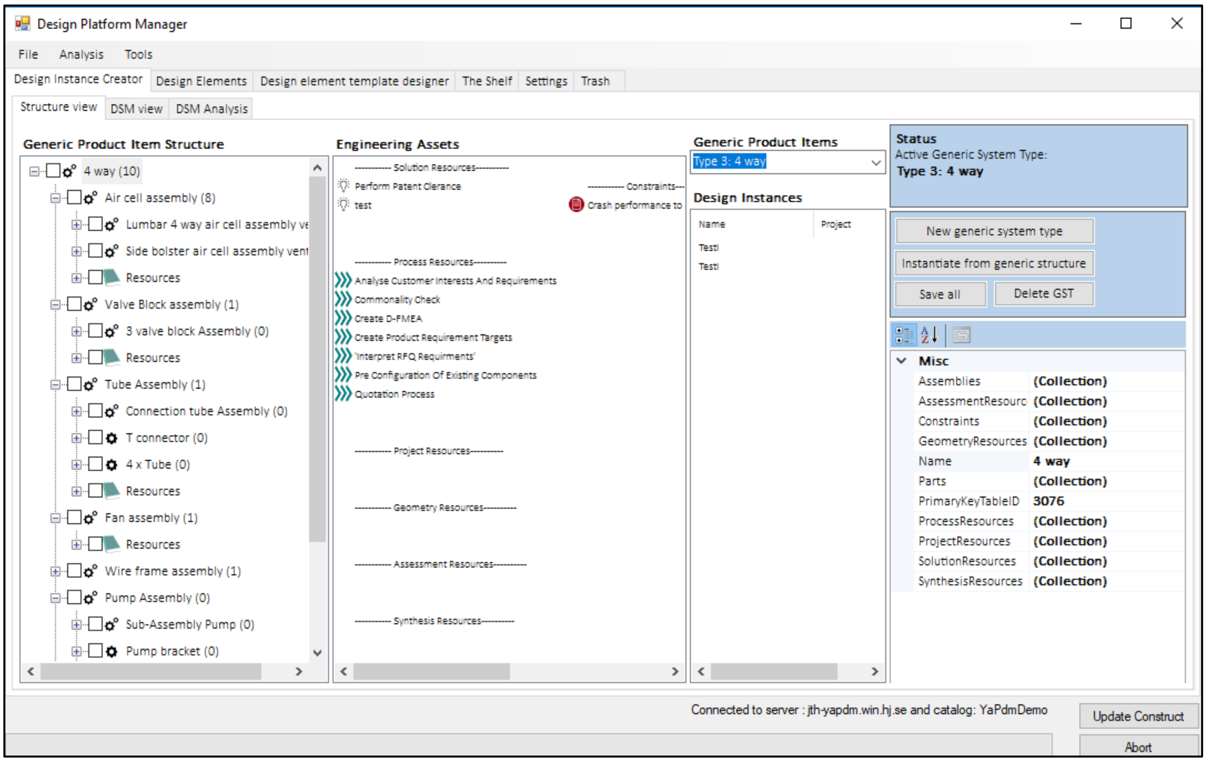

Figure 27. A screenshot of the application Design Platform Manager

5.6 Paper F

An area of concern in product and production development is the design and manufacture of machine tools used for part manufacturing, which are often a large investment and a critical bottleneck in the development of a product. Changes in customer behaviour and market demands have resulted in an increasing number of product variants, decreased product lifecycles, and shorter time in product development. These factors have put high pressure on manufacturing companies and introduced tough competition with low-wage countries. Few examples of product platform approaches are found in this area, which might be due to the high level of uniqueness and the expertise associated with designing the tools. It is challenging to distinguish between a generic and a variant-specific solution when development is conducted in customer-focused projects, and the integral nature of the product makes it difficult to find a division of the product that supports reuse.

Previous papers on the DPA have mainly focused on the identification and modelling of assets into objects. However, the relations between the asset objects also play an important role and can be considered as assets in themselves. The relations can communicate how and in what order assets can be used, and thus they introduce a stronger process focus. The aim of this paper was to investigate the application of the DPA in a company that designs and produces unique high-pressure die-casting (HPDC) tools for different applications and customers. A focus of the paper is on modelling and managing relations within the DP model to enable companies like the case company to utilize platform thinking to a higher degree and thus to increase their efficiency in product development.

5.6.1 The addition of relationships to the DP model

In this paper, a relation is referred to as an object that connects two other objects and holds information about the connection's purpose and nature. Figure 28 shows a class relationship scheme of the DP model constituents and how a generic product item structure holds constructs of different types. The Relation class, which is added to the DP model in this paper, can be of different types and essentially relates the different construct types to each other. In order to support the use phase of the DP model, relations that specify more than hierarchy are needed. These additional relations aim to support the process of using the objects together on a detailed level.

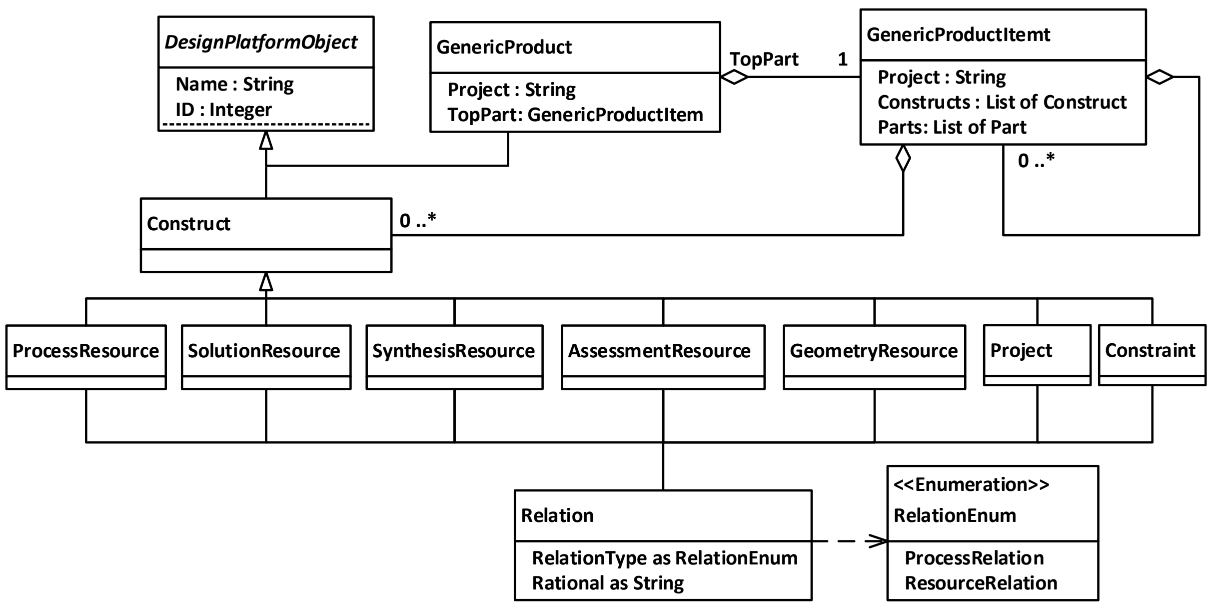

Figure 28. Generic class relationships of the DP model

5.6.2 The asset relationship matrix

Multiple relations exist between assets that cannot be modelled using only the product structure tree. The relations that become obvious in the tree structure are hierarchical and focus on how systems are grouped together. Within and between groups of hierarchical levels, additional guidance is needed to, for example, suggest the sequence to perform activities or indicate which knowledge can be used as a resource in a process. The asset relationship matrix is composed of inter-domain DSMs, for modelling relationships between objects of the same type, and multi-domain matrices (MDM), in which objects are of different types. The generic relationship matrix is shown in Figure 29 with illustrative relations. The relationships that connect the different objects are objects themselves and are instantiated from the following types:

- *Process relationships* (P in Figure 29) state the sequence order of objects.
- *Resource relationships* (R in Figure 29) state if a certain object is to be used as a resource for another object.
- *Hierarchy relationships* (H in Figure 29) are derived from the tree structure and are not an explicit relationship type in the DP model since they implicitly exist in the aggregation relationship of the GPI class.

		Synthesis					GPI					Process					Solutions				
		6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	Sol1	Sol2	Sol3	Sol4	Sol5
Constraints	1. Con1.										R										
	2. Con2.							R													
	3. Con3.									R											
	4. Con4.						R		R		R										
	5. Con5.							R	R												
Synthesis	6. Syn1.	■				P										R					
	7. Syn2.		■		P								R								
	8. Syn3.			■																	
	9. Syn4.	P	R		■									R							
	10. Syn5.				R	■										R					
GPI	11. GPI1.						■												R		
	12. GPI2.							■	H	H	H						R				
	13. GPI3.								■										R		
	14. GPI4.						H	H		■								R			
	15. GPI5.										■										R
Process	16. Pro1.											■				P	P				
	17. Pro2.												■		P				P		
	18.Pro3.		R									P		■				P	P		
	19. Pro4.														■			P			
	20. Pro5.													P		■					P
Geometry	21. Geo1.						R														
	22. Geo2.									R											
	23. Geo3.				R			R													
	24. Geo4.									R											
	25. Geo5.							R													
Assessmet	26. Asse1.																				
	27. Asse2.																				
	28. Asse3.										R										
	29. Asse4.		R						R												
	30. Asse5.																				

Figure 29. Generic relationship view of the DP model

The different types of assets thus create specific domains depending on their function. Constraints are inputs and do not have a domain. Solution resources are the output of the process and, therefore, do not have a domain. These two classes relate to each other through other assets. Also, geometry and assessment resources are supportive resources and relate only to other class types. The remaining domains are explained in the following:

- *Synthesis resource domain*: This domain consists of all the resources used for design synthesis that have been retraced and formalized.
- *GPI domain*: This domain states the hierarchical relationships between generic product items and thus embodies the generic product structure to which other resources are attached.
- *Process domain*: This domain describes the process relationships between process steps and milestones that have been formalized and generalized. These, together with the GPI domain, create the core of the matrix and are essentially placeholders to be related to other types of objects via MDMs.
- *Multi-domain matrices*: The MDMs are used for mapping relationships between different objects of different domains.

5.6.3 The support tool and PLM system

To demonstrate the asset relationship matrix, a tool was developed and presented, which is visualized in Figure 30. The process of applying the DPA was executed in a similar way to the process explained in previous papers. The identified assets were captured and added to a PLM system by altering the system to enable the modelling of generic parts and process nodes.

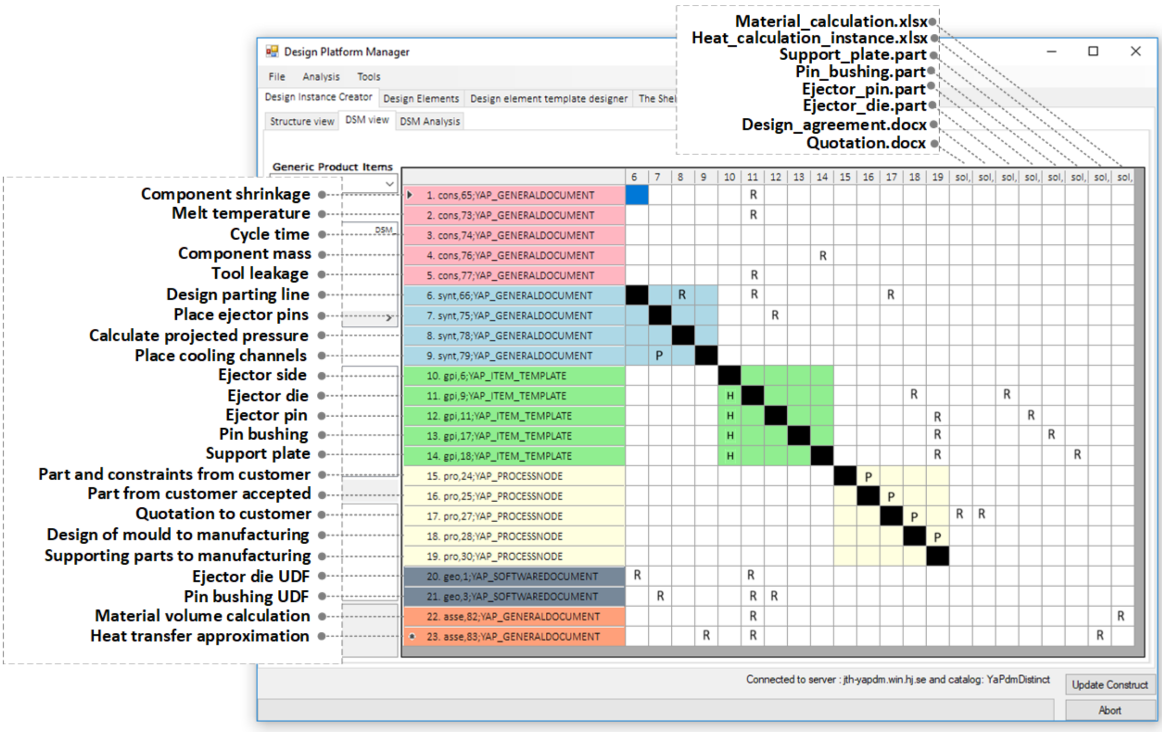

Figure 30. Screenshot of the asset relationship matrix in the application

The support tool prototype application reads the PLM system database, allowing it to access and work on the same data. The objects in the matrix are selected from a database interface where the user classifies them according to assets type. In order to construct the matrix, the objects identified in the previous step need to be saved or created in the PDM system. There are essentially three stages in the DPA at which the matrix can support the company:

- Modelling: the company receives a format that they can use to formalize their assets, such as items, processes, tools, and know-how.
- Execution: the model can act as a map, guiding the engineers to applicable knowledge, which essentially can guarantee higher quality and shorter lead time.
- Maintenance: the model can be used for change analysis. The model provides rationale and traceability for future changes and product updates.

5.7 Paper G

Paper G explores the possibility of DPA application in IHB in order, primarily, to strengthen the validity of the DPA by applying it to several kinds of industries. Product platform strategies have lately become of interest to the IHB industry which faces similar challenges to mechanical manufacturing companies. However, some challenges seem characteristic of IHB, several of which are discussed in this paper. Regarding the management of information, a challenge for IHB is that there are two dimensions to the physical building. One is the traditional part view, which specifies the components used to construct a house. The other important dimension is the rooms, which are merely the effect of, or the desired function achieved by, combining different parts. Information exists that describes both these dimensions, but the common PLM systems, traditionally used by the mechanical industry, often focus on the part dimension. New models are needed that can cope with the additional dimension.

To take a first step into this industry, Paper G presents empirical data from interviews, workshops, and document reviews that focus on identifying the different assets that exist in a IHB company and what intangible knowledge exists with the potential to become formalized assets objects. The case company offers products such as schools, kindergartens, elderly homes, and offices. Standardization in production is fundamental to the business concept. The building system is based on volumetric elements in turn-key contracts, meaning that the company covers all disciplines and the entire construction process. The product portfolio attracts public clients with large budgets. Consequently, these clients are not afraid of setting a narrow frame of demands or to continuously alter their demands. Given the characteristics of the construction industry and the amount of money invested in the products, there is no prototyping. Rather, development is carried out in actual projects and, if the evaluation prescribes, solutions are incorporated into the product platform. However, the claimed product platform has no clear boundaries and seems to be more of a model describing the aim of the company than a direct indication of how the company works. Thus, two efforts were judged necessary in the company. First, to establish a concept to apply the DPA, and, second, to support the use of the DPA. The DPA is suitable for IHB, which needs to use and manage several different types of information in a product platform context. Therefore, the first step was a study to identify assets, followed by a PLM system architecture proposal regarding how to support the DPA.

5.7.1 Identification and analysis of assets in IHB

From the empirical data, two levels of assets were identified. On the concrete level, assets existed that are currently used in the company. These included formalized house types, finished designs with drawings, documents describing requirements, parametric CAD models, and guidelines for design and production. These documents were scattered on the file server and thus did not provide a coherent view of what the product platform comprised. Furthermore, intangible assets were identified that have the potential of becoming objects and of supporting the product platform definition further on. These are presented in Table 7.

Table 7: Potential asset objects in the case company

Sales	Design	Production
The offered range of products and built-in knowledge	Allowed technical solutions, customization/Adaptability	The production process, build-up, and know-how
Government procurement knowledge	Regulations updates, energy efficiency, and sustainability	Disturbance log, experience feedback
Market knowledge and demands	REVIT (CAD software)	Factory limitations
	Guideline system	Protocols and checklists

A more detailed investigation regarding the existing guidelines was performed using a DSM strategy. The guideline system consists of documents describing designs, requirements, and much more. In many cases, they refer to each other in the content, and uncovering these hidden relations became the goal of the DSM investigation. The investigation showed that the documents referred to each other in as many as six steps. In some cases, the document references created in a loop. It was also evident that much needed information to construct a house was left out, leaving it up to carpenters to complete. The result showed that the guideline system has a weak connection to the product platform concept that the company strives to employ. The document identifiers are based on the guild system, as tradition in the building industry, and not on levels in a product architecture. Moreover, the framing is highly specified compared to other areas, leaving these other areas with a low level of standardization.

5.7.2 PLM system architecture for IHB

To support the introduction of the DPA, a PLM system setup was proposed together with the asset relationship matrix presented in Paper F. First, the common capabilities of a PLM system were placed in the context of the company's state of practice, as seen in Table 8. This table describes common functionality in a PLM system and explains how it could support the company's current situation. The proposal includes how product structures could be modelled and linked to the existing documentation that supports the realization of the specific part. Each standard house that exists can be its own generic structure and connected to each instance that is produced. A process was also identified to describe the flow from quotation stage to production, which can also be linked to the generic product structure and documentation. Finally, the paper discusses the future work and additions needed to the DPA to better support IHB. For example, the concepts of "room" and "information receiver" are proposed as new attributes for each asset. These attributes together with e.g. what part the asset concern allows for different views on the assets which can support different disciplines. The paper also concludes that it is important to decide early on what asset types can be incorporated into the DPA and which have to be changed to guarantee a coherent model that supports the DPA.

Table 8: PLM system capabilities to support the company.

Name	Description	State of practice
Revision management	Managing files and objects stored in the PLM file database.	Documents are stored in an MS Windows file structure without support for revision management.
User and access management	Enables different professions within the company to have different kinds of access to the data depending on their role.	The receivers of specific data are not identified, and different levels of access cannot be used.
Part management	Allows for modelling and managing product parts and structures such as components and assemblies.	Parts are only handled in the CAD software without an assembly structure. Parts are not file based in the company CAD system obstructing reuse between projects.
Process management	Enables keeping track of approval status and process progression connected to the data	Many processes are formalized, and no processes are supported by IT tools.
Attribute management	Allows for enriched descriptions and different views on objects and linked data.	No attribute data exist on current documents except for the data encapsulated in the documents and the categorization according to the guild system. Suitable attributes can support both part structures and room descriptions.
Link objects	Makes it possible to link related objects to each other and attach attributes to the link.	Existing links are hidden in the documents and cannot be managed separately from document content, making it hard to obtain an overview of how the assets are related.
Object orientation	Separation of different classes of assets depending on content and designated use.	Only MS office and CAD file formats are used, which does not communicate content or use.

The paper outlines examples of a generic product item and process structure and shows how they are realized in the PLM system. The asset relationship matrix is also applied and discussed. Essentially, the DPA provides a means for the company to work platform-based and thus to become more resource efficient. The PLM system and asset relationship matrix increases the possibility of using the DPA in a structured way, providing information traceability and collaboration. Using the PLM system as the main information source for all employees active in the process of selling, specifying, producing, and delivering a house increases the possibility of not losing information. Today, much of the data is manually moved and transformed between information systems and receivers, which is a large source of waste and errors. Using generic item and process structures means that much of the work is finished at the specification stage and only a few alterations are needed. This is preparing the company information model for the introduction of a configuration system to automatically deal with parts of the specification that could improve lead time and profitability even more. This paper also contributes by introducing production in the DP-model process view, which provides more possibility of collaboration between design and production. This also allows the engineering bill of material and the manufacturing bill of material to be visible at the same time and for the disciplines to gain knowledge of each other's domains. For this specific company case, since the generic item structure could be connected to the production process, a design's documentation could be directly connected to the relevant production step and thus supply the production staff with the correct documents.

6

DISCUSSION

The discussion chapter aims to add different viewpoints to the result in the light of what other authors have written on the subject. The chapter also answers and discusses the research questions and considers the validity of the work.

This research has presented results regarding industry needs and prerequisites concerning the use of product platforms. It has followed the conceptualization, development, introduction, and evaluation of the DPA and support tool to answer to the identified need. The primary aim of the research is to support ETO companies in becoming more efficient. This efficiency involves being responsive to changing and conflicting requirements during the scoping and development processes, as well as reusing the tangible and intangible engineering assets that are continually the outcome of engineering activities. This aim is of a long-term character and was not expected to be fully achieved within the scope of this thesis. The result presented, however, points in a direction that supports this aim. The goal of this thesis was to investigate the ETO industry's current state regarding product platforms and to propose an approach that allows companies in this industry to take advantage of product platform principles. The evaluation results indicate that this goal was achieved. The introduced product platform approach has evolved from an idea in Paper A to a refined, formalized, applied, and generalized approach in the subsequent papers. This chapter starts by answering and discussing the research question. It then provides a discussion of the final DPA, as described in Chapter 4, followed by a discussion concerning the validity of the research result and process.

6.1 Answering and discussing the research questions

The thesis aimed to answer three research questions. The following discussion clarifies the connection between the questions and the results.

6.1.1 RQ1: What is the current state of the utilization of product platforms for ETO companies?

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

The literature and the interviews support the initial assumption regarding the question of applying a component-based product platform at ETO companies. The result indicates a need to engage in platform-based development but acknowledges a real challenge to achieving this aim. This view is supported by(Ulf Högman et al., 2009; H Johannesson, 2014). The challenge lies in the inability to preplan variants for future customer orders for which the specific requirements cannot be known. Requirements also tend to change throughout the development process as a result of unexpected changes by the customer or in interacting system interfaces. While these challenges are frequently pointed out in the literature, there are few available approaches to manage this challenge. The case companies had not explicitly formed models to describe their product platforms, if indeed such platforms existed. However, attempts to reuse assets to some extent were evident in all the companies. The most common way to do this was to use old solutions as baselines and to adapt them for new orders. These solutions included geometry models and standards described in documents. However, the companies seldom had an overall structured method. Neither was there a common view nor an agreement regarding what a product platform was or included, even in the companies who claimed to have one.

Some conclusions could be drawn by investigating company characteristics and the elements that constitute each product platform. Company factors that increase the need for a higher product 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 was; for example, a guideline is more abstract than a manufactured component. It follows that the higher the abstraction level of the product platform, the more engineering needs to be done to deliver a product. Moreover, the higher the abstraction level, the more the companies tended to describe the product platform as an explanatory model rather than something explicitly defined.

TD is a prerequisite that is closely linked to the development of product platforms. However, the connection between the two has not been studied to a great extent in the literature. The definition of TD differed among the 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's product. The type of deliverables also differed, ranging from physical prototypes to feasibility studies using trade-off curves.

6.1.2 RQ2: How can a product platform approach be conceptualized to support customization for ETO companies?

The main result of this research and the answer to this research question is the DPA. The DPA uses assets like engineering knowledge as part of the product platform. The DPA should not only include tangible items but also support 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 product platform, changes throughout the development process needs to be acknowledged. Further, an approach needs to be applied in the scoping, quotation, and order processes to define design spaces that allow for adaptation rather than single solutions. Reuse must permeate the development work by generalizing solutions. The product platform must be able to host a heterogeneous range of descriptions created in a company to maximize flexibility. Thus, the aim of such a product platform is more focused on achieving efficiency in development than in production.

Judging by the results of the interviews and workshops, different needs and resources can be identified in the companies. During a workshop, company representatives were asked what would help them manage fluctuating requirements more effectively. Their answers included being adaptable to changes and accessing previously created knowledge in different ways and formats. Several possible formats were mentioned, such as guidelines, trade-offs, design rationale, and more. In order to function in a platform-based fashion, these ways of working and the diversity of description formats need support, but the current product platform approaches offer inadequate solutions. The methodologies and models for working with product platforms have long focused on physical, component-based product platforms. The results of this thesis indicate that the positive effects of using a product platform can also be leveraged by other constructs and resources already present in the companies. The set of already existing descriptions can be enriched by introducing new types of classes that can embody knowledge and allow them to be part of the product platform. Using a more dynamic product platform definition will also permit its capabilities to evolve over time as new knowledge emerges.

The name design platform was chosen since the term "design" refers both to the activity of designing as well as a design as a thing. Thus, the initial concept aimed to realize the DPA in a way that allowed for both a process and a product domain to be included, since both are paramount when developing a product. In an ETO context, however, these domains cannot be captured in full and thus both are included, with the potential of one domain supporting the other. The DP model has been used throughout the papers as a conceptual image to map the current state and future target conditions of the included companies in Paper D. It has also been formalized beyond an explanatory image by using UML and integrating the information model in computer software.

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

The DPA can be used in two main ways. It can be employed as a mind-set and an approach to relate to the assets present in a company. It can also be used to formalize the approach explicitly and to conduct the necessary work related to setting up the approach. This research question relates principally to the second way of using the DPA. The formalization and application of the approach have resulted in the DPM tool, as well as in two PLM system implementations in different settings. Thus, different ways of realizing the DPA have been provided, supporting the usefulness and validation of the DPA.

First, the complete product platform definition must be brought together into one coherent view that allows the user to work at different levels of abstraction, including 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 product platform. The view must encompass the main processes, GPIs, the related variants, their constructs, and the relations between them. The relationships between assets are paramount to understanding how they are intended to be used together and what function they have relative each other. The model also must include the possibility of instantiating it to create variants. Also, the current documentation used in companies must be evaluated, and a decision needs to be made regarding how these documents should relate to the DPA; they can be integrated as specific types of assets, excluded from the DPA, or changed to better comply with the overall goal. The support cannot be a standalone application but must be used according to a strategy that integrates or is integrated into PLM. This is important for the following reasons:

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

This thesis has presented the DPM as a means to support the model's formalization and to show how a conventional PLM system can be adapted to support the DPA. The application supports all the above criteria by the creation and application of a generic information model. The approach offers an integration with the PLM 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 modelled using the tool. DEs and other asset types were created and saved in the PLM system to be linked to the models created in the DPM. Additionally, other resources important for the realization of product variants were identified and linked to the DPM. The only location used for storage is the company PLM database and vault. The evaluation of the tool shows good results in terms of functionality.

Paper F and G use a conventional PLM system for the sake of setting up and supporting the DPA. This can be a favourable approach due to less software maintenance and a less complex application landscape in the company. The PLM system, however, must be customizable to be able to manage the principles and object types of the DP model. This applies especially to the concepts of a generic product and process, which are linked to each other and their realizing constructs.

6.2 Scientific and industrial contribution

Research within engineering design has the aim of contributing both to science and the industry. From the perspective of science, this work has filled a gap in the literature identified through several systematic literature reviews and by empirical data. The DPA was then developed, guided, and delimited by the identified gap. The DPA builds on established theories of product platforms and embodies a novel approach. It has been verified and validated both by users and through application in different fields.

From an industrial perspective, the industry has gained from this research in several ways. The companies of study have directly been given an approach for further development and refinement. However, similar companies can gain the same benefits because the DPA is described in a generic manner. With the DPA, ETO companies have gained a structured way of viewing and capturing continuously developed knowledge, which increases the possibilities of receiving similar advantages to a product platform. From this perspective, a company can gain increased efficiency from the reuse of company assets, increased control and overview, and using the model to assess the impact of changes and decision-making. The industry has further been provided with demonstrators that have shown the possibilities of the DPA.

6.3 Discussion of the result

The use of the DPA is intended for, but not limited to, companies developing highly customized products, often in B2B environments. Based on the result, it is expected that the DPA has a wide application area, allowing it to be applied to businesses other than those examined for this study. Such a product platform can enhance efficiency and assist companies in leveraging a competitive edge. However, the inability to foresee future requirements and to preplan variants for a future market present challenges to developing such a platform. 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 the right technologies for future products. A successful product platform approach involves enabling a coherent format and method to support preparing adaptable solutions during the TD stage for later customization. The DPA thus supports development outcome, like guidelines, activities, rules, parametric models, etc., all which enable a wide design space. To follow the guidelines outlined in Paper D regarding the TD and PD interface, a wide overview and cross-functional teams are needed, spanning functions such as the product department, TD, PD, production, supply chain, and sales. Models and tools that support requirements management over the product lifecycle are needed to keep track of internal and external requirements. A cross-functional organization needs to be established to manage the DPA, including the assets that have the correct set of models and tools to assess new technology and map it to future products. The DPA supports formalizing best practices and thus allows for omitting more uncertain parts of the process, if necessary.

Definitions regarding product platforms have evolved from being component-based to consisting of knowledge and people. Sawhney (1998) terms the concept “platform thinking” and states that product variants within a product family share a common gene pool. This gene pool has been referred to as something that is generic in this thesis. Different products and business models allow different genes to be the core that is shared between product variants. However, since most companies have a specific niche, every company has a product platform or applies platform thinking to some extent, according to the definitions offered by Sawhney (1998) and Robertson and Ulrich (1998). However, these product platform definitions are not especially helpful; they only place a new name on something that already exists without supporting increased efficiency. This thesis argues that in order for something to be called a product platform intended for PD, it needs:

- A coherent description, such as a model that expresses what is included in the platform and the applicability of its content.
- A methodology supporting the creation, expansion, and use of the model.
- A clearly defined and expressed generality that will be the common denominator among any variants derived from the platform.
- An advantage compared to not using it, meaning that it must support the forming of variants.

Usefulness can be added to these requirements. In order for an organization to accept this kind of model, it needs to be understood by the people who will use it and be supported by it. Therefore, the DPA has been developed with the design engineers in mind. This has not been a strong or well-defined requirement during this work, but it has contributed to decisions contributing to the final DPA. This has resulted in an approach that might not be completely correct in all theoretical aspects because usability has been taken into consideration. Some concepts, like function or other types of relational classes, have been excluded due to the difficulty of working with them in practice.

The traditional way to use product platforms is to preplan variants and optimize the design to achieve both high commonality and high distinctiveness. For ETO companies, this approach increases the risk of losing projects to another company that agrees to a higher level of customization, if their predeveloped variants do not comply with a particular customer's requirements. The DPA acknowledges fluctuations, realizes that change is inevitable, and adopts a flexible model and working approach that permits companies to be better equipped when offering higher levels of customization. The DPA, similar to the technology platform (Ulf Högman et al., 2009) or platform thinking (Sawhney, 1998), focuses on making use of the abstract definition of product platform given by Robertson and Ulrich (1998). Early publications on utilizing product platforms have had a business focus and included whole firms (McGrath, 1995). They have, however, lacked detail when it comes to describing how to apply these ideas in design and manufacturing processes. Later publications have had a detailed focus on the artefact as a way to gain the business advantages of platform thinking (Krause et al., 2014). Levandowski et al. (2015) and in (Hans Johannesson, Landahl, Levandowski, & Raudberget, 2017) provide examples of approaches utilizing a higher-level product platform definition, where modules are both scaled and configured conceptually; however, detailed examples and case applications are few. The DPA introduces and allows for different knowledge formats, from physical components to more abstract descriptions, to be included in the product platform, making it useful for customization. Even though a company has several functions that serve the manufacturing and delivery of products, the GPI is a suitable view for the DPA to be based upon, as in the PVM approach (Hvam et al., 2006), since the product can be said to be the core of the company, according to (Brière-Côté, Rivest, & Desrochers, 2010). A generic process view must also be tied to the GPIs and constructs, thus forming a product platform model that allows two different views. The GPI remains the common denominator between different company functions, which makes it a suitable carrier of knowledge from an array of disciplines. Knowledge from disciplines like purchase and manufacturing, calculation, and quality engineering can be described and coupled with the generic product concept for early and concurrent consideration and support during quotation and design. The generic process, on the other hand, guarantees that assets are used in the correct order. The DPA also differentiates between what is generic and instance specific. This characteristic is opposed to using only a project structure, which requires users' knowledge of previous projects to find applicable information.

When introducing the DPA in a company, there are many possibilities regarding how it is to be applied. For example, what parts are to be considered? What assets and company disciplines can be incorporated? What level of detail is appropriate? These questions can only be answered with the specific company of application in mind. Different types of products and business models provide different opportunities for DPA application. Companies work on different levels of abstraction, which must be kept in mind when setting up a DPA. The asset relationship matrix outlined in Paper F can help to visualize these levels of abstraction. Figure 32 shows a conceptual image of how different layers and widths in a company can work according to the DPA. The matrix can be constructed as a management

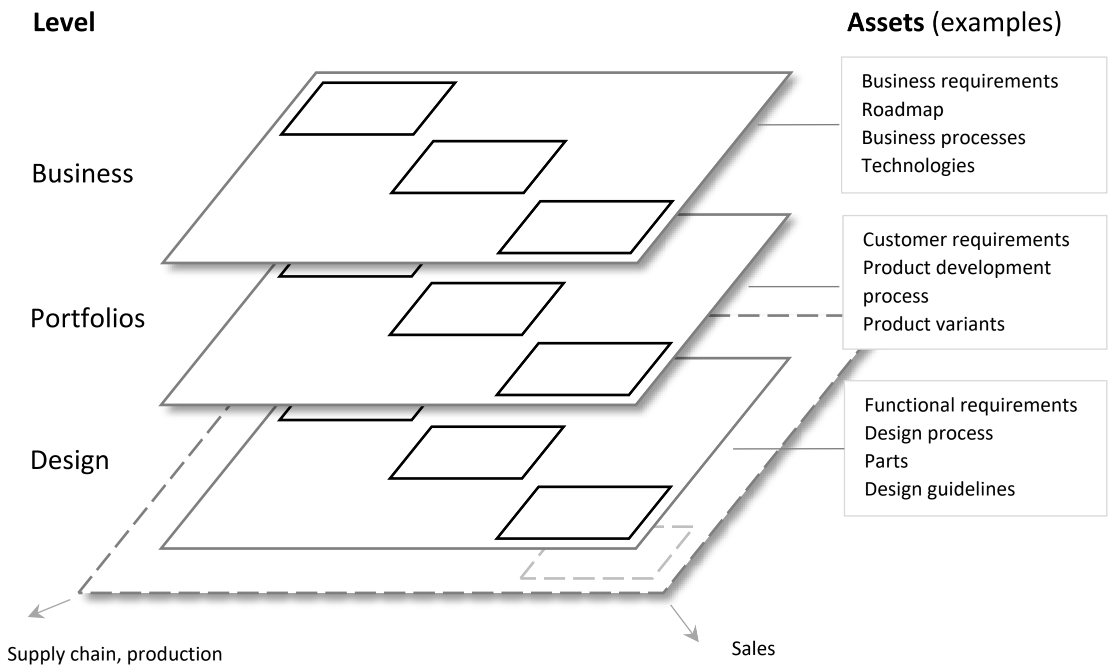

Figure 32. Multi-level platform thinking and expansion

tool, including the primary business processes, primary customer requirements, and so on. It can also be applied on a detailed level in a specific product development category or team within a company, connecting all specific assets used daily and/or on a product portfolio level. It is important to notice that these levels are connected to each other. Depending on the business model, it possible to work either process- or product-based.

- If the company applies a configure-to-order approach, then there can be increased focus on the GPI domain and the synthesis resources describing configuration rules.
- If the company applies an ETO approach or if the product is integral, a higher focus on the process is suitable. The process is, in turn, connected to other kinds of assets to support the design process.

The expansion stage of the DPA requires being proactive during PD to find best practices and being zealous when creating the descriptions that arise when it is used. It is at this stage that the DP model evolves by the addition of new assets, which will be resources in future projects. Introducing generic assets in formats such as parametric CAD models, task descriptions, and trade-off curves enable an approach that uses definitions of spaces rather than point-based solutions. The need to master and manage fluctuating requirements requires the use of a product platform approach; “master” in this sense means being responsive to fluctuations, not avoiding requirement changes. The DPA aids development loopbacks when iterations need to be made and evolves as new knowledge is added to it.

There are, however, risks to be considered when applying the DPA, as with any type of product platform application. The risk most discussed in the literature is not reaching a sufficient product differentiation when product variants share many subsystems and components (Timothy W Simpson et al., 2014). This also becomes an issue when applying the DPA and needs to be carefully considered. Other risks include patents, which make it hard for ETO companies to reuse a specific solution in a new context. It is also not uncommon that manufacturing tools used by an ETO company are owned by the customer. This prevents the ETO company from using the tool in other applications. However, the DPA becomes powerful in this regard as it is built on the assumption that solutions and specific geometry are hard to reuse for these companies. Formalized knowledge, the focus of the DPA, has a high possibility of being reused and avoids the risks previously described.

Revisiting the evaluation results in Paper C and D 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.

6.4 Discussion of the verification and validation of research

Verification and validation of this work have been carried out in several ways and can 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 in which 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 time frame of a doctoral thesis. Design research is typically concerned with developing methods and models to support the design process. To verify and validate these methods and models, tools are developed to implement and observe the methods and models in their intended setting. This, however, creates challenges regarding which objects produce a certain effect; it becomes hard to establish if the method or the researcher affects the situation so as to improve it. Achieving validity is of paramount importance, and it must be preceded by verification that ensures the tools functions in and of themselves.

In this research, verification has been conducted by developing and using the DPM support tool, as well as existing tools (such as a PLM system), to model an array of product systems formally designed by the case companies. In terms of validation, these models have been iteratively shown and discussed with the case companies, confirming they were relevant and correct. Based on the success criteria developed and evaluated in the ChaSE project, the approach supported the designer by reusing knowledge and increased the possibility of reducing lead times, indicating an overall positive impact. The DPA has been applied to a total of six companies, which is to be kept in mind when performing generalizations. The number of case companies affects the number of products used when modelling the DPA. This factor needs attention regarding how product platform constructs are identified and modelled in terms of, for example, the level to which a structure can be generically described. This thesis has not focused on or investigated the trade-off between the level of administration and the usefulness of the DPA. The DPA has been tested on parts of PD projects, which have left certain product lifecycle steps for future research. However, to further strengthen the validity of this research, it can be discussed in the light of Olesen's (1992) five criteria*: internal logic, truth, acceptance, applicability*, and *novelty value*.

Internal logic

This criterion refers to the execution and the theoretical base that the research and research process have been founded upon. The research process builds upon the DRM framework but also integrates other research methodologies. These methods have been evaluated and applied by many others and can, therefore, be said to have validity in themselves. The execution of the methods has been supported by conducting a well-managed PhD project, consisting of several phases and gates to guarantee quality according to a university standard. The three projects, running concurrently, have had detailed project plans and have been executed with weekly meetings to guarantee project progression and delivery. Furthermore, this research is founded upon and supported by systematic literature reviews in several fields, which have created the base of the final DPA. This implies that the result is based on well-grounded theories within the studied area.

Truth

Due to companies' participation throughout the process, the research has related at every turn to the companies' realities. The continuous feedback from the case companies at the different phases of the project helps to guarantee industrial significance to the work. By using different techniques throughout the descriptive phases, real phenomena have been described as far as possible. In the prescriptive phases, real product and process data have been used to apply the DPA and iteratively been demonstrated and discussed by the companies involved. By the development of demonstrators, the DPA has been formalized and realized, which has supported the validation of the approach. Throughout the projects, the results have been compared and discussed with respect to existing literature.

Acceptance

Acceptance is judged from both a scientific and industrial point of view. Scientific acceptance has been assured by following the review processes for conferences and journals. In this way, the global research community has demonstrated its acceptance of the appended papers, which manifest the result of this work. Within the time of this thesis, the appended papers have been presented to, discussed by, and cited by international researchers. Thus, this research has contributed to the research of others, progressing the research field. Industry acceptance has been achieved by conducting evaluations and including companies in the development of the theory. By actively be involved in the company's realities, a common understanding has been developed regarding challenges and solutions. From the presented evaluations, it is also evident that industry representatives believe in the presented approach and judge it useful.

Applicability

The applicability criterion has been fulfilled as the intended users of the support presented in this research judged its usability to be good and relevant. However, within the time

frame and scope of this thesis, it has not been possible to conduct actual measurements to demonstrate the applicability and improvement induced by the DPA. Even if such measurements were possible within the time frame, it would be difficult to establish cause and effect relationships between the measured control and the application of the DPA. The improvements enabled by the DPA are expected to take time and the real advantages will be shown in upcoming years, which is outside the time frame of this work.

Novelty value

From the research gap identified in Chapter 3, it is clear that little previous research has tackled the challenges of applying platform-based development in the ETO industry. Since this thesis's approach has been developed with the real world challenges and opportunities of the participating companies, it has brought new information to the field of research. The DPA and support tool have novelty since no previous examples have been identified that share their scope and aim or apply the same concepts. It has also been clear that the investigated companies have not applied a similar method to manage the challenges.

6.5 Discussion of the research process

Starting a PhD project is an immense task spanning over four years. Ideally, the complete project is planned in the first months, including what papers to write, where and when to publish them, and how they contribute to answering the research questions. In reality, this becomes a challenge for a number of reasons. First, when starting as a PhD student, you are less experienced than when you finish your studies. This effects the relevance of the chosen topic and research questions. Moreover, research projects are needed to finance the student, which might have agendas of their own and usually only span parts of the PhD project. The involved companies also have their own agendas, which must be aligned with the identified research gap. These agendas can, in some cases, leave the student with a small, and difficult to identify, intersection in which to conduct the work.

Fortunately, when the topic has been chosen, tools and methods can be used to guide and to guarantee a level of quality in the research work. This work has applied the DRM framework (L. T. Blessing & Chakrabarti, 2009), which supports researchers within the domain of engineering design. This framework provides overall steps and guidance regarding how to perform research in this field. Within the DRM framework, there are more tools and guidance options available than those used for this work. The main four phases of the framework tell the researcher what kind of outcome that is expected. However, they do not indicate in detail how to reach these outcomes. Several other methods and techniques need to be introduced at each stage to produce a result. For this research, traditional techniques associated with case study research have been applied in the descriptive phases. The prescriptive phase is more explorative in nature, making it hard to outline from a methodology perspective. This phase has rarely been emphasized within the field of engineering design, resulting in a too small a focus on the descriptive phases and a lack of rigor. However, development among engineering design researchers has improved the quality of research methodology over the last years. Yet with traditions still lingering, it is a challenge for researchers to know when to move from one DRM step to another and to consider something as validated or fully investigated. In general, validity is a challenge in this domain due to the nature of engineering design. Measurements are difficult to produce, and cause and effect relationships hard to establish. DRM proposes SC with associated indicators intended for measurement, which is a large step in a positive direction. However, for the DRM, significant time is needed to perform all its steps, to give the introduced support time to have effect, and to execute measurements. It is more reasonable that a PhD project would focus on e.g. two steps in order to guarantee the rigor of the result.

Within engineering design research, as with this specific work and in a similar way to action research and systems development, it is common for the researcher to take part in the studied situation. This creates opportunities for the researcher to learn from a real setting and for the industry to affect the research path. It also allows the researcher to have a direct impact on the studied situation and to put the research result into practice quickly. For example, systems development supports the research process in two ways. First, it takes advantage of the validation of concepts by realizing them in tools. Second, it gives the researcher a method to dig into real problems. By formalizing and solving problems in practical ways, such as developing software, the complexity and context of the problem is better understood. Interviews can also be used to understand the same problems, but they rely on other people's perceptions of the problem and the questions asked in the interview. From a scientific perspective, however, this fact creates the challenge of being objective and distinguishing between the effect of the introduced support and the action of the researcher. This is a challenge for all qualitative data collection techniques, but it requires extra attention in the engineering design research domain due to its prescriptive element.

It is clear that there are many areas in need of improvement within engineering design research. However, the field has travelled far and improved over the years, which is promising for the future.

7

CONCLUSIONS

This chapter states the main conclusions drawn from this work. The conclusions presented here are the final statements that emerge from the previous discussion of the result.

Based on the result, there is both an interest and a need in the investigated companies to engage in platform-based development and utilization in an ETO context. This thesis has shown that traditional component-based product platform definitions are either not applicable or too abstract to be directly applied in the ETO environment. However, the DPA has been shown to enable platform-based development while also making use of the diverse assets that already exist in a company. This aspect lies largely within the contributions that have been made to the industry. This work's scientific contribution has been to expand the body of knowledge regarding the use of product platforms in companies for which component-based product platforms have been difficult, or even impossible, to implement. The presented DPA engages with issues described in the literature regarding the ability of existing product platforms to evolve and manage heterogeneous content. The evaluation of the DPA has enriched the existing literature on product platforms within the area of engineering design.

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

Conclusions regarding the need, challenge, and prerequisites of product platforms in the ETO industry

- Little research has investigated how ETO companies can develop and apply a product platform that fits their context, and even less research presents possible approaches and applications.
- The investigated companies all wish to use a platform-based approach but succeed to different extents.
- The investigated companies were all faced with changing requirements during the scoping, quotation, and subsequent development processes.
- The investigated companies possess different assets that could be used in scoping, quotation, and product development. They would like to reuse appropriate assets but are currently not doing so in a structured manner.
- The companies experience challenges in obtaining 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.

Conclusions regarding the DPA

- In order to realize and apply a product platform consisting of assets (i.e. knowledge, people etc.), as defined by (Robertson & Ulrich, 1998), a model and working approach are required.
- The DPA shows great promise as a way for ETO companies to gain the benefits of a platform-based approach, both as a way to develop the resources residing in the company and as a formalized approach that can be supported by IT applications. This conclusion is supported by the DPA's successful application in six case companies and the support tool development in five companies.
- The DPA has proven useful for mapping a company's current state, future target conditions, difficulties, and limitations.
- The construct types belonging to the DP model have proven useful and applicable to a significant amount of the company assets that have been identified in this thesis, both new and existing ones.
- The generic product items and generalized processes have been shown to provide a way to model generic product concepts and processes, to which engineering assets can be linked.
- Relationships between assets have been shown to be assets in and of themselves. They should be identified and modelled to support reuse.
- The activities associated with the expansion and use of the DP model support the identification, development, and management of engineering assets and the DPA as a whole.

Conclusions regarding support for the DPA

- The DPM has been shown to be a way to realize, in part, the DPA. The tool can manage the DP model by creating GPIs, variants, and assets that are stored in the company's PLM system database.
- The possibility of supporting the DPA using a PLM system has shown to be feasible.
- The DPA 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, user interface, and level of automation regarding data input.

7.1 Future work

Future work will focus on further development, refinement, and implementation of the DPA at more companies. Additional exemplifications are expected to increase the applicability and validity of the approach. Further exemplifications in different PLM software are needed to fully evaluate the approach and to identify blank spots in the PLM systems on the market. Future work should also focus on the following:

- When formalizing knowledge, as proposed by the DPA, the possibility of automating some of the knowledge increases. Therefore, design automation is a promising field to be integrated into the DPA. From a software perspective, there is a research gap concerning the integration of design automation into PLM systems, which could significantly streamline business.
- A focus should also be placed on visualizing the DPA to make its content and abilities clearer to its users. This is crucial as the amount of information in the model grows. For this purpose, filtering, graph and layout techniques, along with smart algorithms, should be investigated.
- Similarly, when the DP model grows, maintenance becomes more complex. To keep track of what information that is valid and applicable requires methods supported by digital tools. Strategies to identify obsolete knowledge need to be developed, and a suitable organization with specific ownerships of the DP model content needs to be created.
- Sensitivity analyses can be further implemented in software systems that support the DPA. A changed requirement can, for example, have severe effects on the production system, which can be hard to foresee. However, with a carefully created DP, these effects can be tracked and simulated in order to propose methods and countermeasures to deal with change.
- The advent of additive manufacturing creates opportunities for companies. The integration of additive manufacturing as the main manufacturing method of a specific part in the DPA allows for high customization, short delivery time for that specific part, and the possibility to offer an extreme level of customization. This can allow a company to integrate different business models for each delivered product by standardizing some parts of a product and using additive manufacturing for others.
- The main aim of a product platform is increased efficiency by sharing assets over product variants and thus reducing the need for resources in terms of, for example, engineering time and production equipment. Another area that focuses on reduced resource utilization is sustainability. Therefore, an investigation is needed that focuses on the potential of product platforms to deliver more sustainable products.

REFERENCES

Alizon, F., Shooter, S., & Simpson, T. (2005). Introduction of the REUSE Method: Retrieving Knowledge From Existing Product Designs. Paper presented at the ASME 2005 International Mechanical Engineering Congress and Exposition.

Almefelt, L., Berglund, F., Nilsson, P., & Malmqvist, J. (2006). Requirements management in practice: findings from an empirical study in the automotive industry. Research in engineering design, 17(3), 113-134.

Andersson, F. (2003). The Dynamics of Requirements and Product Concept Management A Product Modelling Approach (Doctoral dissertation). Chalmers University of Technology, Göteborg, Sweden.

Arvidsson, M., & Gremyr, I. (2008). Principles of robust design methodology. Quality and Reliability Engineering International, 24(1), 23-35.

Avison, D., Davison, R., & Malaurent, J. (2017). Information systems action research: Debunking myths and overcoming barriers. Information & Management, 55(2) 177-187.

Baxter, D., Gao, J., Case, K., Harding, J., Young, B., Cochrane, S., & Dani, S. (2007). An engineering design knowledge reuse methodology using process modelling. Research in engineering design, 18(1), 37-48.

Blessing, L. (2003). What is this thing called Design Research? Paper presented at the DS 31: Proceedings of ICED 03, the 14th International Conference on Engineering Design, Stockholm.

Blessing, L. T., & Chakrabarti, A. (2009). DRM, a design research methodology. Berlin: Springer.

Brière-Côté, A., Rivest, L., & Desrochers, A. (2010). Adaptive generic product structure modelling for design reuse in engineer-to-order products. Computers in Industry, 61(1), 53-65.

Browning, T. R. (2015). Design structure matrix extensions and innovations: a survey and new opportunities. Paper presented at IEEE Transactions on Engineering Management.

Bruun, H. P. L., Mortensen, N. H., Harlou, U., Wörösch, M., & Proschowsky, M. (2015). PLM system support for modular product development. Computers in Industry, 67, 97-111.

Brydon-Miller, M., Greenwood, D., & Maguire, P. (2003). Why action research? In: Sage Publications.

Burstein, F., & Gregor, S. (1999). The systems development or engineering approach to research in information systems: An action research perspective. Paper presented at the Proceedings of the 10th Australasian Conference on Information Systems.

Cabigiosu, A., Zirpoli, F., & Camuffo, A. (2013). Modularity, interfaces definition and the integration of external sources of innovation in the automotive industry. Research Policy, 42(3), 662-675.

Checkland, P., & Holwell, S. (1998). Action research: its nature and validity. Systemic Practice and Action Research, 11(1), 9-21.

Christ, A., Wenzel, V., Faath, A., & Anderl, R. (2013). Integration of Feature Templates in Product Structures Improves Knowledge Reuse. Paper presented at the Proceedings of the World Congress on Engineering and Computer Science.

Claesson, A., Rosvall, B. r., & Johannesson, H. (2005). Capturing design knowledge of robust designs using configurable components. Paper presented at the ASME 2005 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference.

Clarkson, P. J., Simons, C., & Eckert, C. (2004). Predicting Change Propagation in Complex Design. Journal of Mechanical Design, 126(5), 788-797. doi:10.1115/1.1765117

Clausing, D. (1996). Total Quality Development. A step-by-step guide to world-class concurrent engineering, 1994. American Society of Mechanical Engineers, Cambridge Massachusetts.

Cohen, H., S. Keller, D. Streeter. (1979). The transfer of technology from research to development. Research Management 22, 11-17.

Cooper, R. G. (2006). Managing technology development projects. Research Technology Management; 49(6), 23-31.

Corin Stig, D. (2015). Technology Platforms: Organizing and Assessing Technological Knowledge to Support its Reuse in New Applications. Chalmers University of Technology, Göteborg, Sweden.

Corin Stig, D., & Bergsjö, D. (2011). Means for internal knowledge reuse in pre-development–the technology platform approach. Paper presented at the DS 68-6: Proceedings of the 18th International Conference on Engineering Design (ICED 11), Impacting Society through Engineering Design, Vol. 6: Design Information and Knowledge, Lyngby/Copenhagen, Denmark, 15.-19.08. 2011.

Corin Stig, D., U. Högman, D. Bergsjö. (2011). Assessment of Readiness for internal Technology Transfer - A Case Study. 21st Annual incose international symposium.

Duffy, A., & Andreasen, M. (1995). Enhancing the evolution of design science. Paper presented at the Proceedings of ICED.

Eisenhardt, K. M., & Graebner, M. E. (2007). Theory building from cases: Opportunities and challenges. The Academy of Management Journal, 50(1), 25-32.

Elgh, F. (2010). Knowledge Modelling and Analysis in Design Automation Systems for Product Configuration. Paper presented at the NordDesign 2010: proceedings of the 8th biannual conference NordDesign 2010.

Elgh, F. (2012). Decision support in the quotation process of engineered-to-order products. Advanced Engineering Informatics, 26(1), 66-79.

Elgh, F. (2013). Challenge Fluctuating and Conflicting Requirements by Set-Based Engineering - ChaSE (project application dnr JTH 2015/2963-51). Jönköping University, Jönköping.

Elgh, F., & Cederfeldt, M. (2008). Cost-based producibility assessment: analysis and synthesis approaches through design automation. Journal of Engineering Design, 19(2), 113-130.

Elgh, F., & Poorkiany, M. (2012). Supporting traceability of design rationale in an automated engineer-to-order business model. Paper presented at the DS 70: Proceedings of DESIGN 2012, the 12th International Design Conference, Dubrovnik, Croatia.

Engwall, M. (2004). Produktutveckling bortom kunskapens gränser: mot en osäkerhetens grammatik. Lund, Sweden: Studentlitteratur.

Ericsson, A., & Erixon, G. (1999). Controlling design variants: modular product platforms: Society of Manufacturing Engineers.

Erixon, G. (1998). Modular function deployment: a method for product modularisation: Royal Inst. of Technology, Department of Manufacturing Systems, Assembly Systems Division.

Fernandes, J., Henriques, E., Silva, A., & Moss, M. A. (2015). Requirements change in complex technical systems: an empirical study of root causes. Research in engineering design, 26(1), 37-55.

Fogliatto, F. S., da Silveira, G. J., & Borenstein, D. (2012). The mass customization decade: An updated review of the literature. International Journal of Production Economics, 138(1), 14-25.

Forza, C., & Salvador, F. (2002). Managing for variety in the order acquisition and fulfilment process: The contribution of product configuration systems. International Journal of Production Economics, 76(1), 87-98.

Fredriksson, P., & Gadde, L.-E. (2005). Flexibility and rigidity in customization and build-to-order production. Industrial Marketing Management, 34(7), 695-705.

Giesberts, P. M., & Tang, L. V. D. (1992). Dynamics of the customer order decoupling point: impact on information systems for production control. Production Planning & Control, 3(3), 300-313.

Guðlaugsson, T. V., Ravn, P. M., Mortensen, N. H., & Sarban, R. (2014). Front-end conceptual platform modeling. Concurrent Engineering, 22(4), 267-276.

Guerrero, H. H. (1985). The effect of various production strategies on product structures with commonality. Journal of Operations Management, 5(4), 395-410.

Halbleib, H. (2004). Requirements management. Information Systems Management, 21(1), 8-14.

Halman, J. I. M., Hofer, A. P., & Vuuren, W. V. (2003). Platform-driven development of product families - Linking theory with practice. Journal of product innovation management 20(2), 149-162.

Hansen, B. L. (2003). Development of industrial variant specification systems (Doctoral dissertation): Technical University of Denmark.

Hart, C. W. (1995). Mass customization: conceptual underpinnings, opportunities and limits. International Journal of Service Industry Management, 6(2), 36-45.

Haug, A., Hvam, L., & Mortensen, N. H. (2012). Definition and evaluation of product configurator development strategies. Computers in Industry, 63(5), 471-481.

Heikkinen, T., Johansson, J., & Elgh, F. (2016). Assessment of Simulation Ready CAD Models in a Set Based Concurrent Engineering Context. Paper presented at the DESIGN 2016, 14th International Design Conference.

Helo, P. T. (2006). Product configuration analysis with design structure matrix. Industrial Management & Data Systems, 106(7), 997-1011.

Holmdahl, L. (2010). Lean Product Development På Svenska. Gothenburg: ABC-Tryck AB.

Horvath, I. (2004). A treatise on order in engineering design research. Research in engineering design, 15(3), 155-181.

Hsuan Mikkola, J., & Skjøtt-Larsen, T. (2004). Supply-chain integration: implications for mass customization, modularization and postponement strategies. Production Planning & Control, 15(4), 352-361.

Huang, Y., Jiang, Z., Liu, L., Song, B., & Han, L. (2015). Building a knowledge map model situated in product design. International Journal of Information Technology and Management, 14(1), 76-94.

Hubka, V., & Eder, W. E. (2012). Design science: introduction to the needs, scope and organization of engineering design knowledge. Berlin, Springer Science & Business Media.

Hundal, M. (2012). Product Design Variants and Customization: Dealing With Conflicting Requirements. Paper presented at the ASME 2012 11th Biennial Conference on Engineering Systems Design and Analysis.

Hvam, L., Mortensen, N. H., & Riis, J. (2008). Product customization: Berlin, Springer Publishing Company.

Hvam, L., Pape, S., & Nielsen, M. K. (2006). Improving the quotation process with product configuration. Computers in Industry, 57(7), 607-621.

Högman, U. (2011). Processes and Platforms Aligned with Technology Development - The perspective of a supplier in the Aerospace Industry. Göteborg: Chalmers University of Technology.

Högman, U., Bergsjö, D., Anemo, M., & Persson, H. (2009). Exploring the potential of applying a platform formulation at supplier level-The case of Volvo Aero Corporation. Paper presented at the DS 58-4: Proceedings of ICED 09, the 17th International Conference on Engineering Design, Vol. 4, Product and Systems Design, Palo Alto, CA, USA, 24.-27.08. 2009.

Högman, U., H. Johannesson. (2013). Applying stage-gate processes to technology development - Experiences from six hardware-oriented companies. Journal of Engineering and Technology Management, 30(3), 264-287.

Hölttä-Otto, K., Chiriac, N., Lysy, D., & Suh, E. S. (2014). Architectural decomposition: the role of granularity and decomposition viewpoint. In Advances in Product Family and Product Platform Design (pp. 221-243): Springer, New York, NY.

Jiao, J. R., & Chen, C.-H. (2006). Customer requirement management in product development: a review of research issues. Concurrent Engineering, 14(3), 173-185.

Johannesson, H. (2014). Emphasizing reuse of generic assets through integrated product and production system development platforms. In Advances in Product Family and Product Platform Design (pp. 119-146): Springer, New York, NY.

Johannesson, H., Landahl, J., Levandowski, C., & Raudberget, D. (2017). Development of product platforms: Theory and methodology. Concurrent Engineering 25(3) 195-211.
Johansson, J., André, S., & Elgh, F. (2015). Simulation Ready CAD-Models as a Means for Knowledge Transfer Between Technology Development and Product Development. Paper presented at the DS 80-6 Proceedings of the 20th International Conference on Engineering Design (ICED 15) Vol 6: Design Methods and Tools-Part 2 Milan, Italy, 27-30.07. 15.
Kock, N. (2007). Information systems action research: An applied view of emerging concepts and methods (Vol. 13): Springer Science & Business Media, Laredo, Texas, USA.
Kossiakoff, A., Sweet, W. N., Seymour, S., & Biemer, S. M. (2011). Systems engineering principles and practice (Vol. 83): Hoboken, John Wiley & Sons.
Krause, D., Beckmann, G., Eilmus, S., Gebhardt, N., Jonas, H., & Rettberg, R. (2014). Integrated development of modular product families: a methods toolkit. In Advances in Product Family and Product Platform Design (pp. 245-269): Springer, New York, NY, USA.
Lakemond, N., Johansson, G., Magnusson, T., Säfsten, K. (2007). Interface between technology development, product development and production - Critical factors and a conceptual model. International Journal of Technology Intelligence and Planning, 56(1), 317-330.
Land, F. (1982). Adapting to changing user requirements. Information & Management, 5(2), 59-75.
Lange, M. W., & Imsdahl, A. (2014). Modular function deployment: using module drivers to impart strategies to a product architecture. In Advances in Product Family and Product Platform Design (pp. 91-118): Springer, New York, NY, USA.
Lee, J., Deng, W.-Y., Lee, W.-T., Lee, S.-J., Hsu, K.-H., & Ma, S.-P. (2010). Integrating process and work breakdown structure with design structure matrix. Journal of Advanced Computational Intelligence and Intelligent Informatics, 7, 8.
León, H. C. M., & Farris, J. A. (2011). Lean Product Development Research: Current State and Future Directions. Engineering Management Journal, 23(1) 29-51.
Levandowski. (2014). Platform Lifecycle Support using Set-Based Concurrent Engineering: Chalmers University of Technology, Göteborg, Sweden.
Levandowski, C., Forslund, A., Söderberg, R., & Johannesson, H. (2012). Platform Strategies from a PLM Perspective–Theory and Practice for the Aerospace Industry. Paper presented at the 53rd Structures, Structural Dynamics, and Materials Conference (SDM).
Levandowski, C., Raudberget, D., & Johannesson, H. (2014). Set-Based Concurrent Engineering for Early Phases in Platform Development. Paper presented at the The 21st ISPE International Conference on Concurrent Engineering-CE2014.
Levandowski, C., Stig, D. C., Raudberget, D., & Johannesson, H. (2015). Accommodating Emerging Technologies in Existing Product Platforms. Paper presented at the 24th International Association for Management of Technology Conference (IAMOT 2015).
Levandowski, C. E., Jiao, J. R., & Johannesson, H. (2015). A two-stage model of adaptable product platform for engineering-to-order configuration design. Journal of Engineering Design, 26(7-9), 220-235.
Li, Z., Cheng, Z., Feng, Y., & Yang, J. (2013). An integrated method for flexible platform modular architecture design. Journal of Engineering Design, 24(1), 25-44.
Lyu, G., Chu, X., & Xue, D. (2017). Product modeling from knowledge, distributed computing and lifecycle perspectives: A literature review. Computers in Industry, 84, 1-13.
Malmqvist, J. (2002). A classification of matrix-based methods for product modeling. Paper presented at the DS 30: Proceedings of DESIGN 2002, the 7th International Design Conference, Dubrovnik.
Mankins, J. C. (1995). Technology readiness levels. A white paper, Washington, DC: Natianl Aerospace and Space Administration.
McGrath, M. E. (1995). Product strategy for high-technology companies. New York: Irwin professional publishing.

Meyer, M. H., & Lehnerd, A. P. (1997). The power of product platforms - Building value and cost leadership. New York: The Free Press.

Meyer, M. H., Osiyevskyy, O., Libaers, D., & van Hugten, M. (2017). Does Product Platforming Pay Off? Journal of product innovation management, 35(1) 66-87

Michael, S.-K., Kilian, G., & Luciënne, B. (2007). Managing Requirements or being Managed By Requirements-Results of An Empirical Study. Proceedings of the 16st International Conference on Engineering Design (ICED 07) Cite des sciences et de l'industrie, Paris, France.

Mortensen, N. H., Hvam, L., & Haug, A. (2010). Modelling Product Families for Product Configuration Systems with Product Variant Master and CRC-cards. Paper presented at the ECAI 2010 Workshop on Intelligent Engineering Techniques for Knowledge Bases (IKBET).

Mortensen, N. H., & Lřkkegaard, M. (2017). Good product line architecture design principles. Paper presented at the DS 87-3 Proceedings of the 21st International Conference on Engineering Design (ICED 17) Vol 3: Product, Services and Systems Design, Vancouver, Canada, 21-25.08. 2017.

Nilsson, P. (2004). Managing requirements in a design support system : a product modelling approach. Chalmers University of Technology, Götebrog, Sweden.

Nilsson, P., & Jackson, M. (2004). Managing Production Requirements within Product Development. Paper presented at the Proceedings of the TMCE.

Nobelius. (2002). Managing R&D Processes - Focusing on Technology Development, Product Development and their Interplay. (Doctoral dissertation) Göteborg: Chalmers University of Technology.

Nobelius. (2004). Linking product development to applied research: transfer experiences from an automotive company. Technovation, 24(4) 321-334.

Nobelius, D. (2004). Linking product development to applied research: transfer experiences from an automotive company. Technovation, 321-334.

Nunamaker Jr, J. F., Chen, M., & Purdin, T. D. (1990). Systems development in information systems research. Journal of management information systems, 7(3), 89-106.

Olesen, J. (1992). Concurrent Development in Manufacturing-based on dispositional mechanisms. 1992. The Technical University of Denmark: Delft.

Otto, K., Hölttä-Otto, K., Simpson, T. W., Krause, D., Ripperda, S., & Moon, S. K. (2016). Global Views on Modular Design Research: Linking Alternative Methods to Support Modular Product Family Concept Development. Journal of Mechanical Design, 138(7), 071101.

Pahl, G., & Wallace, K. (2007). Engineering design : a systematic approach. Berlin ;: Springer.

Pakkanen, J., Juuti, T., & Lehtonen, T. (2016). Brownfield Process: A method for modular product family development aiming for product configuration. Design Studies, 45, 210-241.

Park, J., Shin, D., Insun, P., & Hyemi, H. (2008). A product platform concept development method. Journal of Engineering Design, 19(6), 515-532.

Pirmoradi, Z., Wang, G. G., & Simpson, T. W. (2014). A review of recent literature in product family design and platform-based product development. In [Timothy W. Simpson, Jianxin (Roger) Jiao, Zahed Siddique, Katja Hölttä-Otto] (eds) Advances in product family and product platform design (pp. 1-46): Springer, New York, NY, USA .

Popper, K. (2005). The logic of scientific discovery: Routledge. New York, NY, USA.

Preston, S., Chapman, C., Pinfold, M., & Smith, G. (2005). Knowledge acquisition for knowledge-based engineering systems. International Journal of Information Technology and Management, 4(1), 1-11.

Raudberget, D. (2012). Industrial Experiences of Set-Based Engineering - Effects, results and applications. Göteborg: Chalmers reproservice.

Ravn, P. M., Gudlaugsson, T. V., & Mortensen, N. H. (2015). A multi-layered approach to product architecture modeling: Applied to technology prototypes. Concurrent Engineering 24(1) 3-16.

Regli, W. C., Hu, X., Atwood, M., & Sun, W. (2000). A survey of design rationale systems: approaches, representation, capture and retrieval. Engineering with computers, 16(3-4), 209-235.

Robertson, D., & Ulrich, K. (1998). Planning for product platform. Sloan management review, 39(4) 19-31.

Rothwell, R., & Gardiner, P. (1990). Robustness and product design families. Design Management: A Handbook of Issues and Methods,(Oakley, M., ed.). In: Basil Blackwell Inc., Cambridge, MA.

Sawhney, M. S. (1998). Leveraged high-variety strategies: from portfolio thinking to platform thinking. Journal of the Academy of Marketing Science, 26(1), 54-61.

Siddique, Z., & Boddu, K. R. (2004). A mass customization information framework for integration of customer in the configuration/design of a customized product. AI EDAM, 18(1), 71-85.

Simpson, T. W. (2004). Product platform design and customization: status and promise. Artificial Intelligence for Engineering Design, Analysis and Manufacturing, 18(1), 3-20.

Simpson, T. W., Jiao, J., Siddique, Z., & Hölttä-Otto, K. (2014). Advances in Product Family and Product Platform Design: Springer, New York, NY, USA.

Simpson, T. W., Siddique, Z., & Jiao, J. (2006). Product platform and product family design - Methods and application. New York: Springer science+Business media, inc.

Simpson, T. W., Z. Siddique, J. Jiao. (2006). Product platform and product family design - Methods and application. New York: Springer science+Business media, inc.

Sousa, R., & Voss, C. A. (2001). Quality management: universal or context dependent? Production and Operations Management, 10(4), 383-404.

Squire, B., Brown, S., Readman, J., & Bessant, J. (2006). The Impact of Mass Customisation on Manufacturing Trade-offs. Production and Operations Management, 15(1), 10-21.

Stechert, C., & Franke, H.-J. (2009). Managing requirements as the core of multi-disciplinary product development. CIRP Journal of Manufacturing Science and Technology, 1(3), 153-158.

Stjepandić, J., Ostrosi, E., Fougères, A.-J., & Kurth, M. (2015). Modularity and supporting tools and methods. In Concurrent Engineering in the 21st Century (pp. 389-420): Springer eBook.

Stokes, M. (2001). Managing engineering knowledge: MOKA: methodology for knowledge based engineering applications: London Professional Engineering Publ.

Stolt, R., Johansson, J., André, S., Heikkinen, T., & Elgh, F. (2016). How to Challenge Fluctuating Requirements: Results from Three Companies. Paper presented at the Proceedings of the 23rd ISPE Inc. International Conference on Transdisciplinary Engineering, Parana, Curitiba, October 3–7, 2016.

Suh, E. S., De Weck, O. L., & Chang, D. (2007). Flexible product platforms: framework and case study. Research in engineering design, 18(2), 67-89.

Sun, Z., & Liu, J. (2008). Reusing design process based on cognitive knowledge templates. Paper presented at the 38th International Conference on Computers and Industrial Engineering 2008, New York, NY: IEEE.

Sutinen, K., Almefelt, L., & Malmqvist, J. (2000). Implementation of requirements traceability in systems engineering tools. Paper presented at the Proceedings of Product Models 2000, Linköping, Sweden.

Swaminathan, J. M. (2001). Enabling Customization Using Standard Operations. California management review, 43(3) 125-135.

Säfsten, K., Johansson, G., Lakemond, N., & Magnusson, T. (2014). Interface challenges and managerial issues in the industrial innovation process. Journal of Manufacturing Technology Management, 25(2), 5-5.

Tang, D., Zhu, R., Tang, J., Xu, R., & He, R. (2010). Product design knowledge management based on design structure matrix. Advanced Engineering Informatics, 24(2), 159-166.

Thomke, S., & Reinertsen, D. (1998). Agile Product Development: MANAGING DEVELOPMENT FLEXIBILITY IN UNCERTAIN ENVIRONMENTS. California management review, 41(1) 8-30.

Tseng, M. M., Jiao, J., & Merchant, M. E. (1996). Design for mass customization. CIRP Annals-Manufacturing Technology, 45(1), 153-156.

Tuli, P., & Shankar, R. (2015). Collaborative and lean new product development approach: a case study in the automotive product design. International Journal of Production Research, 53(8), 2457-2471.

Ulrich, K. (1995). The role of product architecture in the manufacturing firm. Research Policy, 24(3), 419-440.

Ulrich, K. T., & Eppinger, S. D. (2012). Product Design and Development, 2012. In: McGraw-Hill, New York, NY.

Verhagen, W. J., Bermell-Garcia, P., van Dijk, R. E., & Curran, R. (2012). A critical review of Knowledge-Based Engineering: An identification of research challenges. Advanced Engineering Informatics, 26(1), 5-15.

Wikner, J., & Rudberg, M. (2005). Integrating production and engineering perspectives on the customer order decoupling point. International Journal of Operations & Production Management, 25(7), 623-641.

Williamson, K. (2002). Research methods for students, academics and professionals: Information management and systems: Amsterdam Elsevier.

Yin, R. K. (2014). Case study research : design and methods (5. ed.. ed.). London: SAGE.

Zhang, L. L. (2015). A literature review on multitype platforming and framework for future research. International Journal of Production Economics, 168, 1-12.

APPENDED PAPERS

The following papers constitute the foundation of this thesis

Paper A Samuel André, Roland Stolt, Fredrik Elgh, Joel Johansson, Morteza Poorkiany (2014). *Managing Fluctuating Requirements by Platforms Defined in the Interface Between Technology and Product Development.* Proceedings of the 21st ISPE International Conference on Concurrent Engineering, 8-11 September, Beijing, China.

Paper B Samuel André, Roland Stolt, Fredrik Elgh (2015). *Introducing Design Descriptions on Different Levels of Concretization in a Platform Definition.* Proceedings of 12th IFIP WG 5.1 International Conference, PLM 2015, October 19-21, Doha, Qatar.

Paper C Samuel André, Roland Stolt, Fredrik Elgh (2016). *A Platform Model for Suppliers of Customized Systems—Creating an Ability to Master Fluctuating Requirements*. Proceedings of ASME IDETC/CIE International Design Engineering Technical Conferences & Computers & Information in Engineering Conference, 21-24 August, Charlotte, North Carolina, USA.

Paper D Samuel André, Fredrik Elgh, Joel Johansson, Roland Stolt (2017). *The Design Platform—a Coherent Platform Description of Heterogeneous Design Assets for Suppliers of Highly Customized Systems.* Journal of Engineering Design, 28(10-12), 599-626.

Paper E Samuel André, Fredrik Elgh (2018). *Modeling of Transdisciplinary Engineering Assets Using the Design Platform Approach for Improved Customization Ability.* Journal of Advanced Engineering Informatics, 38, 277-290.

Paper F Samuel André, Fredrik Elgh (2019). *Supporting the Modelling and Managing of Relations in the Design Platform*. Proceedings the 22th International Conference on Engineering Design (ICED), 5-8 August, Delft, The Netherlands.

Paper G Samuel André, Martin Lennartsson, Fredrik Elgh (2019). *PLM support for the Design Platform in industrialized housing for efficient design and production of customized houses.* Submitted to a journal.

.

PAPER A

Managing Fluctuating Requirements by Platforms Defined in the Interface Between Technology and Product Development

Samuel André, Roland Stolt, Fredrik Elgh, Joel Johansson, & Morteza Poorkiany

Proceedings of the 21st ISPE International Conference on Concurrent Engineering, Beijing, China, 8-11 September 2014.

PAPER B

Introducing Design Descriptions on Different Levels of Concretization in a Platform Definition

Samuel André, Roland Stolt, & Fredrik Elgh

Proceedings of the 12th IFIP WG 5.1 International Conference, PLM 2015, Doha, Qatar, 19-21 October 2015.

PAPER C

A Platform Model for Suppliers of Customized Systems—Creating an Ability to Master Fluctuating Requirements

Samuel André, Roland Stolt, & Fredrik Elgh

Proceedings of ASME International Design Engineering Technical Conferences & Computers and Information in Engineering Conference IDETC/CIE, Charlotte, North Carolina, USA, 21-24 August 2016.

PAPER D

The Design Platform—a Coherent Platform Description of Heterogeneous Design Assets for Suppliers of Highly Customized Systems

Samuel André, Fredrik Elgh, Joel Johansson, & Roland Stolt

Journal of Engineering Design, 28(10-12), 599-626, (2017).

PAPER E

Modeling of Transdisciplinary Engineering Assets Using the Design Platform Approach for Improved Customization Ability

Samuel André & Fredrik Elgh

Journal of Advanced Engineering Informatics, 38 , 277-290, (2018).

PAPER F

Supporting the Modelling and Managing of Relations in the Design Platform

Samuel André & Fredrik Elgh

Proceedings of the 22th International Conference on Engineering Design (ICED), Delft, The Netherlands, 5-8 August 2019.

PAPER G

PLM support for the Design Platform in industrialized housing for efficient design and production of customized houses

Samuel André, Martin Lennartsson, & Fredrik Elgh

Submitted to a journal.