PHD

Transitioning towards modular system development

Heilemann, Markus

Award date:
2016

Awarding institution:
University of Bath

Link to publication

Transitioning towards modular system development

Markus Heilemann

A thesis submitted for the degree of Doctor of Philosophy

University of Bath

Department of Mechanical Engineering

April 2016

Table of contents

List of figures

List of tables

Acknowledgements

I would like to express my very great appreciation to my academic supervisor Prof. Steve Culley for his guidance, advice, encouragement, constructive criticism, patience and support throughout this research work. It was a great pleasure and an enriching experience working with him. Our meetings always exceeded my expectations. I always left them highly motivated and well-prepared to tackle the next research challenge, even during tough times. I could not have completed this thesis without his brilliant support.

My gratitude also goes to the case companies in industry, particularly to the primary case company (which is kept anonymous), for giving the possibility to obtain intriguing insights into the researched field and for making this research possible. Special thanks go to my former industrial supervisor and colleagues for their considerable contribution to work conducted in the industrial field and for their support.

I acknowledge the support of the members from the Department of Mechanical Engineering, the Faculty of Engineering & Design, the researchers from the Innovative Design and Manufacturing Research Centre (IdMRC), those who contributed to the PGSkills courses and all the other supporting staff and members from the University of Bath. Thank you for exchanging experiences, giving valuable input and lending a helping hand when needed during my visits to the university and beyond.

Fruitful discussions with international researchers of the engineering research community have inspired this research, both, from a topic point of view and from a general research point of view. Thanks to the research community.

I would also like to thank my family for all their support and patience.

Abstract

Modular product architectures play a key role for profitable product life cycles in a wide variety of industries. However, engineering design research to date has tended to focus on the issues and the approaches associated with establishing modular product architectures on new products and systems. Little research has been undertaken on the overall issue of what will be referred to as "modularisation transition". It is thus the main aim of this work to identify and test issues and factors associated with support for transitioning from single product development towards development of modular systems. Based on these findings, it is proposed to develop understanding and engineering design support for the transition.

The research is based on a multi-faceted longitudinal case study with the involvement of a primary case company and 27 complementary secondary cases from different industries in eight countries. The research was conducted during an overall period of five years and included research methods such as a participant observer approach for generating a deep understanding and an action-based interventionist approach for developing support.

One of the main issues that the case companies encountered during transitioning has been identified to be development projects not adhering to the common modular reference architecture during later phases of the modular system life cycle. Hence, the modular system is constantly in danger of losing what can be thought of as its stability because the architectures of products diverge. An evaluation has shown that there is only scarce pre-existing support for industry during these later phases. Thus, a modularisation support framework with focus on stability was created as guidance to develop new support.

As part of the support framework, a modularisation assessment framework has been developed. It is the aim of the assessment framework to ensure that important factors for the stability of the modular system are in place throughout the entire development life cycle. Further to this, modularisation metrics are presented. It is the purpose of the metrics to assess adherence of derivative products to the specifications of the common modular system architecture. Subsequently an approach for the provision of product architecture information in standard IT-systems has been developed. This approach aims at improving transparency about the modular system, for instance to protect the architecture from diverging and to enable efficient assessments. All deliverables of this thesis have been validated in a variety of ways in a variety of industrial settings.

It is suggested that this thesis unprecedentedly highlights the importance of stability for modular system development, particularly during later phases. This is seen as new and vital understanding to make transitioning a successful undertaking. Therefore, the thesis presents novel and compressed insights about issues and support throughout the whole development life cycle. Moreover, an innovative set of support for stability during later phases of the life cycle is provided. This helps companies to avoid costly setbacks during transitioning and to achieve their modularisation goals more efficiently and sustainably.

List of abbreviations

Abbreviation	Description
BOM	Bill of Materials
BSC	Balanced Scorecard
CAD	Computer-Aided Design
CAX	Computer-Aided X, where X is a placeholder for different technologies or fields like Design, Manufacturing, Engineering or Quality
CBSE	Component Based Software Engineering
CMEA	Change Mode and Effect Analysis
CMMI	Capability Maturity Model Integration
DFA	Design for Assembly
DFM	Design for Manufacturing
DFMA	Design for Manufacturing and Assembly
DfV	Design for Variety
DRM	Design Research Methodology
DSM	Design Structure Matrix
DTU	Technical University of Denmark
ERP	Enterprise Resource Planning
FMEA	Failure Mode and Effect Analysis
GBOM	Generic Bill of Materials
HVAC	Heating, Ventilation and Air Conditioning
IEEE	Institute of Electrical and Electronics Engineers
IDES	International Demonstration and Education System
IT	Information Technology
METUS	Management Engineering Tool for Unified Systems
MFD	Modular Function Deployment
MIG	Module Interface Graph

Abbreviation	Description
MIM	Module Indication Matrix
MQB	Modular Transverse Toolkit of the Volkswagen Group *from German: "Modularer Querbaukasten"*
MSDL	Modular System Development Life Cycle
MV	Module Variant
NASA	National Aeronautics and Space Administration
NPD	New Product Development
NTF	New to the Firm
PDM	Product Data Management
PFA	Product Family Architecture
PFMP	Product Family Master Plan
PLC	Product Life Cycle
PLM	Product Life Cycle Management
PMDP	Product and Module Development Projects
PMM	Product Management Map
QFD	Quality Function Deployment
RO	Research Objective
RQ	Research Question
SEI	Software Engineering Institute
SWOT	Strengths, Weaknesses, Opportunities and Threats

1 Introduction

The goal of engineering design is to develop solutions to technical problems under consideration of adjacent disciplines and constraints such as economic or organisational requirements. In order to meet that goal, engineering designers have to apply a wide reaching set of skills, knowledge, experience, techniques, methodologies and principles. One of these principles is modular design. Modular design is used to derive a wide variety of product variants while at the same time reducing cost (Pahl et al., 2007).

This well-known principle from literature and industry follows the logic of dividing a range of products into a set of interchangeable modules. If a set of modules is based on the same underlying architecture and if the modules can be combined to derive a high variety of products, it is defined *modular system* for the purpose of this work.

There has been considerable research on modularisation rationale, modularisation benefits and approaches to establish types of product architecture, see for example Chapter 6 in Smith's and Rienertsen's widely used book on developing products in half the time (Smith and Reinertsen, 1991). Much less is known about the issues associated with introducing and maintaining modular product architectures and modules across different product development projects, brands, markets or organisations, particularly in existing, successful and well established organisations and product ranges. Such an introduction, development and maintenance is termed *transition* in the course of this work. It is considered to be multi-dimensional, fraught with technical, process and operational difficulties and is also cost-intensive. To date, there are only very few studies which cover overall modularisation transition towards *stable* modular systems. For the purpose of this work, "stable" is defined as firmly established and not likely to fail, change or deteriorate (Oxford Dictionaries Online, 2016).

To remedy this, it is the focus of this research to investigate the overall issue of transitioning towards modular system development, covering a wide range of products that originally were not based on the same modular architecture. This research aims to support companies during "modularisation transition" by providing a fundamental understanding and some approaches for engineering design support.

The following sections introduce the topic by outlining the background of the research problem and by identifying the detailed research issues which are solved in the course of this research work.

1.1 Background of the research problem

High complexity has become a major problem for many industries during recent years. The main drivers for this development are individualised and fluctuating market needs, new technologies, market consolidation and internationalisation (Friedman, 2007; Lindemann, Maurer and Braun, 2009).

1.1.1 Individualised and fluctuating market needs

The market environment is characterised by, amongst other factors, diverse needs, unstable demand, uncertain forecasts, strong buyer power and the need to serve market niches with low sales volume. This market development leads companies to "produce ever greater variety more quickly" (Pine, 1992, p. 45) under low volumes and forces them to shorten development time and product life cycles (Clark and Fujimoto, 1989, p. 25). The characteristic trend of volatile and dynamic markets has been known since years and is today still a topic of significant importance (Simpson et al., 2006, p. 1; Jiao, Simpson and Siddique, 2007, p. 5–6; Simpson et al., 2014). For instance, 40 new car models are waiting in the pipeline of Mercedes to fulfil individualised customer needs until 2020 (Keller, 2015). A different example for shortened product life cycles can be found in TV industry where the primary useful life of a TV decreased from about 11 years in 2005 to about 5 years in 2012 (Stiftung Warentest, 2015).

In order to stay competitive, companies have to find means how to handle complexity caused by individualised and fluctuating market needs.

1.1.2 New technologies

The progress in different technological fields, such as automation, computerisation, connectivity or environmental engineering drives companies to offer a wider range of technologies in parallel (Pine, 1992, p. 46–47).

Firstly, new technologies applied within a product can help to extend a product's functionality. Well-known examples of this are embedded driver assistance systems in cars or integrated monitoring systems in drilling machines. These products have originally been purely mechanical systems. Today, they are highly interconnected electro-mechanical systems. A car is a good illustration for that. An estimation from 2011 quantified the number of microcontrollers in a well-equipped car to around 50. This number is applicable only for intra-car control and networking mechanisms (Fleming, 2011, p. 4). Further ubiquitous developments around the internet of things are believed to give a sharp rise in those figures again (Gubbi et al., 2013).

Secondly, for products with the same functionality, companies have to offer a wider range of product portfolio with different technologies. For instance, with rising energy prices there is and will be an increasing need to develop energy-efficient and green products (Department of Trade and Industry, 2010). For example, this means that companies such as heating, ventilation and air conditioning (HVAC) manufacturers have to expand their product portfolio. New products like reversible heat pumps, hybrid systems, fuel-cells, photovoltaic and solar thermal systems, combined-heat-and-power plants and biomass systems will in future be of greater importance. These products and systems will *add to* and *replace* current core products like oil- and gas-fired boilers (Sustainable Energy Authority Ireland, 2013). A similar situation can be found in many industries. To give a further example, automotive industry is also widening its technology portfolio by shifting from fossil fuel propulsion towards a combination with alternative technologies (Schifferes, 2007).

Companies are struggling to cope with complexity that is induced by new technology development (Koehler, Naumann and Vajna, 2014, p. 1811). In order to survive, companies have to find support that helps them to keep track of complexity in their multi-technology product portfolio.

1.1.3 Market expansion and internationalisation

Established markets are saturated with products that once contributed to their economic growth. For example, this can be seen in the case of cars (Schifferes, 2007; KPMG International Cooperative, 2013), computers (Arthur, 2012) or smartphones (Arthur, 2014). This leads to only moderate growth rates, stagnation or decline of sales volume in originally strong established markets. If companies cannot maintain their growth rates by penetrating the market, introducing unique features or new technologies, they have different growth alternatives (Ansoff, 1957): Firstly, they can induce new customer demand by extending product features offered on established markets. Secondly, they can adopt their product portfolio to establishing and emerging markets in order to become successful sellers there. This results in companies struggling with higher mix but lower volume product portfolios.

The parallel trend of globalisation and fierce worldwide competition sets companies under immense cost pressure. New market entrants from emerging markets frequently launch highly cost-efficient products while beating the price level of established competitors. Moreover, market entrants from emerging markets have been catching up in terms of quality and know-how with incredible speed (Friedman, 2007).

Therefore, companies have to find ways how to cut cost without compromising quality, functionality and variety.

1.1.4 Market and organisational consolidation

A common strategy for companies, if they want to gain market access, expand their product portfolio, open up synergies or acquire know-how about new technologies, is to buy, merge (The Huffington Post, 2013; Bosch Group, 2014) or collaborate (Mitteldeutsche Zeitung, 2014) with other companies. The negative effect of this strategy is that engineering, manufacturing and administration processes have to handle an increasing amount of *uncoordinated multi-site and multi-brand complexity* (Skold and Karlsson, 2007). The same situation can be found within companies where different country organisations or business units are working in parallel without proper integration process that consolidates synergies ("silo-mentality"). If products and engineering processes are not consolidated in such environments, companies are facing the problem of wasting resources by doing similar or almost identical things over and over again. For instance, this could be a module or an interface that is not fixed company-wide but newly defined for every development project individually again and again.

If companies do not want to lag behind their competitors, they have to remove the waste that is entailed by this complexity driver.

Once companies are faced with complexity drivers, they get into the dilemma of a vicious circle of continuously increasing complexity. For instance, if a company wants to increase revenues by entering a new market, it also has to generate new products and processes. This increases internal complexity and in turn, internal complexity cost. As the increased complexity costs lead to a loss in profit margin and to a loss in competitiveness due to higher prices, companies face the challenge of over-capacity, fierce price competition and decreasing sales. In order to overcome these problems, companies most often have no choice as to get caught by additional complexity drivers like internationalisation, new technologies or market expansion (Rennekamp, 2013; Renner, 2007; Schuh and Schwenk, 2001). Such a vicious circle of continuously increasing complexity is shown by Figure 1.

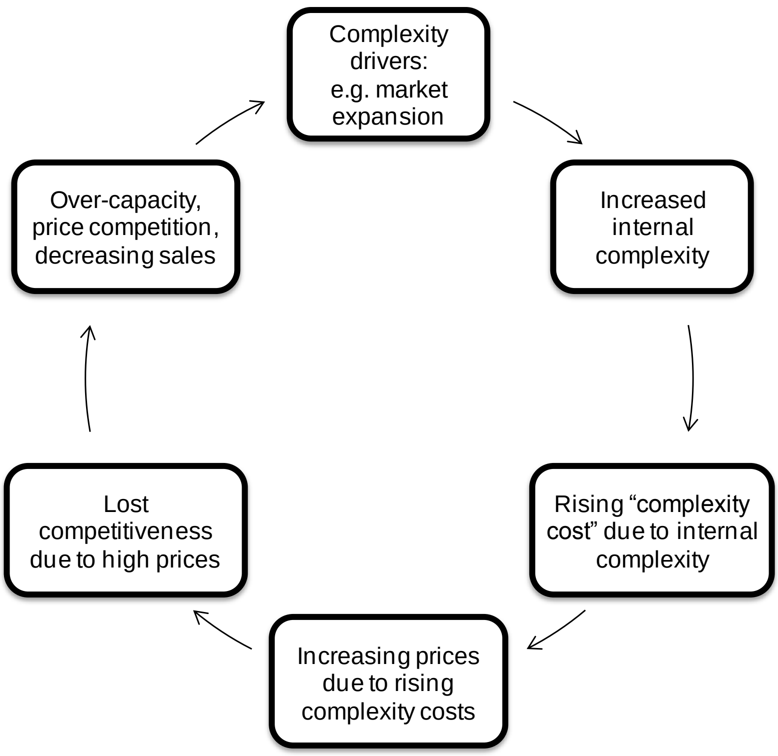


Figure 1: Vicious circle of continuously increasing complexity (Rennekamp, 2013; Renner, 2007; Schuh and Schwenk, 2001)

1.2 Increasing complexity

The given drivers have led to a tremendous increase in complexity that is imposed on products and engineering design processes (Koehler, Naumann and Vajna, 2014; Mikkola and Gassmann, 2003; Pine, 1992). Various studies regularly quantify the trend towards higher intra-company complexity. For instance, a study of the Center Automotive Research (CAR) at the University Duisburg-Essen estimated that the number of different car models offered has been increasing by more than 80% between 1995 and 2015. It is stated that there are 3281 different car variants (only differentiated by type, body design and engine) on the German market with an accelerating upward trend (Der Teckbote, 2012, p. 7). Wildemann (2005) shows an example where the number of used car body types doubled within eleven years. Renner (2007, p. 4) cites a survey presented by Stockmar (2004, p. 17) in which car manufacturers estimate variant growth of 10 % per year. Other studies show a rise in items to be handled by manufacturing companies in general industries of up to 130 % within ten years (Wildemann, 1991; Mühlbradt and Mirwald, 1992, p. 41). More recent studies show that since Ford's standard Model T was launched, the number of different car variants that can be ordered from Ford has increased to more than 3.8 million different variants of Ford vehicles, including colour, interior and optional packages (Simpson, Siddique and Jiao, 2005a, p. 1). Similar examples can be found in all kind of industries with complex technical products (Koeppen, 2008, p. 2).

The increased product and process complexity in direct and indirect organisational domains entails tremendous cost which limits competitiveness of manufacturing companies (Pine, 1992; Simpson et al., 2014). It is claimed that 15 % - 20 % of a product's overall cost are complexity-driven (Caesar, 1991; Piller and Waringer, 1999; Ripperda and Krause, 2013).

To quantify the cost of complexity in detail, researchers introduced the concept of *complexity cost*. According to a definition provided by Thonemann and Brandeau (2000, p. 1), complexity cost is "the cost of indirect functions at a company and its suppliers that are caused by component variety; complexity cost includes, for instance, the cost of designing, testing, and documenting a component variant." Each newly created component variant and the maintenance of the component variant causes process-related complexity costs of considerably more than 2000 € per part number in average (Ehrlenspiel, 2003; Wildemann, 2005; Ehrlenspiel, Kiewert and Lindemann, 2007; Eilmus, Ripperda and Krause, 2013).

Currently, there are 30 000 unique components (Ulrich and Eppinger, 2012, p. 5) and in total three million components in a Boeing 777 (Boeing, 2014). Airbus stocks 3.6 million spare parts and 150 000 tools, respectively 120 000 and 20 000 different part numbers (Airbus, 2014). Another example is given by automotive industry with 30 000 components (Toyota Motor Company, 2014) in a single car and respectively 10 000 unique part numbers (Ulrich and Eppinger, 2012, p. 5). Even smaller products like Hewlett-Packard Deskjet Printers have approximately 200 unique mechanical parts in a single product. If these figures are combined with above mentioned figures and if it is assumed that a company has to handle from several hundred to several million different product variants (3,8 million vehicle variants in the case of Ford (Simpson, Siddique and Jiao, 2005a, p. 1), it is suggested that companies waste substantial amounts of money if they do not handle their complexity in a systematic, scientific manner. For instance, saving 1 000 unique part numbers throughout the product portfolio of a company could mean cost savings of more than two million Euros, depending on characteristics of the specific company and its products.

From an overall perspective, according to the Global Simplicity Index, it is estimated that "the top 200 Fortune global companies lost $237 billion between them in 2010 because of the increasing complexity in their markets and their own organisations" (Chynoweth, 2011). Even though there is no clear cut between good and bad complexity or between unavoidable and avoidable complexity, "there is a direct link between profits and complexity" (Chynoweth, 2011).

From the findings of this section can be concluded that there is huge cost saving potential for companies if they can improve the way they deal with complexity. Hence, the next section introduces mitigating strategies to deal with increasing complexity.

1.3 Modularisation as strategy to deal with complexity

Strategies to mitigate increasing complexity in companies have been well-known in industry and literature since decades.

Henry Ford's statement "Any customer can have a car painted any colour that he wants so long as it is black." (Ford and Crowther, 2005) must not be taken literally[1] and must be seen in its textual context because Ford tailored thousands of T Model variants to individual customer needs (Alizon, Shooter and T. W. Simpson, 2009, p. 602). Nevertheless, standardised components and interchangeable parts have been crucial for the introduction of mass production and efficient assembly lines (Ford, 1926, p. 90–92).

However, standardisation alone is not sufficient to deal with increasing complexity. The importance of managing the trade-off between diverse customer needs and standardisation was already marked during the early 20th century. Ford's struggle between salespeople who "were insistent on increasing the line" (Ford and Crowther, 2005) and engineering people who saw "the advantages that a single model would bring about in production" (Ford and Crowther, 2005) led to the T Model platform from which a huge variety of product variants could be derived (Alizon, Shooter and T. W. Simpson, 2009, p. 602). Consequently, Schuh and Schwenk (2001) argue that the goal of managing complexity is not to

[1] "The Model T was introduced Oct. 1, 1908, and through the 1913 model year buyers had a choice of several colors, including black. Then, in 1926 and 1927, colors included green, light blue, brown, maroon - and, of course, black. Black was the only color the Model T came in from 1914 through 1925, and the reason was economics, not style. Black was the only color paint that could be dried quickly, and speed was important at the Ford plant because of its enormous volume." (Kurylko, 2003)

reduce complexity as much as possible. The challenge is rather to improve the balance between external variety (desired) and internal complexity (undesired).

Common strategies to balance internal complexity with external variety can be classified in different ways. Lindemann et al. (2009, p. 31–36) distinguish between “acquisition and evaluation” (e.g. representation, modelling, metrics), “avoidance and reduction” (e.g. standardisation, platforms) and “management and control” (e.g. managing the trade-off between standardisation and customisation) of complexity. Pine (1992) points out that complexity management strategies either cover development, production, marketing or delivery.

It is widely recognized that modular product architecture design is on the one hand an efficient strategy to improve the balance between standardisation and customisation and on the other hand a strategy to improve development, production, marketing and delivery performance. This is done through balancing internal complexity and external variety at the same time (Pine, 1992; Jiao, Simpson and Siddique, 2007).

The importance of modular product architecture design as an engineering design lever to improve the balance between internal complexity and external variety (see Figure 2) is widely recognized, not only in management and engineering literature, but also in industrial practice. Researchers have shown that modular architectures are, firstly, directly linked to process performance in the supply chain, in manufacturing, administration and in engineering. Secondly, they are also linked to the internal part number count and product variety which can be economically offered to the customer (Ulrich, 1995, p. 426–438; Smith and Reinertsen, 1991, p. 99–110; Jiao, Simpson and Siddique, 2007). Other researchers point out that the product architecture leverages adaption speed, flexibility, design “robustness” while reducing the cost of engineering changes and making the product less sensitive to volatile market demands (Thomke, 1997, p. 117; Thomke and Reinertsen, 1998, p. 25–27; Simpson et al., 2014). Publications about the role of product architecture in industry accredit the role of product architecture design for successful products (ZF Friedrichshafen AG, 2012; Steinbeis, 2011; Sanderson and Uzumeri, 1995; Simpson et al., 2006; Handelsblatt, 2014) and processes (Feitzinger and Lee, 1997; Holzner, 2006; Handelsblatt, 2014). In consequence, these studies and reports from industry show the practical benefits behind the principle of product architectures (Halman, Hofer and van Vuuren, 2003, p. 149; Roland Berger Strategy Consultants, 2012).

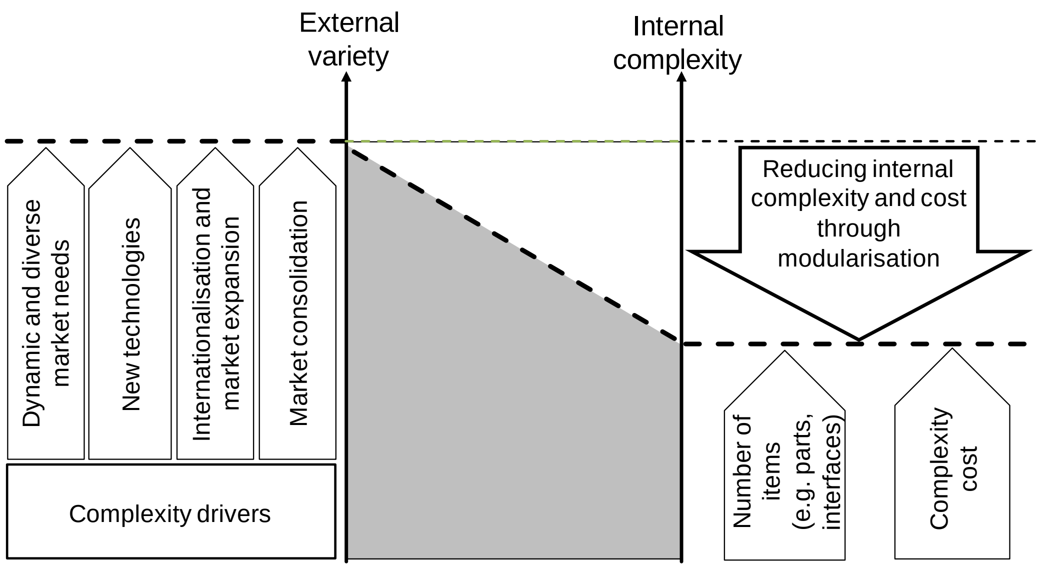


Figure 2: Modularisation as a lever to enable high external variety with low internal complexity, based on (Krause et al., 2014; Renner, 2007; Schuh and Schwenk, 2001)

Figure 2 shows the desired mitigating effect of modularisation on increased internal complexity while enabling a high degree of external variety. This is a combination of three pieces of work, put together to illustrate the type of issue that modularisation is set out to tackle in the research transition.

1.4 Product architectures

The principle of modular product architectures has been established in industry and theory for some time (Lehtonen, 2007). Nevertheless, there has been sharply increased practical and theoretical interest in researching this field due to the context of today's market environment. In a review-based study, Fixson (2007, p. 89) shows that the number of published articles in product architecture research doubled between 1995 and 2005.

The product architecture is in general terms defined as "the scheme by which the function of a product is allocated to physical components" and the way those components interact (Ulrich, 1995, p. 420). The product architecture type has far reaching consequences for manufacturing companies. It determines the number and type of components of a product, a product family or a whole product portfolio. Moreover, it describes how the components interact, how they can be combined and how they are assembled. This means that the architecture also describes the number and type of interfaces of the considered product range. In sum, the product architecture represents the structure of a company's products or of its whole product range (Fixson, 2005, p. 346–347).

The work presented in literature divides product architectures into integral and modular. A modular architecture includes one-to-one mapping between functional and physical elements and de-coupled interfaces that do not have to be changed if other parts of the architecture are changed. In turn, a modular product is composed of modules with standardised interfaces where modules can be flexibly interchanged. Integral product architectures have a complex mapping between functional and physical elements. Moreover, integral architectures have complex, non-standardised interfaces between their components (Ulrich and Eppinger, 2004).

In practice, products are neither fully integral nor fully modular. Products rather have a certain degree of modularity or different parts with different degree of modularity. Ulrich and Tung (1991) define modularity as a gradual property. Especially companies with a broad product range have the possibility to apply different product architecture types. The range of different alternatives varies from a rigid platform strategy (i.e. the platform as one shared module) over the application of modular systems (i.e. several predefined modules) to the free customer-individual configuration of modules (Schuh and Schwenk, 2001, p. 88–90).

In other words, product architecture influences how product families and platforms are structured. This in turn, has a strong and critical impact on how company-wide standardisation and reuse can be accomplished and how variety can be provided to the customer.

There are a considerable number of publications which describe benefits of successful product architecture improvement (Kusiak, 1995; Simpson, 2004; Pahl et al., 2007; Dieter and Schmidt, 2009). For instance, improved product architectures in the Black and Decker Power Tool business reduced product cost by 50 % which was due to a cut in product complexity (Meyer and Lehnerd, 1997). Another case described in literature showed that an intelligently designed product architecture enables high variety which can effectively be sold to the customer. Within only ten years, 160 different product models from the same Sony Walkman platform could be offered to the customer which gave the company a high competitive advantage compared to its competitors (Sanderson and Uzumeri, 1995). More recent examples from industry report product cost savings of 30 % (Pander, 2012) and R&D spending cuts of 30 % – 50 % (Scania, 2009) while tremendously increasing offered product variety through the introduction of modular systems.

Having set the scene on architectures, the next section deals with the attempts of work reported in the literature to support engineering designers in establishing modular product architectures. It focuses on the aspects that are particularly relevant to this research work: how product architectures are seen to be handled in the engineering design process, methods to create them and how product architectures can be assessed.

1.4.1 Product architectures in the engineering design process

The complex nature of the engineering design process can be described by using models. These models can be classified as prescriptive and descriptive models (Finger and Dixon, 1989). Descriptive design models describe what “processes, strategies and problem solving methods designers use” (Finger and Dixon, 1989, p. 52). Prescriptive models either prescribe what attributes the “design artefact” should have or how the ideal design process should be (Finger and Dixon, 1989, p. 55).

An example for a prescriptive design model is Axiomatic Design. The axioms prescribe that the product should have independent functional requirements and a minimum of information content (Suh, 2001, 1990). Other examples for prescriptive process models that describe the necessary procedure from an abstract towards a concrete product (Hubka, 1982, p. 23) are given by Pahl et al. (2007), Dieter et al. (2009), Roozenburg (1995), Lindemann (2009), Ponn and Lindemann (2008), Ulrich and Eppinger (2008), Ullman (2003), and the association of German engineers with the VDI Guidelines 2221 (Verein Deutscher Ingenieure, 1993) and 2206 (Verein Deutscher Ingenieure, 2004a). In the given models, product architecture creation is explicitly mentioned as important activity in the design process. It can be concluded that system structuring or product architecture creation is a vital part of the design process. For instance, the VDI Guideline 2221 (Verein Deutscher Ingenieure, 1993) takes product architecture creation as own phase in their process model with seven phases in total. Ulrich and Eppinger (2012) take it as phase-overarching activity which brings the architecture from an abstract state into a more concrete state.

It has to be stated that in reality the steps of the process models are not undertaken in a linear manner like is depicted by prescriptive engineering design models. In fact, the development process is passed in a highly complex and iterative manner (Verein Deutscher Ingenieure, 1993, p. 11). This is constituted by iterative feedback cycles between action and evaluation (see Figure 3). Action phases create different alternatives which undergo evaluation phases in later stages (Dieter and Schmidt, 2009; Roozenburg and Eekels, 1995). The spiral development model which is derived from software development describes such an intensive cycle between target setting, evaluation, action and planning (Boehm, 1995). Close relation of action and evaluation in general development are also valid for processes which are used to establish product architectures and modular systems.

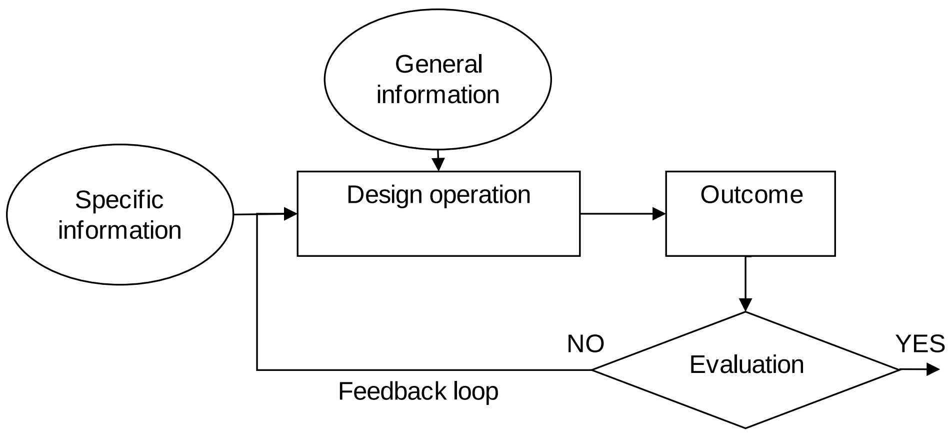


Figure 3: Iterative cycle in engineering design, based on (Dieter and Schmidt, 2009, p. 7)

To establish product architectures, the iterative design cycle with a focus on product architecture specific elements has been surveyed by many researchers. It has been shown that the most frequently researched means of support are operational methods to establish product architectures (Heilemann et al., 2012). Therefore, the next section briefly describes methods to establish product architectures.

1.4.2 Methods to establish product architectures

There are a wide range of methods from engineering design research that have the goal to create and improve product architectures. All methods have the same goal of restructuring the product architecture for a certain purpose. The methodological approaches have a very large set of characteristics. In order to be able to categorise the large set of methods and to bring them into the context of this research work, an adopted classification framework was created based on the work of Daniilidis et al. (2011) and Hackl et al. (2014). For this research work they were classified according to *what* they consider or relate to for product architecture improvement during the engineering design process. They are listed below:

- Modularisation principles

It has been analysed that abstract and theoretical methods strive to improve product architectures based on incorporated modularity principles. These methods either consider *functional relations* within *and* between products (Dahmus, Gonzalez-Zugasti and Otto, 2001; Day, Stone and Lough, 2010; Kurtadikar et al., 2004; Meehan, Duffy and Whitfield, 2007; Stone, 1997; Zamirowski and Otto, 1999), *functional-physical relations* within products (Tseng and Jiao, 1997; Stake, 2000; Goepfert, 1998) or *physical interactions between product elements* (Pimmler and Eppinger, 1994; Kusiak and Huang, 1996; Helmer, Yassine and Meier, 2008).

- Strategic factors

Strategic methods use as their basis a focus on all kind of *strategic factors* for module grouping. Many researchers take product life cycle reasons like similar processes or service characteristics into account (Coulter et al., 1998; Gu and Sosale, 1999; Ji et al., 2013; Newcomb, Bras and Rosen, 1996; Yu et al., 2011) when they establish the product architecture. Others cluster the product architecture based on a whole set of different strategic reasons for module grouping (Blees, Henry and Krause, 2009; Ericsson and Erixon, 1999; Erixon, 1998; Jonas, Gebhardt and Krause, 2012).

- Holistic methods

Researchers have also shown increasing interest in combining the methods mentioned above into holistic research methods considering whole sets of factors for product architecture improvement during the engineering design process (Blackenfelt, 2000; Blees and Krause, 2008; De Weck, Suh and Chang, 2004; Emmatty and Sarmah, 2012; Gonzalez-

Zugasti, Otto and Baker, 2000; Krause et al., 2014; Marshall and Leaney, 2002; Simpson et al., 2012; Simpson, Maier and Mistree, 2001).

The factors that are considered by these methods range widely from requirements collection over detailed design up to product disposal.

All of these topics are dealt with in more detail in Chapters 2 and 3.

1.5 Summary: state-of-the-art

It has been shown that the prevailing diverse and volatile market environment has led to an increasing research interest in product architectures and modularisation. Researchers describe principles and the underlying rationale for different types of product architectures. From this, theoretical benefits and the limitations of modular product architectures are derived. Numerous practical cases from industry reported in the literature also support the findings from theory. Thus, significant scientific value is ascribed by general engineering design literature to product architecture design. In turn, this has led to a large and growing body of literature describing how product architectures can be designed. In line with the iterative model of the design process, predominant discussions are on methods that deal with the improvement and evaluation of product architectures along the engineering design process.

However, there are still some important gaps and further research needs in this contemporary and significant research area, the distillations of these are dealt with below.

1.6 Research problem and knowledge gap

Product architecture design and implementation in organisations is on the one hand a very complex activity (Alizon et al., 2008; Kreimeyer, 2014; Plaikner et al., 2012; Simpson et al., 2006, p. 1–2; Simpson, Siddique and Jiao, 2005a, p. 5) and on the other hand a critical factor for the success of a company (Alizon et al., 2008; Muffatto, 1999, p. 1). It has been shown that successfully designed product architectures that are implemented company-wide create competitive advantage for manufacturing companies (Alizon et al., 2008; Muffatto, 1999, p. 2, 1996; Simpson, 2004; Simpson et al., 2014). However, poorly designed, poorly implemented and poorly managed product architectures consume high amounts of resources (Automobil Produktion, 2014; Halman, Hofer and van Vuuren, 2003, p. 159; Muffatto, 1999; Simpson et al., 2006; Sundgren, 1999; Zacharias and Yassine, 2008) while making derived variant products lagging behind that of competitors (Alizon et al., 2008; Cusumano and Nobeoka, 1998; Sörensen, 2006, p. 174). It can be derived that product architecture design is vital for the competitiveness of a company. However, product architectures cannot be adjusted over night (Simpson et al., 2014).

Even though, the introduction of platform- and modular design is not new for both, theory and practice, companies are pursuing modular strategies more aggressively and are investing significantly more resources in developing platforms since the last few years, and in particular the last two to three years (Simpson et al., 2014, p. v–vi).

The difference to the past and the new challenge is to think, first, in higher "variability" (Simpson et al., 2014, p. vi), i.e. in a *wider range* of product variants, product generations, technologies, markets, brands and development sites. The second different point is to derive different product families from flexible modular systems instead from rigid platforms. The third point is the extended strategic top-down view from an overall corporate level (Simpson et al., 2014, p. 779–782). These points make companies *without* historically grown product architectures or with a shift of the scope of the product architecture having to really rethink their product architecture strategies. If they have one at all!

Considering the circumstances of the new challenge, making the *transition* from single product development towards multi-product development with common modular product architectures is not straightforward. This is particularly true in the case of industrial practice (Automobil Produktion, 2014; Freitag, 2014). In fact, such an undertaking is intricate and complex (Arnoscht, 2011; Kreimeyer, 2014; Simpson et al., 2014, p. v–vi).

1.6.1 Main field of research

The major gap in existing research about product architecture design support arises from the fact that the <u>overall issue of transitioning</u> from single product development to the development of modular systems in industry is only rarely considered by the current literature. Only very few researchers have been able to draw on any systematic research into modularisation transition. A study from 2011 focused on rather organisational aspects of developing and introducing modular systems in industry and was only evaluated through discussions in expert workshops in agricultural machinery industry (Arnoscht, 2011). Two other studies from vehicle industry are ongoing in parallel to this work (Kreimeyer, 2014; Vietor and Hoffmann, 2014).

In addition, only few studies from *adjacent fields* exist which report on how to implement standardisation or platforms in industry (Gudmundsson, Boer and Corso, 2004; Karandikar and Nidamarthi, 2007; Muffatto, 1999; Shibata and Kodama, 2013; Wijnstra, 2004). Besides that they are not focused on transitioning towards modular system development, they have the flaw that they focus on i) single examples, ii) standardization or platforms, iii) just on the initial implementation activity, iv) only on software or v) on management issues. Moreover, another flaw is that the required depth is missing and that the topic is not treated in much detail from an engineering design perspective. Thus, these studies are not sufficient for supporting industry in making the transition towards modularisation.

1.6.2 Implementation into industrial practice

This major gap identified leads to further problems of this research field. In general, engineering design research is criticised for a lack of use of results in practice (Blessing and Chakrabarti, 2009, p. 7). In many cases, the developed methods are overly specified, miss the real issues of practitioners and produce “solutions to problems that do not exist” (Eckert, Stacey and Clarkson, 2003). Modularisation support (see Section 1.4.2) is only, if at all, initially evaluated in industry without giving insights of its implementation into daily practice. However, engineering design support should be tested in serious industrial use, which means the methods themselves as well as “the process of introduction” (Eckert, Stacey and Clarkson, 2003; Halman, Hofer and van Vuuren, 2003).

Moreover, existing support from literature tends to focus on single products or product families with rather narrow focus (Jonas, Gebhardt and Krause, 2012; Marti, 2007) or on single methods to establish modular product architectures (Arnoscht, 2011). However, this is not sufficient as transitioning itself is complex (Arnoscht, 2011). Hence, to maximize benefits of product architecture design in engineering organisations, it is suggested that it is necessary to make holistic considerations (Götzfried, 2013) while taking into account “variabilities” of practice (Simpson et al., 2014, p. vi). Halman et al. (2003, p. 161) state that the gap is really associated with “strategies to manage the risks and problems related to platform and product family development and implementation” in different industries. In addition, this means that it is necessary to go beyond current methods (Kristjansson, 2005) which have a strong focus on new product development and not on the fact that most designs evolve iteratively and are adoptions of past designs (Arnoscht, 2011; Clarkson and Eckert, 2005; Reddi and Moon, 2013; Vietor and Hoffmann, 2014).

1.6.3 Maintaining stability of the architecture

Current approaches assume that modularisation is completed after establishing the product architecture (Bahns and Krause, 2013; Nielsen, 2010; Schuh, Aleksic and Rudolf, 2015; Vietor and Hoffmann, 2014; Wijnstra, 2004). However, studies in practice have shown that the stability of the common product architecture is jeopardised exactly after this phase (Arnoscht, 2011; Koziolek et al., 2013; Munk, 2011; Nielsen, 2010). Interestingly, literature does not suggest any solution how the product architecture of complex product families can be kept stable over time without eroding or breaking apart. This “breaking apart” of architectures is closely related to “platform divergence” which has been described recently as major future research direction in the field of platform design (Boas, 2008; Montano, 2011; Simpson et al., 2014) and modularisation (Bahns, Gregor Beckmann, et al., 2015).

1.6.4 Summary: knowledge gap

In sum, it is the main argument of this research that far too little attention has been paid to the <u>overall issue of transitioning</u> from single product development towards <u>modular system development</u> based on common, modular and <u>stable product architectures</u> in <u>practice</u>. Consequently there is a clear knowledge gap in this area.

As a result, it is the purpose of this research to develop understanding about overall issues, important factors and support for the transition from single product development to the development of multiple products based on common, modular and stable product architectures. Further, this deeper understanding allows for developing support for the transition towards modularisation.

1.7 Research aim

The aim of this research is to identify and test critical issues and important factors associated with support for the transition towards modular system development with stable product architectures.

Based on these findings, it is proposed to develop engineering design support for the transition.

In order to achieve the research aim, this research work will be divided into two parts which examine four main research questions:

- Part 1: Establishing a deep understanding of transitioning towards modular system development in industry:
  a) What are the vital elements that have to be considered for transitioning towards modular system development?
- Part 2: Developing support for the transition in industry based on the findings of part 1:
  b) Does a modularisation assessment framework support companies in making the transition? What is an appropriate modularisation assessment framework?
  c) Does the assessment of product architectures support companies in making the transition? What are appropriate metrics to assess product architectures during the transition?
  d) Does the provision of product architecture information in standard IT-Systems support companies in making the transition? What is an appropriate approach for the IT-integration of product architecture information?

These research questions (RQ) have been “translated” into research objectives (RO). Moreover, they have been used to derive research activities and the deliverables of this research work (see Table 1).

Table 1: Research objective in relation to research activities and deliverables

Research objective	Research activity	Deliverables
RO1 (from RQ1): To identify vital elements for modularisation transition	• Participating in and observing activities of a transitioning company • Coding and classifying field notes and other documents • Semi-structured interviews within organisation and benchmark organisations • Testing engineering design support in industry	• List of issues and corresponding important factors that must be in place • List of use and limits of support for modularisation transition • Support framework for modular system development
RO2 (from RQ2): To develop a modularisation assessment framework for the support during modularisation transition	Developing, testing and validating the modularisation assessment framework in industry	Modularisation assessment framework
RO3 (from RQ3): To develop metrics that support companies in modularisation transition	Developing, testing and validating metrics in industry	Modularisation metrics
RO4 (from RQ4): To develop an approach for provision of product architecture information into standard IT-Systems for modularisation transition	Developing, testing and validating an IT-Integration approach in industry	Approach for IT-Integration of modularisation

1.8 Research methodology overview

The "twin goals" of engineering design research is to understand designing and to improve the way designing is done (Eckert, Stacey and Clarkson, 2003). This requires a totally different set of varied research methods. When design research is undertaken with industry, it is important that there is a clear differentiation between doing a "consultancy" for industry and doing real research with proper research methods. The goal of industry focussed consultancy is to produce immediate solutions that work somehow. However the topic of Engineering design itself is much more complex. Thus, it is the challenge of engineering design research to produce both, a) valid, reproducible, innovative and well-grounded research results as well as b) solutions to practical problems (Eckert, Stacey and Clarkson, 2003).

Blessing and Chakrabarti (2009) identified three main issues in engineering design practice: lack of overview of already existing research, failure to produce results that are applied in practice and lack of scientific rigour. In order to overcome these main issues, they introduced a Design Research Methodology (DRM) that supports engineering designers to produce valid research results with high potential for practical application.

DRM comprises four research phases. Firstly, "Research Clarification" sets the research focus by analysing state of the art, identifying a knowledge gap and establishing research goals. Secondly, the "Descriptive Study I" builds the foundation for later design support by generating deep understanding about respective design issues and factors for success. Thirdly, during the "Prescriptive Study" the actual design support is developed. Finally, the "Descriptive Study II" validates the developed design support (Blessing and Chakrabarti, 2009).

The left column of Figure 5 shows the research phases of this work:

- "Research Clarification" is setting up the research focus by screening literature and by doing preliminary studies in industry.
- "Descriptive Study I" is concerned with generating understanding about modularisation transition. Therefore it builds upon a longitudinal case study approach in industry. Mainly qualitative data collection is done during participant-observer studies, semi-structured interviews, observations, document analyses, experiments, action research and small surveys. The collected data is qualitatively analysed through coding. This has the purpose to find out predominant themes and relationships. The results of this research phase are issues, important factors and tested support for modularisation transition. The results of this research phase can be seen as requirements for developing support in the next research phase.
- During the "Prescriptive Study" actual support for modularisation transition is developed. From a methodological research viewpoint, this research phase heavily depends on action research. Action research is an adequate research method to improve matters in organisations through a cyclical process of questioning, reflecting, investigating, developing, implementing and refining (McIntyre, 2008). The support comprises a modularisation assessment framework, modularisation metrics and an approach for IT-Integration of modularisation.
- The "Descriptive Study II" evaluates the support. This is mainly done through expert interviews with multiple investigators, workshops with practitioners, application and implementation in industry, publication of findings and triangulation. Focusing on a case study, even with multiple cases, makes it difficult to generalise research out-

comes. Therefore, it is claimed that generalisation is achieved with the mentioned means of validation and through reflecting outcomes with accepted theory in literature.

The descriptive and prescriptive elements of this research were to a large extend conducted on site in the central engineering department of a large German manufacturing company which is making the change from single product development towards modular system development. The primary case company is a major player in international HVAC industry and headquartered in Germany. This case was reinforced through numerous secondary cases which led to a longitudinal case study with several cases. For instance, secondary cases comprise benchmark partners and consultancies. It is claimed that a case study is the most appropriate way to collect in-depth information over a longer period like it is necessary for this research enquiry. Moreover, a case study provides a sound base to develop and test engineering design support. Based on the knowledge of the descriptive studies, an adopted action research approach was used to develop support for modularisation transition in industry. The goal of the developed support is to prevent drawbacks during modularisation transition. Hence, it can be concluded that the overall research methodology helps to stringently close the described knowledge gap.

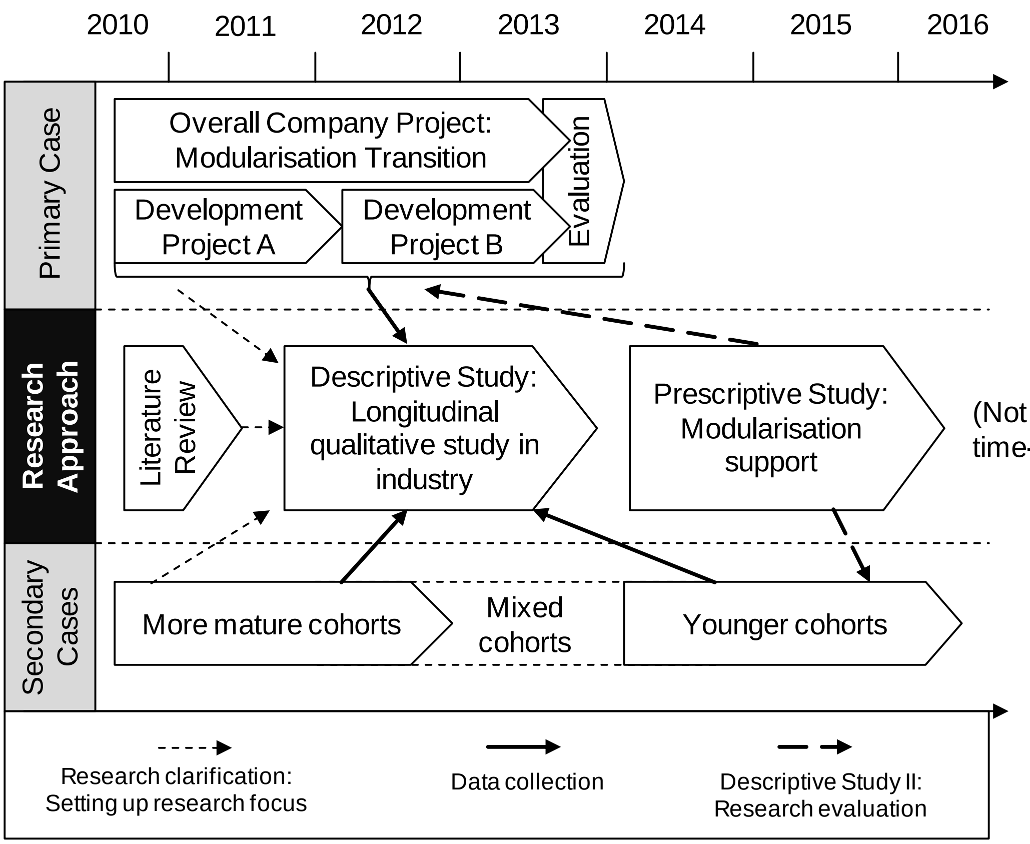


Figure 4: Overview of interplay between industrial cases, data collection, research evaluation and research contribution

Figure 4 shows an overview of the longitudinal case study and the interplay between research contribution, primary case and secondary cases. It becomes clear from the figure how the research study draws upon three kinds of sources: central engineering department of the primary case company, development projects in different business units of the primary case company and different secondary cases. In the first major research phase, it is the goal to gather and analyse information about how companies can make the transition towards modular system development (Descriptive Study). Therefore, data was collected from all research cases involved in order to achieve RO1. The succeeding Prescriptive Study aims to achieve RO2-RO4. Based on the improved understanding, the outcome of the Prescriptive Study is support which is based on intervention and validation mainly in primary but also in secondary cases in industry.

1.9 Outlined structure of this thesis

The introduction to this research work is presented within this chapter (Chapter 1).

Chapter 2 presents the literature review and describes the principles of modular product architecture design. This comprises: principles of product architectures, platforms and modular systems as well as the role of product architecture in the engineering design process.

Existing modularisation support, i.e. methods to establish modular product architectures are presented in Chapter 3. At the end of Chapter 3, common understanding about existing support to modular design is established. Moreover, Chapter 3 will also present examples and issues from industry. At this point of the thesis it becomes clear that there is still a clear knowledge gap in the field of modularisation transition. To fulfil the needs of industry, this gap has to be understood and closed effectively.

In order to close the knowledge gap and to ensure that scientific rigour is applied during the case study in industry, a well-defined research methodology was set up. Chapter 4 explains the research framework, the mode of data collection & -analysis, the method for developing support and how the research results were validated.

Chapter 5 gives an overview of issues that companies encounter during transitioning towards modularisation. Moreover, it derives important factors and tests support for modularisation transition in industry. To sum up, Chapter 5 presents deep understanding about modularisation transition and it is therefore the base for development of innovative modularisation support in the next chapters.

Chapter 6 is about guidance on what needs to be done for modularisation transition in companies. For this reason, a modularisation assessment framework that identifies weak spots and leads organisations towards the right actions is presented. The delivery of Chapter 6 can really be seen as framework for assessing the process side of modularisation transition.

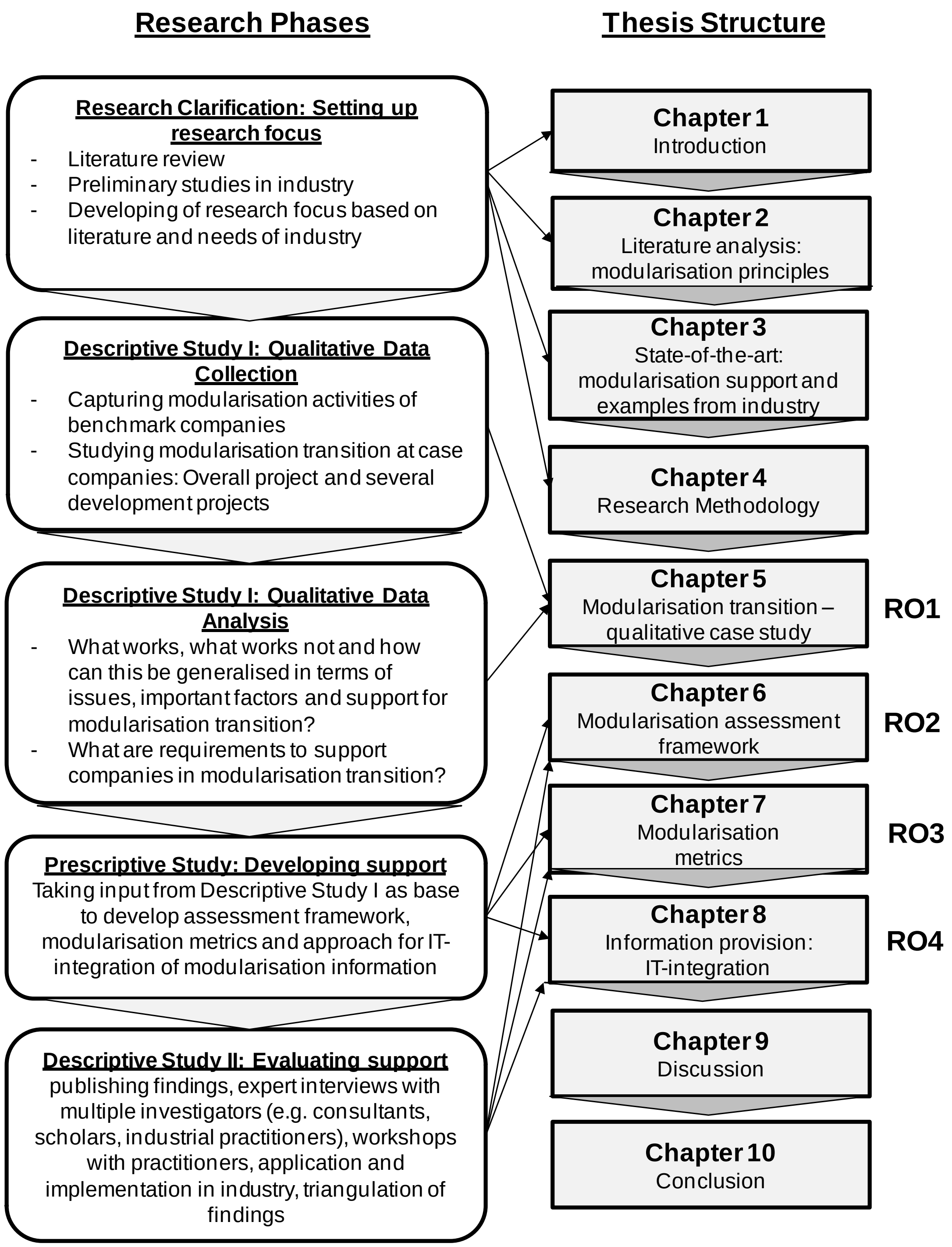


Figure 5: Layout of this research report: Relation of chapters (right hand side of figure) to research phases (left hand side of figure)

Complementary, Chapter 7 gives support on how to assess rather physical aspects (e.g. products, modules) during modularisation transition. To start with, it shows requirements for evaluating modularisation transition within industry. Moreover, it shows how a set of metrics that support companies in transitioning towards modularisation were developed, applied and validated.

It has been shown that there is a lack of *explicit* information about product architectures and its related fields in industrial organisations. Hence, it is difficult or nearly impossible to efficiently apply the support of Chapters 6 and 7 in daily organisational practice. In order to remedy this and to provide a coherent set of applicable support, Chapter 8 introduces an approach for IT-Integration of product architecture information. It is the purpose of this chapter to show a valid approach that helps engineers and engineering managers in gaining transparency about modularisation transition and in assessing modularisation transition, both, from a process and product view.

The relation of research phases to the layout of chapters of this thesis is depicted in Figure 5.

2 Literature review: principles of modularisation

It is the purpose of this chapter to build the basic foundations and research rationale for this work. It deals with what is already known about product architectures and modularisation. As was pointed out in the introduction of this thesis, modularisation has the main purpose of attacking increasing complexity in companies. Modularisation transition done as described within this thesis is a proposed new enabler.

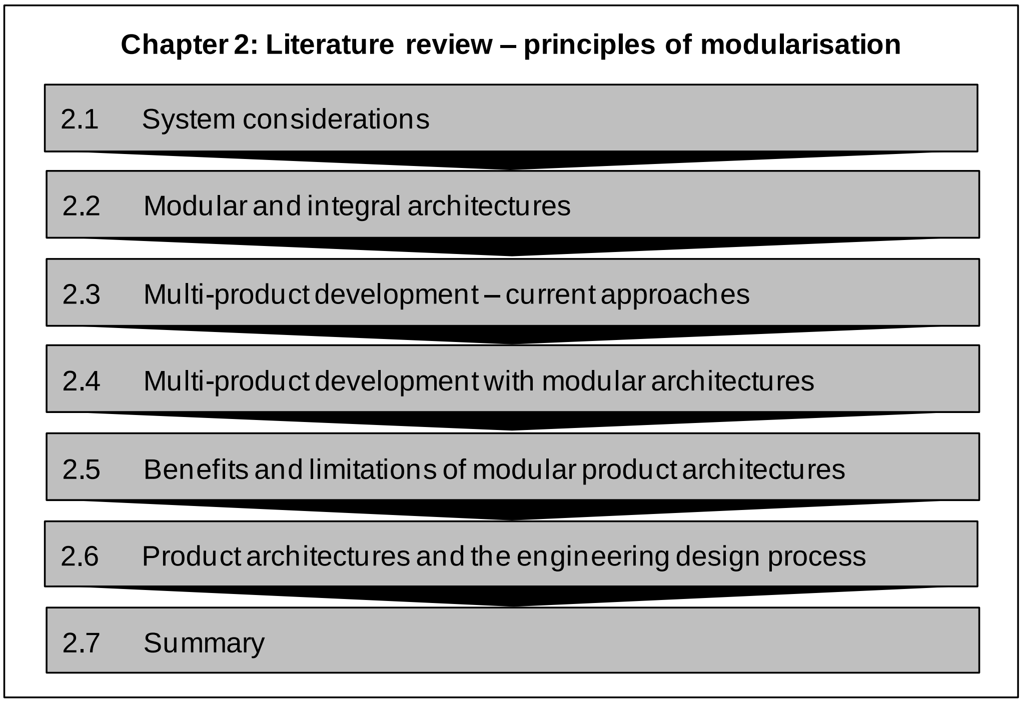


Figure 6: Elements of Chapter 2

It is worth briefly reflecting on the nature of complexity. The problem of increasing complexity is often treated by using the term “complexity” to describe various phenomena: sometimes the *nature* of complexity and sometimes the *effects* of it. Therefore, the term has to be specified to make the use of the term complexity understandable. For the purpose of this work, complexity is described as structure-related characteristic of a system. Consequently, the characteristics of complexity are:

- Diversity and variety of elements, i.e. number of different types of elements and overall number of elements (Franke et al., 2002; Malik, 2003)
- Intensity and diversity of interactions between elements, i.e. strength, number of different types of interactions and overall number of interactions (Ehrlenspiel, 2007; Franke et al., 2002; Gomez and Probst, 1997; Malik, 2003)
- System’s change rate (i.e. high system’s dynamics), (Gomez and Probst, 1997)

Having considered complexity this chapter starts by describing and looking at the nature of systems and uses this theoretical base to inform the other considerations (see Section 2.1). It then covers what modular and integral architectures actually are (see Section 2.2). In the subsequent sections, the concept of multi-product development (see Section 2.3) is introduced with a particular emphasis on modular architectures and modular systems (see Section 2.4). Afterwards, benefits and limitations of different product architecture types are dealt with in Section 2.5. To complete the analysis, the role of product architectures in existing engineering design process models is described (Section 2.6). Figure 6 shows the elements of Chapter 2.

2.1 System considerations

The system is the overarching descriptor of the product and the product elements and it is here that the underlying complexity can be generated and can start to emerge. Accordingly, to improve the complexity of a system, it is necessary to optimise the variety and diversity of elements, the intensity and diversity of interactions between elements and to control the system's dynamics. This improvement has to be done while meeting fixed requirements (e.g. product requirements from the customer) and underlying constraints (e.g. limited resources). It is claimed, that this can be achieved by reorganising a system's elements. In other words, the main lever to improve the complexity of a system is the improvement of a system's structure with its constituting elements (Pine 1992; Baldwin & Clark 2000; Goepfert 1998; Piller 2001). Before it is described how this can be achieved, the next sections provide the background to understanding the relationships between systems and product architecture and modularisation.

2.1.1 Definition of system

Systems can be described as "a construct or collection of different elements that together produce results not obtainable by the elements alone" (National Aeronautics and Space Administration (NASA), 2007, p. 3). Hubka (1982, p. 110) adds that a system is a "set of elements and their relationships within a clearly defined boundary". Figure 7 shows an abstract depiction of a system. The international standard ISO/IEC/IEEE 15288:2015 follows these definitions and gives a system a designated purpose: "combination of interacting elements organized to achieve one or more stated purposes" (ISO/IEC/IEEE, 2015, p. 9). In sum, a system can be described as follows:

- set of different elements
- interactions between these elements
- clearly defined boundary
- designated purpose

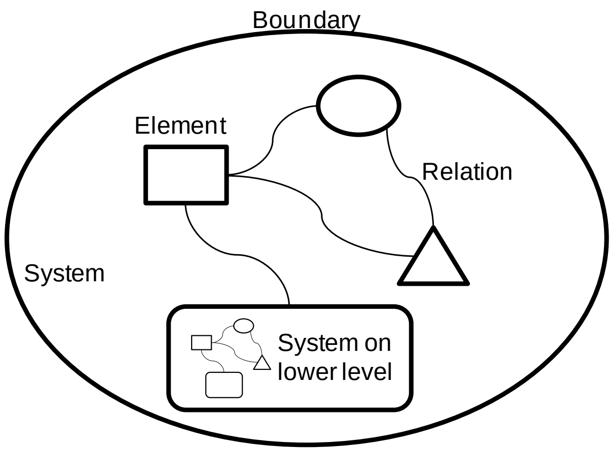


Figure 7: Abstract depiction of a system (similar to Schaeppi et al. (2005, p. 31))

2.1.2 System types and system levels

On the one hand, structuring can be applied on different types of systems (see left side of Figure 8). For instance, it can be applied on technical systems like products (Salvador, 2007), on organisational systems (Sanchez and Mahoney, 1996) or in production (Pandremenos et al., 2009). The focus of this work is on technical systems such as described by Hubka (1982).

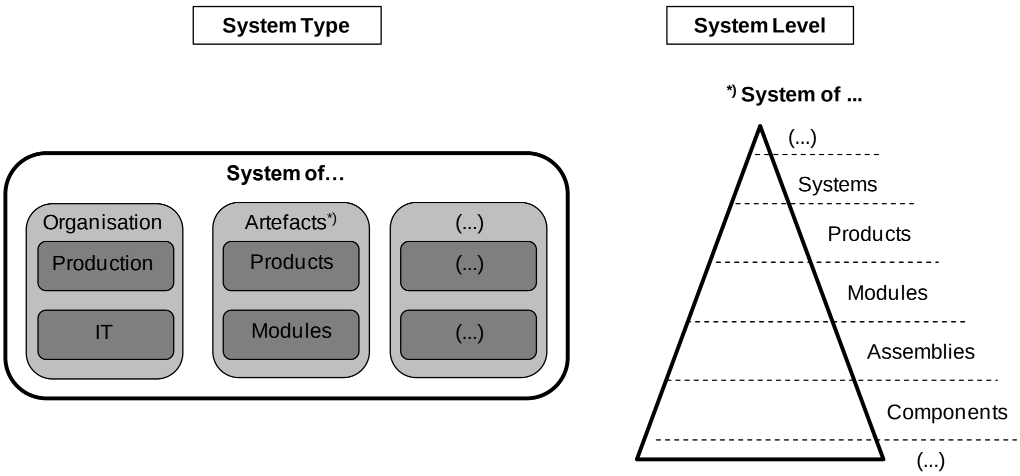


Figure 8: Different types and levels of systems (based on Eeles and Cripps (2010, p. 19))

On the other hand, a technical system can be considered on different levels (see right side of Figure 8). Therefore, it is always necessary to clarify exactly the scope of a technical system (just referred to as "system" in the further course of this work).

A system can have totally different scopes (Eeles and Cripps, 2010, p. 18–19). For instance, a system can be seen as an artefact that comprises different final products of different companies. Moreover, a system can be seen as something that includes different final products of the same company. It is also possible to regard a system as a final product itself or on the level of a product's modules, assemblies or components.

2.1.3 System structure and product architecture

Structure is a major characteristic of a system (Hubka, 1982, p. 110) which determines the complexity of the system (Schuh and Schwenk, 2001, p. 73). A specific focus of a "system structure" is a "product architecture", mainly used in engineering design literature (Lindemann, Maurer and Braun, 2009, p. 24). Literature and industry have come to no agreement on the exact definitions of the terms. This is resulting in "different meanings for the same term (homonyms) and two or more terms that mean the same thing (synonyms)" (Eeles and Cripps, 2010, p. 18). Rechtin and Maier (2000) discuss architecture and structure and come to the definition that an architecture is "the structure (in terms of components, connections, and constraints) of a product, process or element". Therefore, "structure" and "architecture" are used as synonyms for the purpose of this work. This means that a "system structure" can have the same meaning as a "product architecture". When an architecture is considered, it is always necessary to consider the above mentioned scope of the system as well.

There are numerous views on "architecture" by scholars in engineering design:

Erens and Verhulst (1997, p. 170) define product architecture as the "composition of a product from a number of component products". More extensively, architecture can also be described as "the components, together with their interfaces and operation" (Erens and Verhulst, 1997, p. 170). However, Erens and Verhulst (1997, p. 170) leave it open whether a component is of functional, technological or physical nature.

Other researchers from the system and software engineering domain have a more extensive definition of architecture and add design and evaluation principles as well as interactions with the environment (Institute of Electrical and Electronics Engineers (IEEE), 2010, p. 20):

> *Fundamental organization of a system embodied in its components, their relationships to each other, and to the environment, and the principles guiding its design and evolution.*

The concept of "product architecture" referred to mostly in new product development is related to the "layout, configuration, or topology of functions and their embodiments" (Van Wie et al., 2003, p. 1). Similarly, Ulrich and Tung (1991) view product architecture as the allocation of functions to components.

As well as functional organisation, and functional-physical allocation, there is also the relation between physical elements. Two frequently referenced and widely recognized definitions of "product architecture" among researchers are the definitions of Ulrich and Eppinger (2012, p. 185) and of Ulrich (1995, p. 420). These definitions are closely related to research about product architectures in engineering design and to general research about modularisation.

The definition of Ulrich (1995, p. 420) considers the structure of functional elements, functional-physical relations and interactions between physical elements:

> *(1) the arrangement of functional elements; (2) the mapping from functional elements to physical components; (3) the specification of the interfaces among interacting physical components*

The definition of Ulrich and Eppinger (2012, p. 185) takes the same characteristics into account while, in contrast to Ulrich (1995, p. 420), excluding functional elements on their own:

> *The architecture of a product is the scheme by which the functional elements of the product are arranged into physical chunks and by which the chunks interact. (Ulrich and Eppinger, 2012, p. 185)*

All definitions deal more or less with the organisation of physical or non-physical elements, the relations between them and additional constraints and conditions. The smallest common denominator of all definitions is the breakdown of the system into its elements and the relations between them. Accordingly, Crawley et al. (2004, p. 2) introduce a generic definition while not just focusing on physical elements but also on non-physical elements like functional elements, etc.:

> *System architecture is an abstract description of the entities of a system and the relationships between those entities.*

It is suggested that this definition is well suitable to cover the definition of "product architecture" and that it is most suitable and sufficiently generic for the purpose of this work. Therefore, the definition of Crawley et al. (2004, p. 2) is taken as guiding definition for this work with the assumption that the scope of the system guides the scope of the architecture, according to the definition, an architecture can be on different levels. For instance, a "logical architecture" may be technology-independent whereas a "physical architecture" may be technology-specific (Eeles and Cripps, 2010, p. 3). Moreover, an architecture may comprise entities of different levels and relations between entities of different levels (e.g. functional domain and physical domain).

However, whenever it comes to describing the type of a product architecture or the principles of modularity, literature comes back to a more specific definition. Therefore, in cases where the principles of modularity are explained from a literature point of view, the more detailed definition of Ulrich (1995, p. 420) is used. It is claimed that the definition of Crawley et al. (2004, p. 2) is on a higher level and covers the more specific definitions of Ulrich (1995, p. 420) and of Ulrich and Eppinger (2012, p. 185).

Before different types of product architectures are described, an illustrative example of a product architecture is given. Figure 9 shows two different car drive architectures for the case of a rear wheel drive and a front wheel drive. It is assumed that both architectures have a front engine. For both architectures, the functional elements of the product are arranged into the same physical chunks. However, the rear wheel architecture has an additional transmission function which is arranged into an additional drive shaft. Additionally, the two types differ in the way by which the physical chunks interact. For instance, for the rear wheel drive architecture there is an interaction between the drive shaft and the rear axle whereas the front wheel drive architecture does not have such an interaction. This means that the interfaces between the interacting elements are totally different between the two architectures.

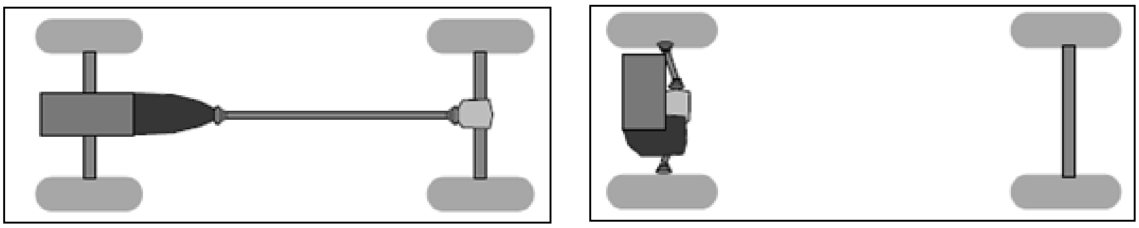


Figure 9: Exemplary front wheel drive (right) and rear wheel drive (left) architecture (Whitney, 2004, p. 3)

It is the purpose of the next section to describe different typologies of product architectures.

2.2 Modular and integral architectures

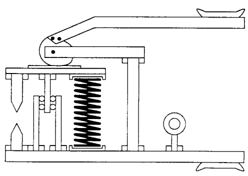
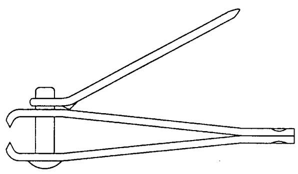
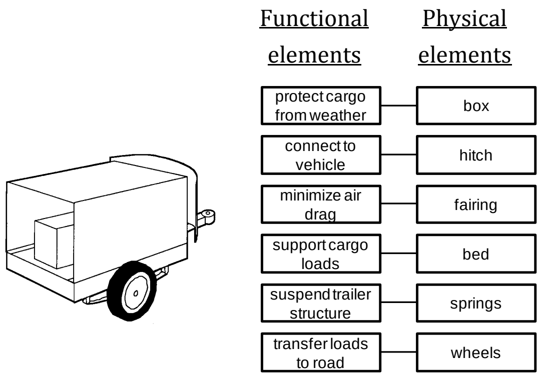
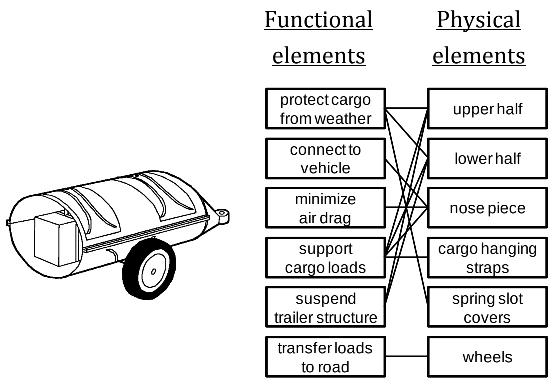
Every system has an architecture, even if it is simple and comprises only a single element (Eeles and Cripps, 2010). However, the architecture of a system is not always directly obvious and needs further investigation.

The two extreme types of a product architecture are integral architectures and modular architectures (Jose and Tollenaere, 2005, p. 376). According to point 2) and 3) of Ulrich's (1995, p. 420) definition and of the entire definition of Ulrich and Eppinger (2012, p. 185), there are two main characteristics in the nature of the product's architecture that determine the degree of modularity:

2.2.1 Arrangement of functional elements into physical elements

Ulrich (1995) and Erens & Verhulst (1997) denote that a modular architecture has one-to-one mapping from functional elements to physical elements. In this case, the subsystems of the product architecture are functional independent. The other extreme, an integral architecture has a more chaotic structure. Either a functional element is mapped to more than one physical elements (1:N relation), more than one functional elements are mapped to one physical element (M:1 relation), or several functional elements are mapped to several physical elements (M:N relation). Table 2 shows two examples of a purely integral and a purely modular architecture. A well-known example for a modular product is a desktop computer if simplifying assumptions are made. Therefore, the functions of the computer are "realize data input", "process data", "display data", "play sound". In a modular desktop computer, these functions are all mapped to another part of the product. Data input is done with the keyboard, data is processed in the tower, data is displayed from the monitor, and sound is played from the speakers. All simplified functions of the desktop computer are mapped one-to-one to parts of the computer. On the other hand, another well-known example for an integral architecture is a laptop computer. If the same simplifying assumptions are taken, at least the functions data input and processing data are mapped to the keyboard panel whereas playing sound and displaying data are mapped to the monitor panel. For this reason, the functions cannot be mapped one-to-one to components.

Table 2: Different architecture types of a nail clipper (MIT, 2011, p. 9) and a trailer (Ulrich 1995, pp.421–422) based on functional-physical arrangement

<table>
<tr><th>Modular architecture</th><th>Integral Architecture</th></tr>
<tr><td>Nail Clipper 1
</td><td>Nail Clipper 2
</td></tr>
<tr><td>Trailer 1

</td><td>Trailer 2

</td></tr>
</table>

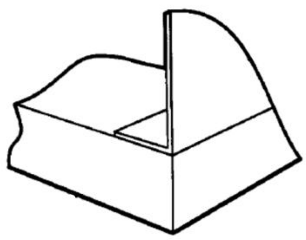
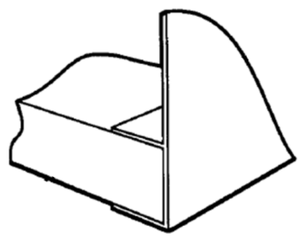
2.2.2 Interactions between elements

There are two points that make it possible to characterize product architectures based on interactions between elements:

- "Coupled" and "de-coupled" interfaces:
  A modular architecture has "de-coupled" interfaces. In an integral architecture, elements are linked with "coupled" interfaces. Interfaces are defined as coupled if a change in one component entails a change in another component. If components can be changed independently from each other, they are linked with de-coupled interfaces (Ulrich 1995, pp.421–422; Jose & Tollenaere 2005, p.376). Taking again the example of

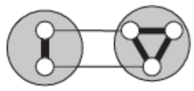
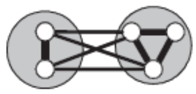
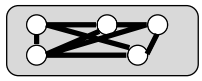
the computer, the desktop computer has de-coupled interfaces whereas the laptop computer has coupled interfaces. If the speakers of the desktop computer shall be changed, this can be done by unplugging it via a standardised, totally de-coupled interface. Whereas a change to the speakers of the laptop computer could possibly cause a change to the monitor panel because the interfaces are coupled. Table 3 shows two concrete examples of coupled and de-coupled interfaces. In the modular architecture on the left, changing the trailer bed does not require a change to the trailer box. In the integral architecture on the right, a change of the trailer bed requires to change the trailer box in order to maintain a functioning whole.

Table 3: Example for coupled and de-coupled interfaces (Ulrich, 1995, p. 423)

Modular architecture	**Integral architecture**
Trailer bed and box: De-coupled interface 	Trailer bed and box: Coupled interface 

- Relative strength of interfaces inside subsystems to strength of interfaces between subsystems:
  There is an additional way how interactions between elements can be described. To characterise the type of product architecture, it is also possible to determine the strength and location of interaction between the elements. In modular architectures, the internal interactions in a subsystem (modules) are much stronger than the strength of interaction between subsystems (modules). Therefore, modular systems are decomposable systems which incorporate relatively autonomous modules that act nearly independently (Goepfert, 1998). In contrary, a product architecture is integral if there is no difference between the strength of internal interactions in a subsystem and the strength of interactions between subsystems (see Table 4).

Table 4: Characteristics of product architectures based on the relative strength of internal and external interactions (Goepfert, 1998, p. 32)

Type of architecture	Internal to external interaction strength	Visual example
modular architecture	internal interaction >> external interaction	
integral architecture	internal interaction ≥ external interaction	 

Modular or integral product architectures are relative and not absolute. As both extreme types of product architectures have their pros and cons, most products are hybrids and somewhere in between strictly modular and strictly integral (Dieter and Schmidt, 2009, p. 302; Ulrich and Tung, 1991). Consequently, a product has a certain "degree of modularity". Even more, the degree of modularity may not be equally distributed over the product. In fact, a product has certain parts which are more integral and other parts which are more modular. For instance, in the BMW model R1200S motorcycle, the transmission case is integrated into the frame whereas the drivetrain is a separate module (Dieter and Schmidt, 2009, p. 302).

Having defined what is meant by a product architecture and what is meant by different types of product architectures of a single system, the thesis will now move on to discuss architectures across different systems.

2.3 Multi-product development – current approaches

As was pointed out in the introduction to this thesis, most companies have to cope with increasing variety of their product portfolio. A "product portfolio" may be defined as all of a firm's product variants that fulfil the needs of different customer groups (Jiao, Simpson and Siddique, 2007, p. 7). A product portfolio can be further broken down into different "product families". In literature, the term "product family" tends to be used to refer to a set of similar product variants that are based on a common product architecture to meet the requirements of a particular customer group or segment (Meyer and Lehnerd, 1997, p. 35). The term "product line" is often used for the same meaning as a "product family" (Wijnstra, 2004, p. 13). These synonyms will also be valid for the purpose of this work. Each distinct product within a product family is defined as "product variant". Each product variant addresses a particular set of customer requirements (Jiao, Simpson and Siddique, 2007, p. 7). Although, academic literature still focuses on single design processes (Clarkson and Eckert, 2005, p. 22), it has been well known that there is a need to shift the focus from single products towards product families or product portfolios.

Meyer and Lehnerd (1997, p. 2) state that many companies with single product development have difficulties to “embrace commonality, compatibility, and standardisation, or modularization among different products or product lines”. This lack of “structure and reuse in the design process” has a negative impact on other company and product life cycle related processes. Due to this situation, one can find different screws or switches for the same purpose within a company for example. Or one can find numerous different components with different materials but with the same technical specifications. This all increases costly complexity within a company while not achieving advantages of product communality.

Platform and modularisation approaches may remedy these issues (Andreasen, McAloone and Mortensen, 2001, p. 14). More generally expressed, a “well-planned” product architecture is the enabler for successful multi-product development (Du, Jiao and Tseng, 2001, p. 309):

> *As the backdrop of product families, a well-planned architecture - the conceptual structure and overall logical organization of generating a family of products will provide a generic umbrella to capture and utilize commonality, within which each new product instantiated and extends so as to anchor future designs to a common product line structure.*

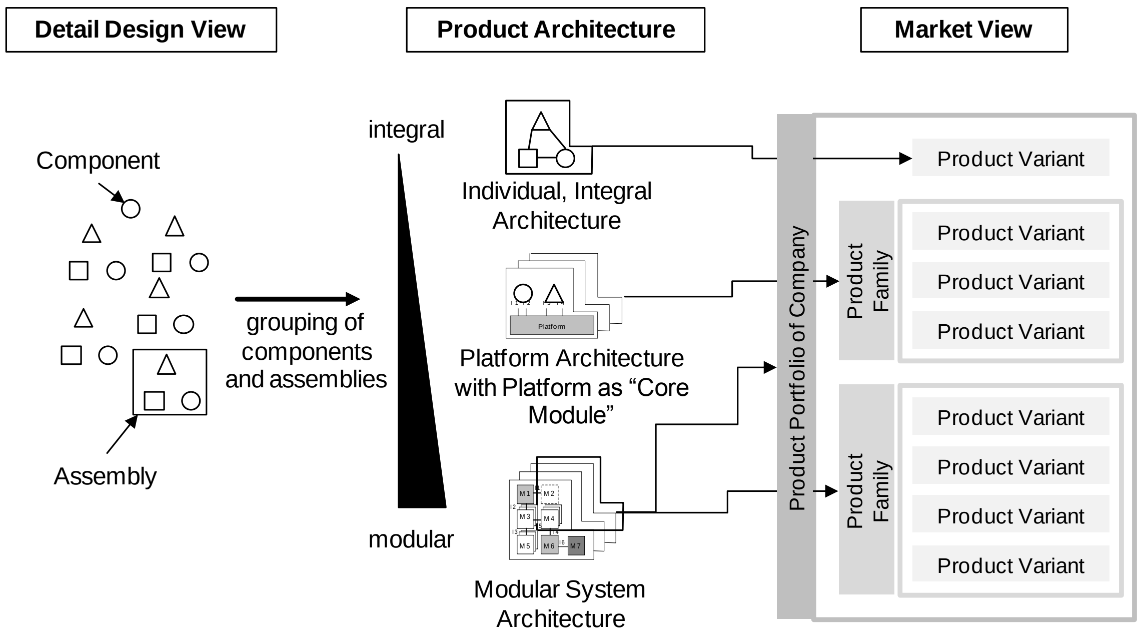


Figure 10: Different product architecture types that guide the grouping of elements in individual and product family development

As was mentioned in the previous chapter, products can have a certain degree of modularity. In other words, products can have different kinds of product architectures. This is also true for common architectures that are used across a set of product variants, i.e. a product family. Figure 10 shows an overview of different product architecture types that will be further described in this section. As can be seen in the figure, the product architecture type is the guiding principle how components and assemblies are grouped in order to derive individual products or product families.

2.3.1 Integral architectures

If the variety of products is low or if single products have to be optimised, an integral product architecture is the most advantageous type of product architecture. In such an environment, the integral architecture is quite individual and differs from product to product. In this "classical way", of developing products, it is not necessary to tediously matching modules across products (Jose and Tollenaere, 2005, p. 375).

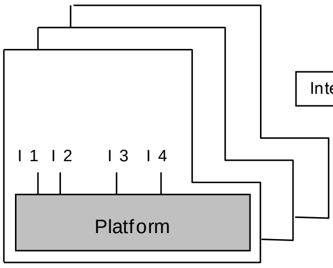
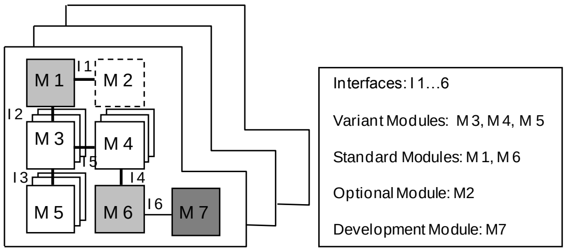
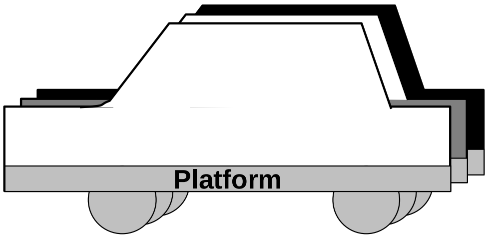
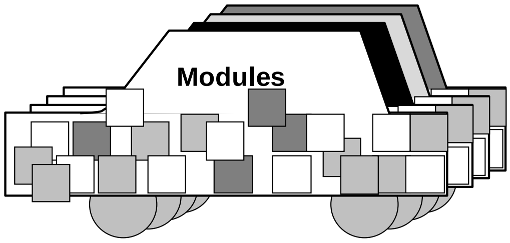
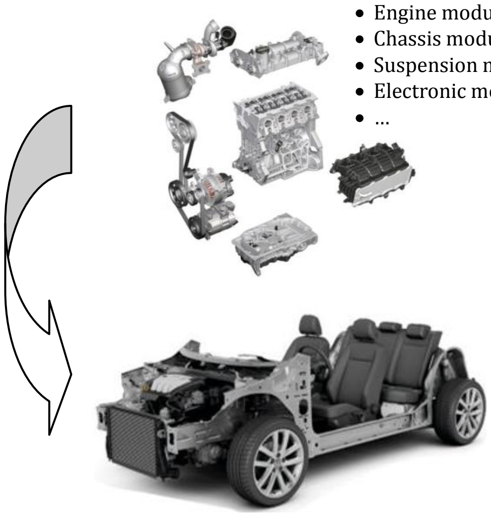
2.3.2 Platform approaches

Platform development is closely related to platforms which are known from automotive industry (Pahl et al., 2007, p. 515). In this sense, platforms are "a set of common components, modules or parts from which a stream of derivative products can be efficiently developed and launched" (Meyer and Lehnerd, 1997, p. 7). Robertson and Ulrich (1998, p. 20) relate product related entities to other entities of a company and view platforms as any asset which could be components, processes, knowledge, people and relationships "that are shared by a set of products". Although the understanding of platform is diverse and different from company to company (Kristjansson et al. 2004), the most common understanding of <u>platform</u> is the "lowest common denominator" across a set of products (Pahl et al., 2007, p. 515). Similarly, it is defined as "a collection of the common elements, especially the underlying core technology, implemented across a range of products" (McGrath, 1995, p. 39). The definition of McGrath (1995, p. 39) will be the working definition for the purpose of this thesis.

2.3.3 Modular system approach

Simpson et al. (2005a, p. 6–9) state that there are scale-based (parametric) and discrete (configured), module-based product families under the roof of modularisation. For the purpose of this work discrete, module-based product families are further considered. There are products that are modular but that do not share a common modular architecture across each other. These products do not have a generic module structure, but they may still use common modules. For the purpose of multi-product development, it may also make sense to share a common modular architecture across products. In this case, the "<u>reference architecture</u>" is a generic modular architecture (Nielsen, 2010, p. 18). It is the purpose of a so-called reference architecture to provide "an initial starting point upon which to build a new system" (Eeles and Cripps, 2010, p. 94). A modular reference architecture is the basic foundation for a "modular system". According to Pahl et al. (2007, p. 495), a modular system is a set of fixed individual modules (function units) that are combined to derive a high variety of modular products. As modular systems are a fundamental concept for this thesis, they are further specified in the next section.

Table 5: Classical platform approach compared to modular system approach

Rigid Platform approach	**Modular system approach**
	
 (Vietor and Hoffmann, 2014, p. 5)	 (Vietor and Hoffmann, 2014, p. 5)
<u>Car underbody platform</u> (Alizon, Shooter and T. W. Simpson, 2009, p. 597) 	<u>Modular car system</u> (Pander, 2012) • Engine modules • Chassis modules • Suspension modules • Electronic modules • … 

Although, frequently mixed up in literature, modular system development as approach to multi-product development is not identical to platform design, even if a "modular platform" is used. In platform-based product families, all products are based on the same platform. Products that are based on modular systems, do not necessarily share a common "core" (Arnoscht, 2011, p. 23). Moreover, "the product variants based on a platform construction are not principally configured out of predefined modules" (Pahl et al., 2007, p. 515). This difference gives a higher flexibility to module-based product families (Arnoscht, 2011, p. 23). Thus, platform approaches rather simply aim on commonality whereas modularity approaches focus on both, commonality and variety.

However, according to the definitions, common modules which are shared across products can be seen as platforms. Table 5 compares the platform approach to the modular system approach.

2.4 Multi-product development with modular systems

The previous section gave an overview of how different types of product architectures can be used to derive multiple products. It is the focus of this section to give a more detailed overview of how modularity can be used for multi-product development. Salvador (2007) presents almost 50 definitions of modularity.

Modularity

For this thesis, modularity is simply defined as the characteristic of a system to be built of modules (Arnoscht, 2011, p. 20). It can be derived that a "modular" artefact is anything that is built of modules. Thus, a modular product is created of modules. It is important to note that "modularity" of an architecture may not be the same as "modularity" of an artefact. An architecture is an abstract description whereas an artefact is a concrete, physical construct. This means that there could be slightly different definitions between "modular architecture" and "modular artefact". For instance, the description of an architecture could include functional mapping whereas the description of a product might only consist of physical elements. It is possible to say that a modular product consists of modules, whereas it is not possible to say that a modular architecture consists of modules. However, it is possible to say that a modular architecture describes how products are divided into modules.

2.4.1 Different types of modularity in modular product development

According to Ulrich (1995), there are three different kinds of modularity with discrete modules:

- Bus modularity (Pine, 1992; Ulrich, 1995)

Bus modularity uses a standard module with standardised interfaces to connect several different kinds of modules with the same type of interface. The term "bus" comes from the computer and electronics industry where a bus is used to transfer data between different components. Figure 11 shows two examples for bus modularity where different modules (M) are connected with the same bus via the same type of interface.

For instance, an extension card of a computer or an adjustable roof rack for an automobile is of bus modularity.

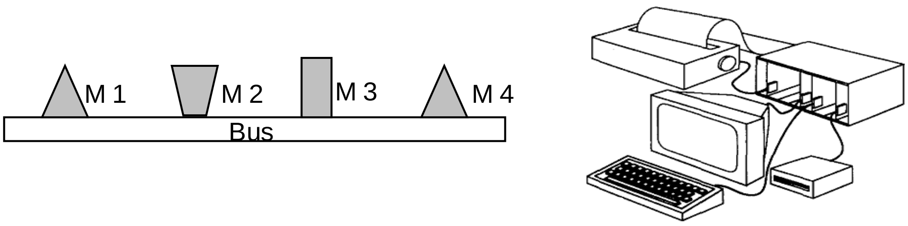


Figure 11: Examples for bus modularity (Pine, 1992; Ulrich, 1995)

- Sectional Modularity (Pine, 1992; Ulrich, 1995)

In sectional modularity, all interfaces are of the same type. In contrary to bus modularity, there is not a standard module to which all other module connect. Sectional modularity provides the greatest flexibility and variety among all modularity types. However, it is very difficult to achieve sectional modularity in practice as products are functional chains which only work if the function chain is in the right order. This type of modularity allows to combine any number of different types of components, even if the product architecture changes. Figure 12 illustrates sectional modularity where modules (M) can be freely combined via the same type of interface.

Examples for this type of modular product architecture include Lego building blocks and many sorts of piping systems.

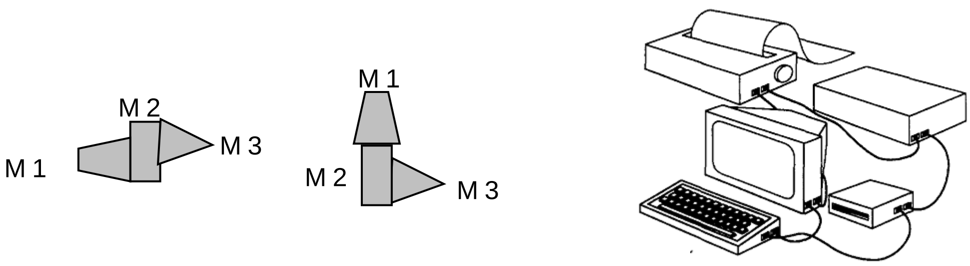


Figure 12: Examples for sectional modularity (Pine, 1992; Ulrich, 1995)

- Slot modularity (Ulrich, 1995)

In this type of modularity, each interface is different. This means that different modules cannot be interchanged and remain on the same place in the functional chain of the product. However, different variants of a module can be freely interchanged. Figure 13 shows illustrations of slot modularity where different variants of a module can be interchanged but where different modules cannot be interchanged due to different types of interfaces.

For instance, in a car it is possible to connect a seat to the defined interface for the seat and it is also possible to connect many variants of the seat as long as they have the same interface to the car. However, it is not possible to connect other modules to the seat interface. Another example for slot modularity would be a car radio.

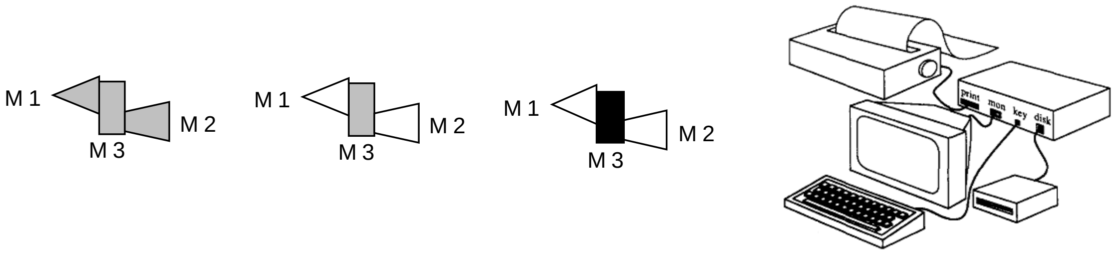


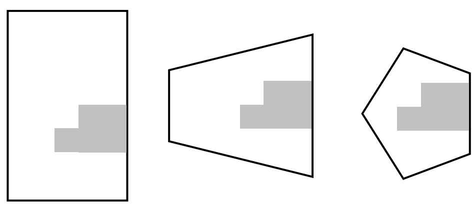
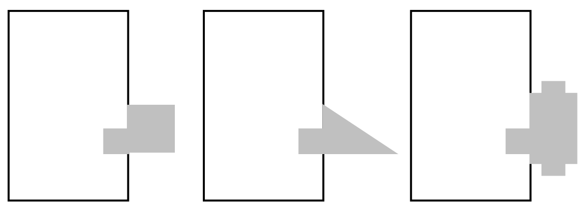
Figure 13: Examples for slot modularity (Ulrich, 1995)

Pine (1992) defines two further types of modularity which can be seen as subcategories of slot modularity (see Table 6). These two types have in common that there is only one module shared across different products whereas architecture and interfaces of the other parts of the product are not further specified. The common module is connected via an standardised interface. An accumulator of handheld power tools would be an example for slot modularity.

In "component-sharing modularity", the same standardised module is used across different products. This type of modularity is similar to the platform approach. However, the common module might not be considered as main base within this type of modularity.

In "component-swapping modularity", the same base-product is used, but slightly differentiated with different modules via a standard interface to customize the product. This kind of modularity is exactly the contrary of component-sharing modularity. Examples for this type of modularity include Swiss watches or eyeglass frames with lenses added at a local shop.

Table 6: Illustrations of component-sharing and component-swapping modularity (Pine, 1992)

Component-sharing modularity	**Component-swapping modularity**
	

2.4.2 Definitions of modular systems

What is meant with the term “modular system” is derived from the German term “Baukasten”. In literature there is no common agreement how the term “Baukasten” can be translated into English. For instance, “modular construction kit”, “modular toolkit”, “set of modules”, “modular matrix”, “modular platform” or “modular design” are just a few illustrations how many different terms are used to mean the same thing in literature.

As stated above, a system is a) a set of different elements with a defined purpose that cannot be achieved by the elements on their own (National Aeronautics and Space Administration (NASA), 2007, p. 3), b) the relationships between the elements (Hubka, 1982, p. 110) within c) a clearly defined boundary (Hubka, 1982, p. 110). “Modular” means that a physical artefact consists of “modules” (Arnoscht, 2011, p. 20).

Theoretically from these considerations, a “modular system” could have three different meanings. It is important to note that a system’s elements can be considered on various different levels:

1. It could be a set of modular products, i.e. a product family or a product portfolio with modular products and modules. For the purpose of this thesis, this case is referred to as (modular) product family or product portfolio.
2. It could be a <u>single</u> system/product/part that is modular and that can be directly given to an internal or external customer for further processing or consumption, i.e. a single modular product, assembly, etc. For the purpose of this research, this case is defined as modular product. It has to be clear that a product can be considered on many different levels.
3. It could be a set of modules that are used to derive modular products for a modular product family. This sort of system is a company-internal construct which is not obvious or directly sold to the customer. This case will be considered as “modular system” for the purpose of this work.

For the third case, a common modular reference architecture across products is required. Such a type of architecture can also be referred to as “modular system architecture”.

Pahl et al. (2007, p. 495) understand of the term “modular system” a set of fixed individual building blocks (modules) that can be combined for the development of modular products.

Kohlhase and Birkhofer (1996) define that a “modular system” consists of modules “that are selected and combined in order to configure different customized modular products”. These products can in turn be the modules for a modular system on a higher level.

Lehtonen (2007, p. 88) defines “modular system” as a system which consists of modules and which involves the interchangeability of these modules.

A comprehensive view on what is understood as modular system by different researchers is given by Arnoscht (2011, p. 25–28). Based on his review, he states that a modular system comprises modules which may consist of assemblies or components. Interfaces between modules are defined and standardised so that combinability allows to efficiently creating product variants (Arnoscht, 2011, p. 28).

For the purpose of this work, following definition of modular system is derived from literature:

- Modular system

A modular system is a set of predefined modules and respective module variants which are combined to create final product variants. The modular system is based on particular “design rules” that ensure combinability and reuse of modules. These design rules prescribe the common modular reference architecture (Nielsen, 2010, p. 18) and, thus, interfaces that need to be fixed across modules (Baldwin and Woodard, 2008, p. 20; Tiwana, Konsynski and Bush, 2010, p. 676).

- Module

A module is a component or assembly with standardised interfaces that is part of a modular system (Arnoscht, 2011, p. 30; Jose and Tollenaere, 2005; Kohlhase and Birkofer, 1996; Lehtonen, 2007, p. 88; Pahl et al., 2007, p. 495). Other studies in literature have shown that many organisations and research communities have a different understanding of modularity and modules. Additional definitional perspectives include “component commonality”, “component combinability”, “function binding”, and “loose coupling” (Salvador, 2007, p. 222). In some cases, the definition even differs from department to department within the same organisation (Hansen and Sun, 2010, p. 174). An example for the variety of definitions of modules and modularity is given by Salvador (2007). Sometimes companies use the term module just to describe their assemblies without any further intention (Arnoscht, 2011, p. 30). Such a view is not shared within this research work.

2.4.3 Classification of modular systems

Kohlhase (Kohlhase, 1996, p. 39–44) and Arnoscht (2011, p. 30–38) make a detailed literature analysis on classification of modular systems. They report that modular systems can be classified in different ways.

A first finding of Arnoscht’s (2011, p. 30–38) literature review is that modular systems can either be technical (e.g. machines), natural (e.g. molecules), or immaterial (e.g. characters and words). The clear focus of this thesis is on technical modular systems (Arnoscht, 2011, p. 31).

The main essence from Kohlhase’s (1996, p. 39–44) and Arnoscht’s (2011, p. 30–38) work is that technical modular systems can possess different characteristics. The following points give an excerpt of characteristics that were considered as relevant for the purpose of this work:

- Level of architectural fixation:

Modules can either be freely combined without being bounded to architectural requirements. For instance, Lego bricks or piping systems are relatively free from architectural requirements. In order that the modular system works, modules may also be fixed to certain architectural requirements. This could be the case for a car engine where it is necessary to meet architectural attachment, interface and design space requirements.

- Level of purity:

This characteristic defines if a modular system is purely made of predefined modules, or if there are also elements that are not predefined. An example for the second case is an injection nozzle which has a defined “body” which ensures accurate injection mechanisms. However, defining the connector to the fuel pipe could be the customer’s freedom.

- Level of free combinability:

This characteristic describes whether all possible combinations within a modular system are allowed or whether there are restrictions to combinability. This characteristic is particularly important for the customer, but also for the sales organisation of a company. For instance, when all possible combinations are allowed, the customer can customize the product himself and might have the choice between thousands of product variants. Restrictions in module combinability set by the organisation allows for focusing on most profitable products and eventually for placing those products in a product catalogue.

- User of modular system

This characteristic describes if a modular system is used by a customer or by a manufacturer. Modular systems for customers are completely delivered to customers and used by them. For instance, they could include toolkits to build a webpage or do-it-yourself construction kits. Modular systems used by manufacturers have the purpose to generate distinct products for customers (e.g. machines).

Further characteristics like the level of abstraction of a modular system’s elements (e.g. concrete module variants versus virtual models or drawings) and number of dimensions that can be altered (e.g. scalability of pipes in various dimensions versus scalability of pipes in only one dimension) will not be further considered for the purpose of this work.

Table 7: General classification of modular systems (adapted from Arnoscht (2011, p. 34))

Characteristics	Options	
Architectural fixation	free	**fixed**
Purity	**mixed**	**pure**
Combinability	without limits	**defined combinations**
User	customer	**manufacturer**

Table 7 shows an overview of how modular systems can be classified. The options in bold present the type of modular system that is relevant for this work. Thus, a modular system is defined as a) fixed to an architecture, b) consisting mainly of modules, c) defined, restricted number of combinations between modules and c) used by a manufacturer to create products for customers. What this precisely means is described in the next section.

2.4.4 Representation of a modular system

Several researchers use graphs to depict the structure of a modular system (Kohlhase, 1996, p. 28; Arnoscht, 2011, p. 37). Those graphs are usually on different hierarchy levels and describe the relations between single elements, modules and product variants derived from a modular system. Figure 14 shows an abstract example for a modular system of mixed purity.

The graph shows that products are derived from predefined modules that comprise different elements on a lower hierarchy level. In this example, elements could be components or assemblies. In addition, the products are also made of elements that were not predefined and grouped into configurable modules.

This kind of representation of modular systems is strongly connected to multi-level bill of material representation with tree graphs (Aydin and Güngör, 2005; Hegge and Wortmann, 1991).

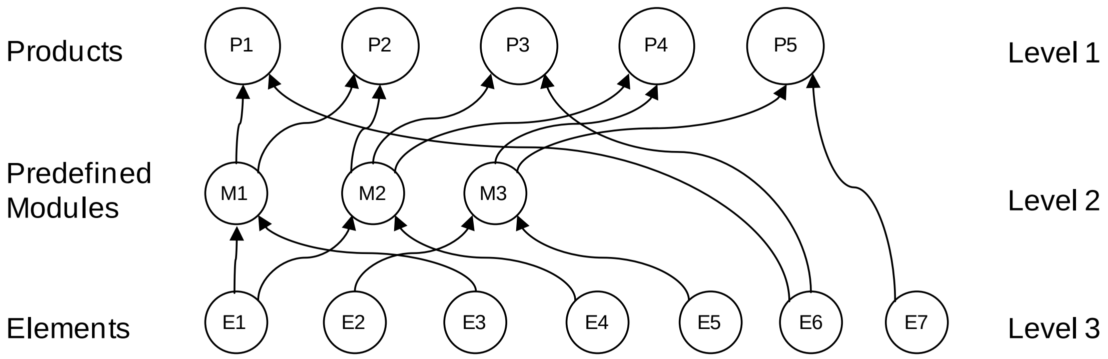


Figure 14: Depicting the structure of a modular system (Kohlhase, 1996, p. 28; Arnoscht, 2011, p. 37).

2.4.5 Summary: understanding of modular system within this work

This section described the concept of multi-product development with modular architectures. Therefore, different types of modularity in modular product development are described from the perspective of literature. After that, the concept of modular system development was explained by defining terms, classifying modular systems and showing how modular systems are represented by other researchers.

It is now the purpose of this summary to extract the essence of the literature review so far and to show in detail the understanding of a modular system for the purpose of this work.

Figure 16 takes the graph of Figure 14 and transfers it into a more detailed representation of a modular system with the support of a “Morphological Box” or “Morphological Matrix” (Zwicky, 1971, 1966). Morphological matrices can be used as design catalogues as well as overview for combining sub-solutions into overall solutions (Pahl et al., 2007, p. 94; Lindemann, 2009, p. 281). The rows of a morphological matrix represent main sub-functions of the overall solution. The columns of the morphological matrix show concrete solution principles for each sub-function (Pahl et al., 2007, p. 104). An abstract example of combining different solutions into an overall solution is depicted in Figure 15.

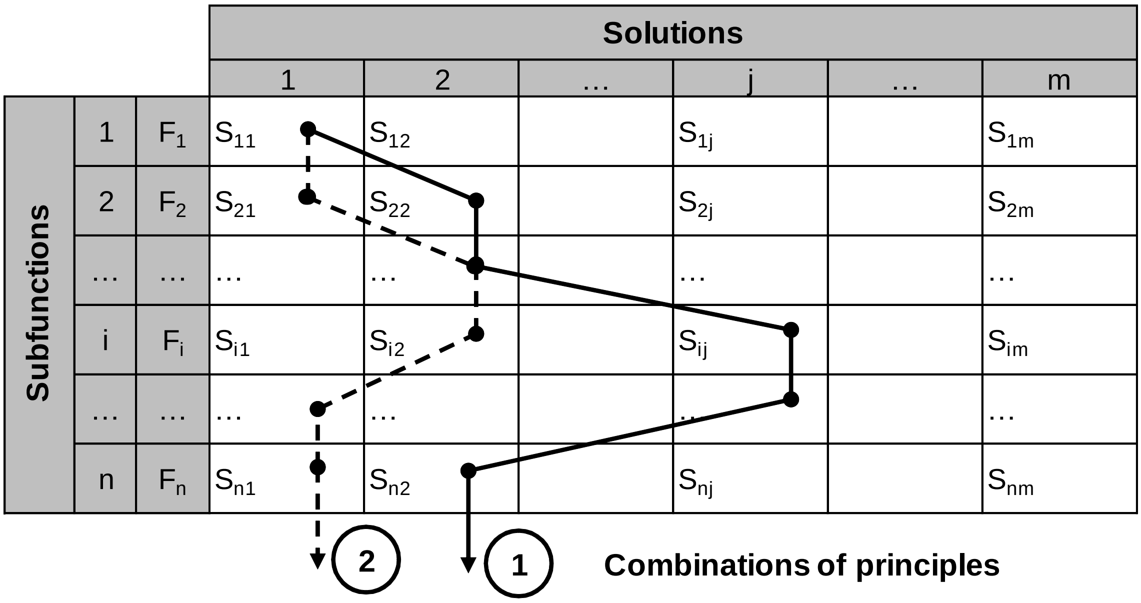


Figure 15: Abstract example of a morphological matrix (Pahl et al., 2007, p. 104)

Figure 15 shows how different sub-solutions are combined to generate a high variety of overall solutions. However, in practice the main problem of such matrices is the combinability of sub-solutions, i.e. compatibility of sub-functions or sub-solutions. In real world problems, compatibility of such systems is usually restricted. If compatibilities are reflected in the same matrix, the matrix is also named as “Compatibility Matrix” (Pahl et al., 2007, p. 105). If the number of elements and combination restrictions in a morphological matrix gets too high to depict them in one single matrix, it is also possible to represent elements, combinations and combination restrictions by computable mathematical models (Pahl et al., 2007, p. 106). For instance, overall solution 2 of Figure 15 could be described as $S_{11} + S_{21} + ... + S_{n1}$. Moreover, there could be a combination restriction between S_{12} and S_{21}.

Figure 16 shows how the given definitions, Figure 14 and the concept of a morphological matrix (see Figure 15) can be combined to clearly describe what a “modular system” is for the purpose of this work.

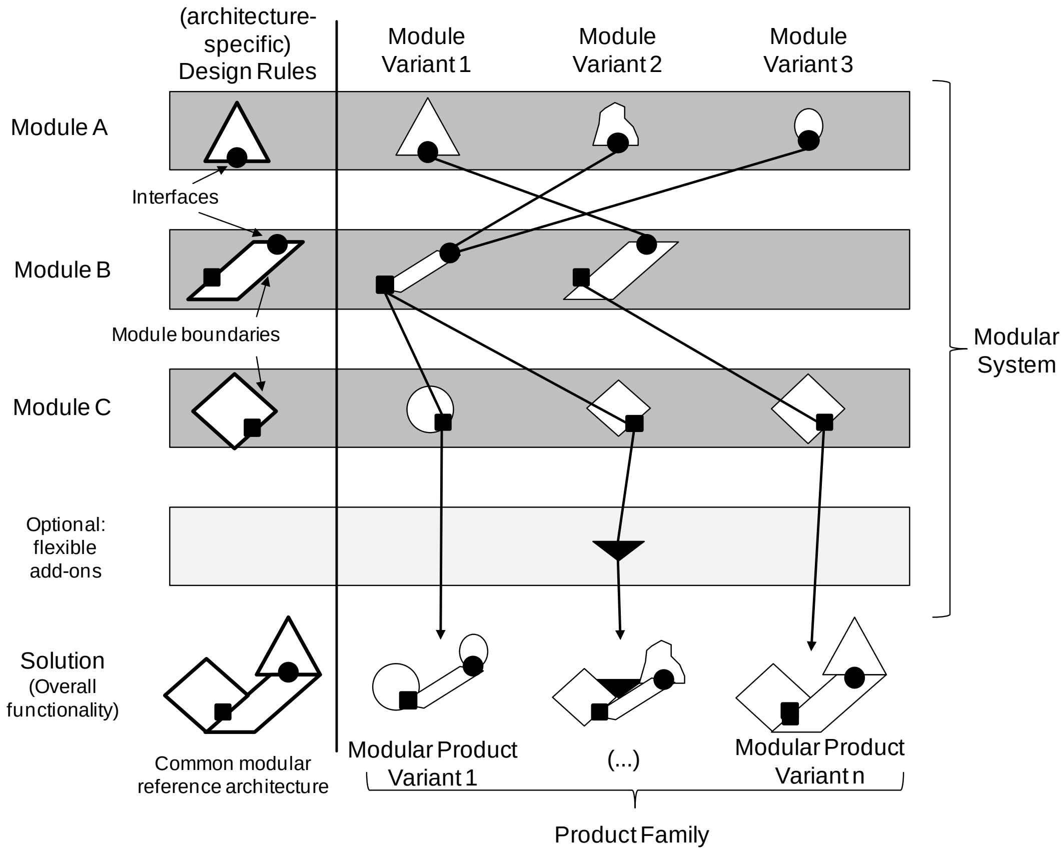


Figure 16: Understanding of a modular system for the purpose of this work

Figure 16 shows that a modular system consists of modules and respective module variants that can be combined to generate different product variants. This combination is possible because all modules are based on well-defined (product-architecture-related) design rules. These design rules specify how the modules interact and, thus, prescribe standardised interfaces and module boundaries to which all module variants of a module have to stick to. The design rules of the overall modular system make up the common modular reference architecture. All products (solutions) that are derived from the modular system are based on the same common modular reference architecture. Therefore, these products build a product family. It is also possible that the modular system is not purely modular. This means that certain carefully-selected parts of the product are not predefined, but optional. In such cases, additional functionality can be added to the products by installing flexible add-ons (e.g. components, assemblies).

The terms and definitions of Figure 16 are summed up in Table 8.

Table 8: Summary of definitions linked to a modular system

Term	Definition	Linked Reference
Modular system	A set of predefined modules and respective module variants which are combined to create final product variants. The modular system is based on particular "design rules" that ensure combinability and reuse of modules.	(Arnoscht, 2011, p. 28; Kohlhase and Birkofer, 1996; Lehtonen, 2007, p. 88; Pahl et al., 2007, p. 495)
Product family	A set of similar product variants that are based on a common product architecture to meet the requirements of a particular customer group or segment.	(Meyer and Lehnerd, 1997, p. 35)
Variant	A form or version of something that differs in some respect from other forms of the same thing or from a standard.	(Oxford Dictionaries Online, 2015)
Module	A module is a component or assembly with standardised interfaces that is part of a modular system.	(Arnoscht, 2011, p. 30; Jose and Tollenaere, 2005; Kohlhase and Birkofer, 1996; Lehtonen, 2007, p. 88; Pahl et al., 2007, p. 495)
Module variant	Concrete versions of a module with distinct characteristics.	Derived from the definitions of "Module" and "Variant"
(architecture-specific) Design rules	Design rules ensure combinability and reuse of modules. These design rules prescribe the common modular reference architecture and, thus, interfaces that need to be fixed across modules.	(Baldwin and Woodard, 2008, p. 20; Tiwana, Konsynski and Bush, 2010, p. 676).
Modular reference architecture	Same modular product architecture across the product family.	(Eeles and Cripps, 2010, p. 94; Harlou, 2006, p. 48; Nielsen, 2010, p. 18)
Architecture	Abstract description of the entities of a system and the relationships between those entities.	(Crawley et al., 2004, p. 2)

Term	Definition	Linked Reference
Modular	Characteristic of a system to be built of modules. Characteristic of an architecture which describes how a system or a set of systems is divided into modules.	(Arnoscht, 2011, p. 20)
Interface	Specification (e.g. of connections or protocols) and constraints (e.g. boundaries) that govern the relationship among modules and how they interact.	(Baldwin and Woodard, 2008, p. 7; Tiwana, Konsynski and Bush, 2010, p. 676)

Modules are the physical foundation for derivation of products from modular systems. In literature, modules are frequently further classified. Firstly, on a higher level modules are classified according to their purpose in the life cycle phase (e.g. production modules, design modules, modules in use (Kamrad, Schmidt and Ulku, 2013, p. 290; Pandremenos et al., 2009, p. 148; Salvador, 2007, p. 234)). Secondly, modules are classified according to their potential to either create variance or commonality:

- Jonas (2012, p. 6) differentiates between carryover modules, carryover candidates and variant modules.
- Alizon (2009, p. 245) distinguish between common, variant and unique modules.

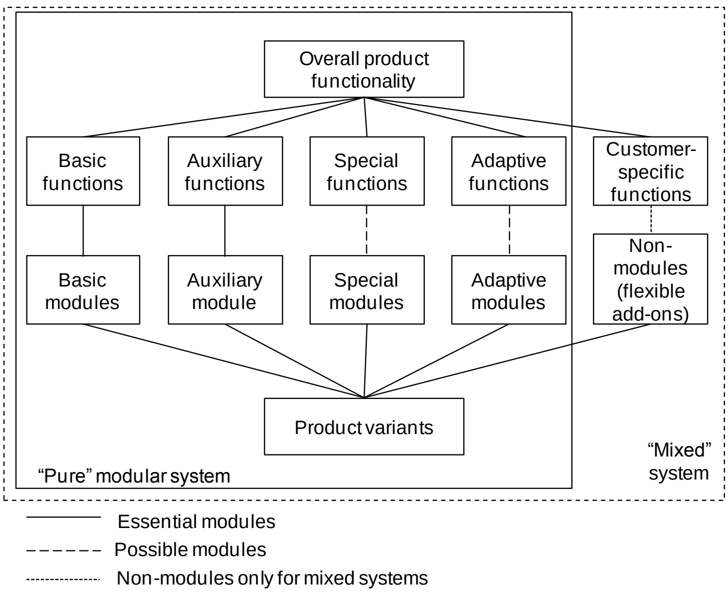


Figure 17: Different types of modules according to Pahl et al. (2007, p. 496–497)

- Pahl et al. (2007, p. 496–497) classify modules into basic modules (essential and fundamental to the system), auxiliary modules (e.g. locating or joining basic modules), special modules (e.g. optional additions or add-ons to basic modules), adaptive modules (not fully fixed, but within boundaries for unpredictable circumstances) and non-modules (to fulfil customer-specific functions). Figure 17 shows how the different types of modules and non-modules make up a "pure" modular system and a "mixed" system that contains both, predefined modules and unspecified elements (Arnoscht, 2011, p. 32; Kohlhase, 1996, p. 44; Pahl et al., 2007, p. 496–497).

So far, it was the purpose of the literature review to show how single products and multi products can be developed from an architectural perspective. It has been described what a modular system is, thus, it gets clear what it means to transition from "single product development" towards "development of modular systems" from a functional and technical point of view.

Section 2.5 will now move on to the possible benefits and limitations of establishing modular systems within companies. Therefore, it provides the rationale for companies to transition towards modular systems or to make a deliberate decision not to transition.

2.5 Benefits and limitations of modular systems

The product architecture which determines the relations between various elements on different levels of a system has various effects directly on complexity as well as on other areas of interest for a company. Various theoretical and practical benefits and limitations of modular architectures are derived in literature. In a nutshell, the main purpose of establishing a modular system is to economically create high product variety with high commonality between products (Dieter and Schmidt, 2009, p. 304; Pahl et al., 2007, p. 495). This has various effects on the product portfolio as well as on other strategic aspects (e.g. supply chain performance) of a company (Ulrich and Eppinger, 2012, p. 184). While there are potentially huge benefits when embarking upon modularisation, the side effects should not be neglected. Companies might also have good reasons not to transit towards modularisation (Hino, 2006).

2.5.1 Commonality, standardisation and reuse

Commonality, standardisation and reuse are themselves supported by modular product architectures. The usage of common and standardised modules is facilitated if modules with the same functionality have the same definition and identical interfaces. One-to-one mapping between modules and functions allows for identifying those functions that are also used by other products. Consequently, modules can be shared as common unit among different products and product generations. Moreover, decoupled and standardised interfaces make it possible to reuse the same interfaces across the product range. Standardised and decoupled interfaces are less sensitive to future chance which in turn leads to higher future commonality across modules and interfaces even if product functionality alternates over time (Ulrich, 1995, p. 431).

However, commonality, standardisation and reuse are not an end in itself. Positive effects that are of concrete interest for a company are given in the next section. It has to be noted that the benefits and limitations impact each other on different levels. For the purpose of this work, complete cause-effect relations between all factors are not given.

- Positive effects of commonality, standardisation and reuse

The usage of common modules and standardised interfaces across products and generations has positive influence on production volumes (Lau, 2009, p. 2046), general economies of scale (Kusiak, 1995, p. 261; Lau, 2009, p. 2046) and learning curve effects across the company (Lau, 2009, p. 2046).

Reusing already defined and existing modules reduces development lead time (Kamrad, Schmidt and Ulku, 2013, p. 289; Kusiak, 1995, p. 261; Lau, 2009, p. 2046) and development effort (Goepfert, 1998, p. 116; Kamrad, Schmidt and Ulku, 2013, p. 289). Simpson et al. (2005a, p. 3) similarly see the benefits of increased reusability and carryover of modules, interfaces and processes in “reduced development time and system complexity, reduced development and production cost”. In detail, sharing modules between products may result in decrease of product development lead times by 30% and in 50 % reduction in production capital investment (Muffatto, 1999, p. 148).

For example, Volkswagen claimed to save $1.7 billion per year through transitioning towards modularisation by sharing product and process elements between 19 models of its four major brands VW, Audi, Skoda and Seat (Bremmer, 1999, p. 30–38; Dahmus, Gonzalez-Zugasti and Otto, 2001, p. 409). Chrysler’s rolling chassis module supplied by Dana Corporation allegedly saved $700M in investment due to commonality when developing their new Dodge Dakota facility (Kimberley 1999 cited by Simpson et al. 2006, p.7).

The goal with new modular systems like the MQB of Volkswagen is to standardise components, dimensions and positions. For instance, the number of different positions for the engines is claimed to be reduced by 88% from 18 to two and the percentage of common components is alleged to be between 60% and 70% across 30 compact models. With the alignment of product commonalities to development, purchasing, logistics and process commonalities worldwide, Volkswagen expects to reduce assembly time per car of about 30%, and to reduce unit cost and investment cost of about 20% while still maintaining high differentiation (Hrachowy, 2011). These examples triangulate above mentioned findings of Simpson et al. (2005a, p. 3) and Muffatto (1999, p. 148).

For reused modules and interfaces, it is possible to overtake certification and testing evidence from already existing artefacts. This can significantly reduce testing and certification effort that companies have to invest prior to launching new products to the market (Simpson, 2004, p. 4).

Standardised interfaces and modules facilitate standardising production processes and other company processes. Furthermore, this also enhances flexibility to produce the same module in different plants across the world in the same quality. Hence, production volumes can be flexibly balanced across the company (e.g. for capacity, cost or sales reasons) (Muffatto, 1999, p. 145–146).

- Negative effects of commonality, standardisation and reuse

Disadvantages or threats that are entailed by standardisation and commonalities are losses in originality and uniqueness of products. Moreover, products may become less attractive due to problems differentiating the products (Goepfert and Steinbrecher, 2000, p. 6; Kim and Chhajed, 2001; Lau, 2009, p. 2046). For instance, Chrysler engineers in the 1980s were criticized for relying too much on the K-car-platform while missing to bring out innovative and distinctive products (Lutz, 1998, p. 17). Lower end models are often cannibalising higher end models if they are based on the same high-end core (Kim and Chhajed, 2001). Thus, too much commonality can damage a brand's image (De Weck et al. 2003, p.3). For instance, if brands are aligned with the same modules, customers may ask why they should buy the more expensive high-end products with no superior quality. Such concerns are expressed by customers of Volkswagen products that compare their products with congruent Skoda and Seat products. Such concerns pose a real challenge for the Volkswagen Group (Pander, 2014b; Skodaportal, 2014).

Commonality in modular products has often to be paid with reduced product performance (Goepfert and Steinbrecher, 2000, p. 6). Such compromises in performance could affect speed, efficiency, lifetime, accuracy, noise and the like (Ulrich and Eppinger, 2012, p. 189). Moreover, modular products often force companies to make compromises concerning size and mass (Lau, 2009, p. 2046).

In addition to compromises in performance, "undesirable functions can be introduced to the system, causing unexpected technical difficulties to the platform-based product family" (De Weck et al. 2003, p.3). For example, an apparently simple Audi TT rear wheel pressure problem turned out to be complex as it was traced back to the utilization of platform elements which had "unexpected side effects" (De Weck et al. 2003, p.3).

High reuse rates of modules and usage in various different products might lead to great negative impacts if components are prone with quality flaws. These high impacts of quality flaws, amongst other reasons like increasing electromechanical complexity, have led to the situation that the recall rate has reached a new record level. For instance, the number of recalled vehicles exceeded the number of delivered new cars by the factor 1,3 in the US in 2013. Manufacturers like BMW who want to double the share of platform products by 2019 even achieved a recall rate of 2,33 per delivered new vehicle. Reason for such records are, to a large extent, not the number of recalls, but the number of impacted car variants (Pander, 2014a).

High production volumes, modular architectures and standardised interfaces make it attractive and feasible for competitors to substitute modules and spare parts. This gives new opportunities for other companies to copy designs (Goepfert and Steinbrecher, 2000, p. 6; Lau, 2009, p. 2046).

Another thread of modular design is the time-consuming pre-thinking of module variants for reuse (Goepfert and Steinbrecher, 2000, p. 6). Other researchers stress the importance as well as the cost factor for platform development. In automotive industry, the cost for platform development accounts for approximately 60% (Sundgren, 1999, p. 42) to 80% (Muffatto, 1999, p. 149) of overall development cost of vehicles. Ulrich & Eppinger (2004) and Lau (2009) point out that the cost for developing a product platform can be enormous and sharing modules between low end and high end products increases variable cost due to over-sizing of standard modules.

Modular design can have adverse effects on production cost programmes like Design for Assembly (DFA) and Design for Manufacturing (DFM). Such programmes foster the minimisation of parts to be assembled by defining integrated parts. If these programmes are not aligned with modularisation initiatives, modularisation might increase direct production cost (Dieter and Schmidt, 2009, p. 302; Ulrich and Eppinger, 2012, p. 190–191).

2.5.2 Variety and flexibility

- Positive effects of variety and flexibility

Modularisation is an enabler for mass customisation which lays the foundation for economies of scope and economies of scale (Dieter and Schmidt, 2009, p. 304; Lau, 2009, p. 2046; Pine, 1992). Variety and flexibility are facilitated with combinable modules of modular architectures. Predefined modules with specified functionality and standardised interfaces create the possibility to replace modules with modules that have different parameters or specifications, but the same general functionality and identical interfaces. The combinability of standard and variety modules allows for creating a huge amount of final product variants to satisfy diverse customer requirements based on a basic set of modules. This can be done with relatively low development and production effort (Erens and Verhulst 1997, p.170).

- Product Variety:
  With established modular architectures, it is possible to increase the range of different products that can be delivered to the customer (Ulrich and Eppinger, 2012, p. 188). Manufacturers can develop "differentiated products efficiently, increase the flexibility and responsiveness of their manufacturing process, and take market share away from competitors that develop only one product at a time" (Robertson & Ulrich 1998, p.20). In the 90's, "platform" car manufacturers achieved a 5.1 percent growth in market share yearly compared to a decline of 2.2 percent of those companies that did not apply platform strategies (Cusumano and Nobeoka, 1998; Simpson, Siddique and Jiao, 2005a, p. 4).

- Product Change and Flexibility:
  Companies also achieve greater flexibility for future product generations by carry-

ing over several modules from previous generations instead of newly developing the whole product. In turn, new products which have carried over modules achieve market maturity much faster. Expected or planned changes to one or several product functions over time or in the next generation can be accommodated by single modules instead of changing the whole product (Ulrich, 1995, p. 436; Pahl et al., 2007, p. 509). Modular architectures can respond quickly to changes in styling such as colour or shape if those components with the same styling life cycle are grouped into a module which can be easily interchanged (Erixon, 1998; Stake, 2000). Moreover, following general change benefits that are particularly facilitated with modular products fall into this category (Hansen and Sun, 2010; Kamrad, Schmidt and Ulku, 2013; Lau, 2009; Ulrich and Eppinger, 2012):

- Upgrades: Upgrading single modules of a product like the storage disc, graphic board or software of a personal computer
- Add-ons: Some modules might be added on to a rather standard product (e.g. IPhone apps, pollen removal filter in a car)
- Adaptions: Incorporating switch mechanisms from a gasoline system to a propane fuel supply in distinct modules
- Wear: Placing wear and tear components in a module that can be easily replaced (e.g. replacement of carbon brushes in a motor)
- Consumption: Making frequently consumed materials easily replaceable through changing a single module (e.g. printer cartridge).
- Flexibility in use: For instance, accumulators might be used in different power tools

With modular systems, it is possible to implement these changes with one single module, without big redesign effort for the whole product.

Moreover, modular architectures are also suitable for unforeseen changes. Robustness to change of the whole product architecture is enabled by de-coupled interfaces. This means that changes to certain modules do not affect other modules which in turn means that the impact of change to the product is relatively low (Erens and Verhulst, 1997, p. 170). Therefore, modular product architectures are suitable to incorporate <u>uncertainty</u>.

- Negative effects of variety and flexibility

In most cases, high variety and flexibility of products is a desired goal of companies. However, there are several points that have to be considered when the transition towards modular system development is undertaken.

Firstly, higher variety and flexibility have to be paid with more <u>expensive pre-planning</u> of product variants and <u>initial investment</u> in standardised interfaces and modules (Goepfert and Steinbrecher, 2000, p. 6; Riepe, 2003, p. 39).

Even though, it is the goal of modularisation to increase variability, the degree of achievable variety is limited to exchanging and adding modules. With modularisation, it is not always possible to respond to very special wishes economically. The contrary of a discrete module-based product family would be a scale-based product family with unlimited variety along scalable dimensions or a totally customised design with an integral architecture (Pahl et al., 2007, p. 509; Simpson, 2004, p. 5).

Finally, the potential of a modular system to generate variety and commonality depends on its composition of standard and variety modules. An emphasis on standard modules could lead to higher commonality. An emphasis on variety modules could lead to higher variety. Combining a number of discrete variety modules could lead to a tremendous number of final product variants. In any case it has to be considered that high product variance is not an end in itself. It is also possible that inappropriately high product variability has detrimental effect on the demand of customers. For instance, if a customer has too many choices, her/his decision becomes too complex and she/he might consider buying a competitor's product with a straight-forward buying decision (Cutrone, 2013; DeAngelis, 2004; Mael, 2014; Schwartz, 2005; Zoltan, 2014). In sum, what is true for so many other things could also be true for variety – more is less.

2.5.3 Strategic aspects

The general advantage of modular products is that modules can be grouped for certain strategic reasons into the same building block. In such a module, all parts have ideally the same strategic intend.

Change to the product over its life cycle can be achieved by establishing a modular architecture that groups components with similar life cycle properties into the same module. Moreover, standardised and decoupled interfaces allow to quickly exchanging modules. From the life cycle view it is beneficial that modules can be created to effectively support following life cycle motives: maintenance, repair, service, reuse, recycling, and disposal (Kamrad, Schmidt and Ulku, 2013, p. 289; Kusiak, 1995, p. 261; Lau, 2009, p. 2046; Newcomb, Bras and Rosen, 1996; Ulrich, 1995, p. 427). For instance, modules can be easily changed in case of any defects or recycled based on material compatibility.

The modules of a modular product architecture can be tested separately. This separate testability is enabled by the functional independence of modules and standardised interfaces. In case of changes to the functionality of the product, it is possible to test the module which incorporates the change instead of the whole product. Moreover, with this strategy, defects of modules are detected before they are built into the final product which reduces quality related losses. However, defect-free modules do not replace tests for end product variants as defect-free modules do not guarantee defect-free product or system functionality (Goepfert and Steinbrecher, 2000, p. 6).

Modular product architectures allow for aligning product structures, organisational structures and process structures (Goepfert, 1998; Oosterman, 2001; Persson and Ahlstrom, 2013; Sanchez and Mahoney, 1996). Functional independent modules and decoupled interfaces reduce the need for information exchange between developers of different modules. This allows for working on different modules independently and simultaneously (Lau, 2009, p. 2046) which "permits overlapping activities and reduces the length of the design process" (Smith & Reinertsen 1991, p.101). Moreover, modularisation makes it possible to assign specialised teams to development of special/highly innovative modules (Lau, 2009, p. 2046). For instance, Daimler made tremendous investments for aligning its organisational structure to its module structure (Daimler, 2014).

Modularisation makes it possible to integrate strategic module suppliers that independently develop and produce certain modules. Such a concentration on core-competencies and specialisation also enhances the technological development and know-how accumulation for modules (Baldwin & Clark 2000). Independent modules also offer the opportunity to reduce manufacturing lead time through concurrent internal and external production processes. In the ideal case, this can be achieved through modularity in production (Jacobs et al., 2011).

With the help of modular product structures, it is possible to shift the point of variance creation towards the end of the supply chain. This means that cost can be reduced by applying principles of mass production until a late point where actual variety is generated. This principle widely known as postponement (Ernst and Kamrad, 2000; Feitzinger and Lee, 1997).

2.5.4 Effect on cost and profit

The above mentioned benefits and limitations of modularisation have in turn a more or less direct effect on cost and profit of an organisation (Hansen and Sun, 2010). For instance, faster delivery times might increase sales, reused modules might decrease cost in production and general overhead cost. On the other hand, high commonality might compromise product performance, thus, this might decrease the willingness of customers to pay appropriate prices.

Literature has not presented a universal model that makes it possible to directly list financial implications of modularisation. However, researchers have shown that there are factors that indicate the financial effects of modularisation. (Fixson, 2005; Martin and Ishii, 1996). The outcome of such analyses are more or less estimations of the effects of modularisation. Such estimations can be conducted before and after modularisation transition.

More recent advancements have come up with activity-based costing systems that measure the actual effects of modularisation throughout the company (Park and Simpson, 2008; Thyssen, Israelsen and Jørgensen, 2006). Such cause-effect relationships between benefits, limitations, cost and profit have to be established situation-specific for each considered company. It has to be considered that such systems only come up with reliable financial figures after transitioning towards modularisation and that implementation of such measurements requires a tedious and revolutionary shift from traditional costing systems towards activity based costing.

2.5.5 Overview: Benefits and limitations

In principle, the strengths of a modular architecture are the weaknesses of an integral architecture: Providing relatively high variety, flexibility and changeability with high commonality and reuse.

This is of high importance for companies with complex product portfolios which want to save cost without reducing variety provided to the customer. High product commonalities and modular structures have a beneficial impact on unit cost, investment cost as well as on many other processes of the company. Some of them are depicted in Figure 18.

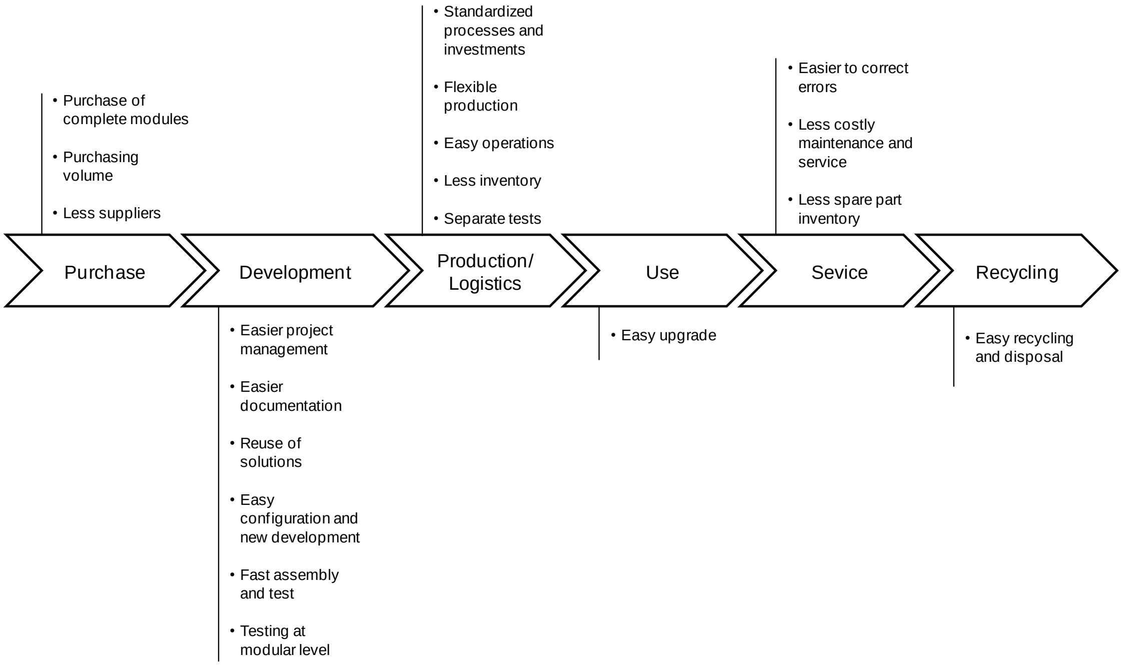


Figure 18: Effects of modularisation (Blees, Kipp and Krause, 2010; Miller, 2000; Rathnow, 1993; Schuh and Schwenk, 2001)

On the other hand, the strengths of an integral architecture are the weaknesses of a modular architecture. Product programs with low variety and high volumes should not be realised with modular architectures (Erixon, 1998). In integral architectures, interfaces can be reduced or optimized by integrating parts or by creating un-detachable interfaces. This means that if single products have to be optimized regarding cost, integral product architectures might be more suitable (Smith and Reinertsen, 1991, p. 101). Accordingly, integral product architectures are the better product architecture if the performance of single products has to be optimised. The common elements of modular architectures are often compromises between several different product variants which can have a negative impact on the performance of single products. Moreover, interfaces may become the weak point of modular architectures. For instance the interface of legs to seats of wooden chairs often become the weak link or the performance of electronic circuits decreases if the electronic modules are separated (Smith and Reinertsen, 1991, p. 101). Another disadvantage of modular architectures is that their weight and structural dimension requirements are greater than those of integral architectures. Moreover, if very special customer wishes have to be met, an integral architecture might be the preferable option (Pahl et al., 2007, p. 509).

In the introduction at the very beginning of this chapter, it was pointed out how complexity is defined for the purpose of this work. According to the definition of complexity, modular architectures are indeed a way to reduce a company's complexity (Simpson, 2004, p. 4). It has been shown that modular architectures reduce (1) the variety and overall number of elements in a range of products by drawing upon common elements, (2) the intensity and diversity of interactions by using standardised and de-coupled interfaces across different products, and (3) system dynamics by using a functional-independent decoupled architecture which is less sensitive to change over time. Once, the complexity is reduced in these dimensions, this has benefits not only to the cost structure of the product but also to the complexity related activity-based costing for all processes of the company (e.g. administration and general overhead cost). It has to be stated that the effort that is needed for modular product architecting is only justifiable if a large variety of customer requirements has to be flexibly satisfied, the customer is willing to pay for the variety or if other benefits such as described in Figure 18 are achieved. If this is the case, a modular system is more cost-efficient than a specially-designed product with an integral architecture (Pahl et al., 2007, p. 509–510).

Having discussed the potential benefits and limitations of modular design, the thesis will now move on to present how literature recommends establishing modular architectures. To get an understanding how product architectures are established during the development process, the next section will bring product architecture related issues into the context of the engineering design process.

2.6 Product architectures in the engineering design process

This section will briefly introduce to the principles of the engineering design process and show where modularisation is placed within this process.

To understand the implications of product architectures with the engineering design process, it is necessary to reflect upon the general design process as a whole. Finger and Dixon (1989) distinguish between descriptive design models and prescriptive design models.

Descriptive design models describe how designers create designs and what techniques, processes, methods and strategies they use. This is done by collecting data about the designer's behaviour or by creating cognitive models that describe mental behaviours of designers. Some descriptions follow intuitive sense while others are based on formal observations of the design process (Finger and Dixon, 1989).

Prescriptive design models can also be categorized into two categories. The first category describes how the ideal result of the design process should be and the other category describes how the ideal design process should take place (Finger and Dixon, 1989).

A prominent example for the first category of prescriptive design models is Axiomatic Design. Axiomatic[2] Design prescribes "a fundamental set of principles that determine good design practice" (Suh, 1990). In short, Axiomatic Design expresses that "good design meets its various functional requirements independently and simply" (Finger and Dixon, 1989, p. 56). Of special interest for the design of product architectures in this sense, is the mapping from the functional domain to the physical domain and the complexity of interfaces. It can be derived that modular design with functional independent physical modules and standardised interfaces meets the ideals of Axiomatic Design to a large extend (Dieter and Schmidt, 2009).

A large volume of research has also been published on the prescription how the new product development process (NPD) should ideally be. Predominant publications prescribe the design process as a chronological, linear process combined with iteration cycles. Such prescriptive models are, for instance, given by Cooper (2014), Pahl et al. (2007), Dieter et al. (2009), Roozenburg and Eekels (1995), Lindemann (2009), Ponn and Lindemann (2008), Ulrich and Eppinger (2012), Ullman (2003), and the Association of German Engineers with the VDI Guidelines VDI 2221 (Verein Deutscher Ingenieure, 1993) and VDI 2206 (Verein Deutscher Ingenieure, 2004a).

2.6.1 The new product development process (NPD)

A study of Sharafi et al. (2010) came to the conclusion that the process phases of the different models are similar and that the phases are on the same position. Moreover, a main finding of their study is that the task specifications from various design stages can be clustered into three product development domains: Product Concept, Product Design and Production Design ((Sharafi et al. 2010, p. 1733) in accordance with (Krishnan and Ulrich, 2001)).

For the purpose of this work, actual production design phase is discarded and the requirement domain is taken as separate development domain (following (Lindemann, 2009; Pahl et al., 2007; Suh, 1990; Verein Deutscher Ingenieure, 1993)). Nevertheless, production needs in combination with other relevant life cycle needs are considered in all phases of product design. The task of each resulting phase can be described as follows:

---

[2] "Axioms are fundamental truths that are always observed to be valid and for which there are no counterexamples or exceptions. Axioms may be hypothesized from a large number of observations by noting the common phenomena shared by all cases; they cannot be proven or derived, but they can be invalidated by counterexamples or exceptions" (Suh, 1990, p. 47)

- Requirement Phase

The aim of this phase is to understand the needs and requirements of the customers, to identify internal requirements for the design problem and to identify overall market trends (Dieter and Schmidt, 2009; Pahl et al., 2007). This understanding is brought into correlation to the strategy and plans of the company (Pahl et al., 2007). After deep analysis of all factors that influence the requirements for the design artefact, the product design specification or in other words, the requirements list has to be created and officially accepted. The requirements list sets the course for all succeeding activities and has to be adjusted and managed throughout the whole product development process (Lindemann, 2009, p. 44–45).

- Concept Phase

Functions that the product has to fulfil are defined in this phase in order to satisfy functional requirements from the requirements list. The result of the functional analysis is the functional structure of the product (Verein Deutscher Ingenieure, 1993). This allows for determining the cause-effect-relationships for each product function (Lindemann, 2009, p. 45). As soon as the basic principle behind each product function is clear, the designers can start to screen, evaluate and select solution concepts for the implementation of product functions (Dieter and Schmidt, 2009). The result of this stage is the conceptual solution of the product.

- Design Phase

The design phase requires the product architecture to be established. This means that the overall system has to be divided into modules already at this stage. To which phase establishing the product architecture is assigned to, is handled differently by different researchers. The setup of the product architecture is either assigned to concept phase (Stone, 1997), to embodiment design (Dieter and Schmidt, 2009) or to a design phase called "system-level design" (Ulrich and Eppinger, 2012). The actual design phase for this work comprises embodiment design and detail design of the product:

- *Embodiment design* concerns sizing, configuring and parameterising modules according to the specifications of the overall system (Dieter and Schmidt, 2009).
- The purpose of the next phase, *detail design*, is to generate a complete design description of a producible and tested product (e.g. engineering and assembly drawings, bill of material, and verified test protocols) (Dieter and Schmidt, 2009).

Figure 19 gives a summarising overview of the linear design process in a slightly finer resolution (VDI 2223).

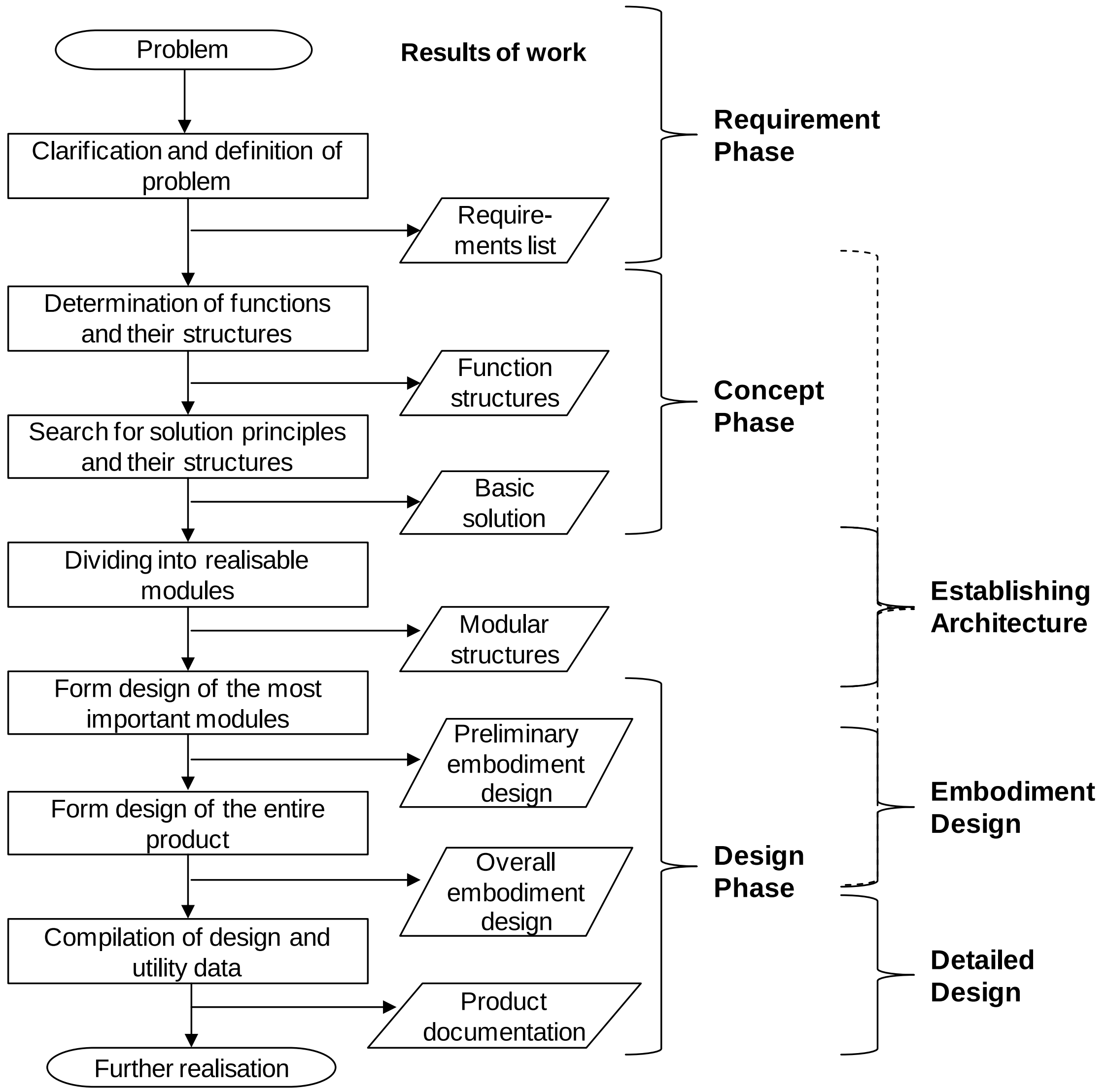


Figure 19: General procedure of systematic design (Verein Deutscher Ingenieure, 2004b, p. 5)

2.6.2 Establishing architectures in the design process

Figure 19 shows product architecture definition as an own step in the systematic design process ("dividing into realizable modules"). This view is shared by Dieter and Schmidt (2009) who also see product architecture definition as an own step. However, Dieter and Schmidt assign the step "Establishing Architecture" within embodiment design of the design process. Other researchers establish product architectures already during concept phase (McAdams, Stone and Wood, 1999; Meehan, Duffy and Whitfield, 2007; Stone, 1997).

In contrast, Ulrich and Eppinger (2012) and Pahl et al. (2007) see it as a phase overarching process that ends just before design phase. Ulrich and Eppinger (2012) suggest to start at the late requirements phase and to end during the design phase. Pahl et al. (2007, p. 499–508) go even further and assign product architecture relevant activities from task clarification to detail design. Architecting as phase-overarching activity is indicated in Figure 19 by the dotted bracket. The excerpt of the process model of Ulrich and Eppinger (2012, p. 9) is depicted in Figure 20. This figure shows a more recent product development process model. It gets obvious from the figure that the product architecture has to be established on system or product level before detailed design. This makes it possible that the defined modules can be designed in detail independently.

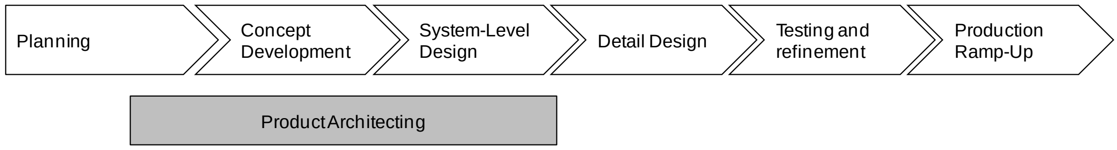


Figure 20: Product architecting in the product development process of Ulrich and Eppinger (2012, p. 9).

The right process phase for modularisation is indeed controversially discussed in engineering design literature. On the one hand, it is desirable to establish the architecture as early as possible. On the other hand, if the product architecture is established too early in the design process under incomplete information, it might "fail in meeting the constraints that become apparent later in the design process" (Kusiak, 1995, p. 261). To date, it seems that there is still no agreement on the right process phase for modularisation. Thus, this issue has to be resolved situation-specific. The next chapter will shed more light on this issue and will give an overview in which design phase other researchers start to establish product architectures.

The tasks of each phase of the design process are not processed in a linear, rigid and subsequent manner. An important feature of the whole design process which is also valid for the product architecture process is the iteration between later and earlier stages (Roozenburg and Eekels, 1995; Verein Deutscher Ingenieure, 1993). This has the purpose of constantly gaining knowledge and improving the design artefact stepwise. Moreover, Dieter and Schmidt (2009) also describe the need for constant iterations between design and design evaluation steps. Such an iteration between design action and design review provides a method for identifying problems early and, thus, improving the quality of design through preceding action phases. In addition, planned design reviews also provide a solid base for planned knowledge capture and lessons learned (Dieter and Schmidt 2009).

To accommodate the iterative feature of the product architecture design process and to get a deep understanding of what other researchers do to establish and review product architectures, the next chapter provides an extended overview about product architecture methods.

2.7 Summary

It was the purpose of this chapter to present the basic foundations and research rationale for this thesis. The chapter showed what is already known about product architectures and modularisation.

Product architecture is an abstract construct that describes the entities of a system and the relationships between them. When literature analyses products in terms of their modularity, the most frequently used entities of the product to describe their architecture are functional and physical elements. An architecture is defined as modular if there is 1:1 relationship between functional and physical elements, if interfaces are de-coupled and if relationships inside a module are greater than relationships between modules. On a purely physical level, a product can be defined as modular, if it consists of modules.

In order to fight product and process complexity in companies, single modular products are of no means. For the purpose of complexity reduction, modularisation is only a strong lever if it is applied on product family level or broader. As an extension of the platform concept, modular system development enables companies to generate both, high product variety and internal commonality by combining predefined modules. Besides high variety and decreased complexity, modularisation has numerous additional benefits. However, it has been shown that the benefits cannot be harvested for free. When a company wants to exploit the potential of a modular system, it first has to carefully look at limitations and disadvantages of modularisation as well.

Nevertheless, the literature review has shown that from a theoretical perspective, the transition towards modular system development can indeed be seen as a very promising way to fight the complexity problem that is increasingly pressurising industry (e.g. see Section 2.5.5).

The design of modular product architectures requires special attention during the development process, either as phase-overarching activity or as individual major phase. It will be the purpose of the next chapter to show how literature deals with establishing product architectures in detail during product development.

3 State-of-the-art: modularisation development process support and industrial examples

This chapter analyses architectural considerations in the engineering design process in detail. The main focus of the research reviewed is modular product development with modularisation methods as its essence (Gershenson, Prasad and Zhang, 2004, p. 39). Therefore, this chapter will present and review methods that are proposed by literature to establish modular product architectures (see Section 3.1). In order to get a comprehensive overview of existing modularisation approaches, they are characterised and related to the product life cycle in Section 3.2. To understand the relationship between theory and practice, this chapter also contains a review of product architectures in industry (see Section 3.3). Finally, the chapter closes with a summary of state of the art and the need for further research. Figure 21 shows the elements of this chapter.

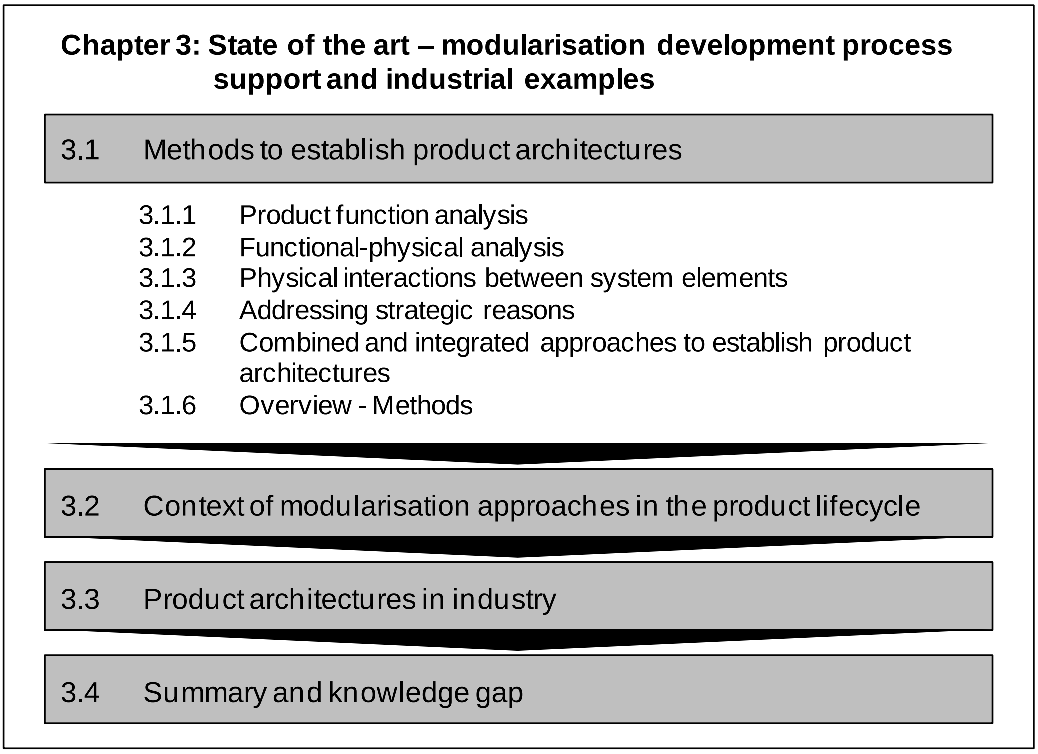


Figure 21: Elements of Chapter 3: State of the art of modularisation support and examples from industry

3.1 Methods to establish product architectures

A report published by the Technical University of Denmark in 2001, highlighted the need "for a comprehensive design methodic, using a design language that leads to efficient product models, and bring in both the proper understanding of the modularisation effects and the possibility to combine more types of modularisation. This, in turn should lead to effective design operations for creating product variants" (Andreasen, McAloone and Mortensen, 2001, p. 46). Holtta and Salonen (2003) also published a report which points at the paucity of modularisation methods and the lack of knowledge when and how to use them. In the meantime, a large number of modularisation methods and tools which are applied on a wide variety of issues have emerged (Simpson et al., 2014; Simpson, Siddique and Jiao, 2005b; Jiao, Simpson and Siddique, 2007).

The previous chapter explained what modular product architectures are, it introduced different types of modularity and discussed the effects of it. This section follows on and presents a literature review about methods, which are applied in the engineering design process, to establish modular product architectures. On the one hand, the development of modular products is supposed to be the "heart" of research into modularity and on the other hand it is fraught with difficulties and problems. For this reason, "methods for developing modular products are essential" (Gershenson, Prasad and Zhang, 2004, p. 39). Therefore, literature from the English- and German-speaking world about methods establishing product architectures was extensively reviewed. The methods were identified by consulting standard literature, screening databases that are used to store literature about engineering design (e.g. Compendex, Web of Science, IEEE Xplore, SAE Digital Library and Scopus), visiting libraries, visiting conferences and by using contacts to universities, industry and consultancies.

"Methods" are descriptions of a rule-based and planned action process. Methods are prescriptive, target-oriented, consciously applied and operational while being related to an action (Lindemann 2006; Roozenburg & Eekels 1995). An action in this meaning is the "intervention in the autonomous transformation process of a system." Through the intervention, the system is led into another direction, compared to not intervening (Roozenburg and Eekels, 1995, p. 40). The difference of a method to a process model is its formal description, its operational character and the description of "how" to perform the processes (Lindemann 2006, p.58).

The value of applying methods in the design process becomes evident when looking at their advantages. Methods are means of support to handle complex problems which product design tasks usually are. They help to break down complex problems into smaller sub-problems, to identify target conflicts and focus working areas (Lindemann 2006, p.58). Moreover, methods coordinate processes between individuals and support effective handling of information and knowledge. For instance, methods incorporate traceable documentation of decision processes which facilitates the knowledge management between different departments and projects (Lindemann 2006, p.59). The general development risk, i.e. achieving targets without any unplanned fallbacks, is also minimized by applying methodologies (Lindemann 2006, p.59).

Due to the large quantity of product architecture methods found in literature, a classification scheme was set up. Amid many different possibilities to classify product architecture methods described in literature (Daniilidis et al., 2011; Jiao, Simpson and Siddique, 2007;

Koeppen, 2008; Koppenhagen, 2004; Meehan, Duffy and Whitfield, 2007), the next sections are classified and subdivided according to the factors that are considered during product architecture design. These factors are assigned to different product life cycle phases in section 3.2 later on.

3.1.1 Product function analysis

Functional analysis is an important tool to solve design problems by abstracting the task which a system or product has to fulfil (Lindemann, 2009, p. 267). There are two different research streams considering functional relations during product architecture design. The first research stream analyses functional elements and the second research stream analyses functional flow structures, functional structures or functional hierarchies. Product architecture methods that take use of functional analysis draw upon general principles of good design (Suh, 1990) and upon the general principles of product architectures (Ulrich, 1995). Both research streams have in common that their methods can be applied independently without having technical solutions on hand. This is of special interest in the early concept phase of product development.

- Functional element analysis

Functions are grouped into modules for reasons of functional independency which can be seen as a characteristic of modular design. This means that changing a function of the product (e.g. triggered by the customer) or system is facilitated by only changing one module instead of changing several distributed parts of the product. Another point is that functions that are identified as “common” function for a wider range of products support communality within product families. On the other hand, functions that are identified as “variant” can be grouped into variant or optional modules.

Dividing the function structure into “common” elements and into “variant” elements for the product variants of the family allows for a convenient derivation of product variants by combining common and variety elements of the product family (Pahl et al., 2007; Siddique and Rosen, 1999; Zamirowski and Otto, 1999). The categorisation of functional elements can be further specified by defining basic, special, auxiliary and adaptive functions from which modules or non-modular parts of the product can be derived (Pahl et al. 2007). Such approaches are usually done visually on graphs and, thus, can be seen as graph-based functional module clustering.

Product architectures can also be established by analysing several functions at once or chains of functions for commonality. McAdams et al. (1999) divide functional structures into “common flow function chains[3]”, “causally linked but flow independent function

[3] Common flow function chains: Every function works on a common basic flow (McAdams et al. 1999, p. 7)

chains[4]" and into "independent, non-causal function chains[5]". Afterwards, they identify modules based on these classifications. For instance, common flow function chains are candidates for standardised modules whereas independent, non-causal function chains should not be grouped together into modules.

Dahmus et al. (2001) add a modularity matrix subsequent to graph-based functional module clustering (see Figure 22). The modularity matrix is used to relate common and variety functions of the functional structure to the products of the product portfolio. The matrix can also be used to compare qualitative characteristics and target values between the product variants of the portfolio. For instance, the modularity matrix shows whether "Motor A" or "Motor B" is used in a certain product to fulfil the function "Convert Electricity to Motion". This tackles the shortcomings of other approaches of just comparing sub-functions without looking at the target values behind the sub-functions.

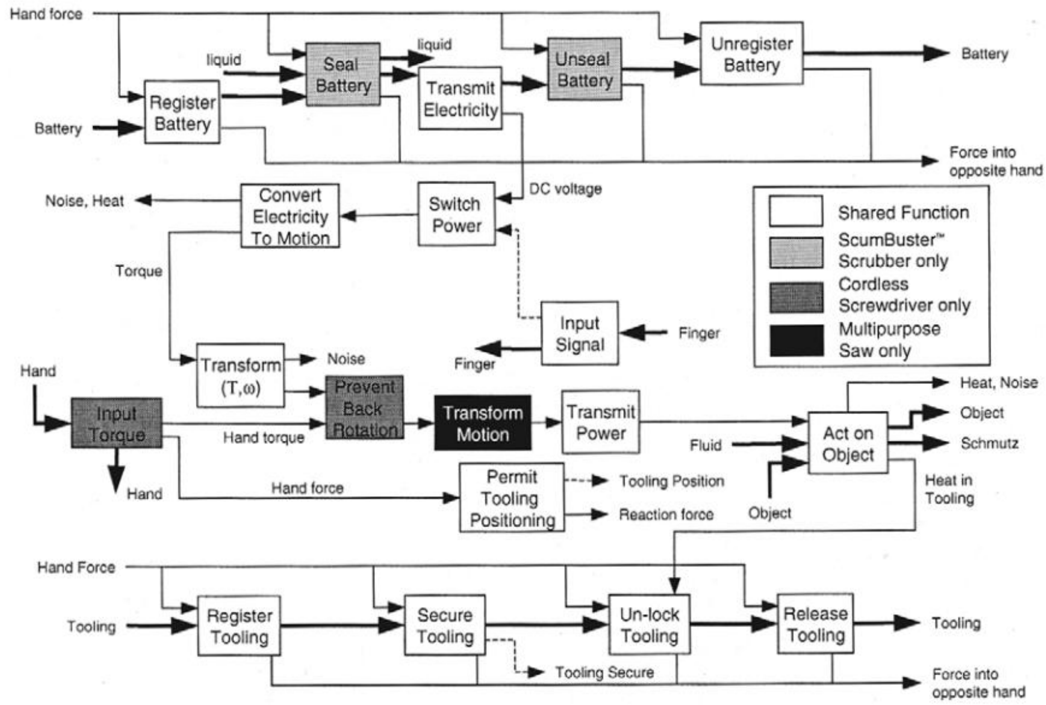


Figure 22: Generic function structure fort the product portfolio of exemplified power tools (Zamirowski & Otto 1999; Dahmus et al. 2001)

- Functional flow analysis

Modules which are created based on functional flows are built because less functional flows between modules indicate fewer interfaces between the modules. This characteristic

4 Causally linked but flow independent function chains: The functions work on an obvious flow link even they do not work on the same link (McAdams et al. 1999, p. 7)

5 Independent, non-causal function chains: Functions which work separately from each other, share no common flows and have their source in different customer needs (McAdams et al. 1999, p. 7)

supports independency of different modules from each other within the product or the product portfolio.

Holtta et al. (2003) group functional flow chains into modules based on the similarity of in- and outputs of its functional elements. To derive modules from the functional flow structure, Stone (1997) and Stone et al. (2000, 1999) develop heuristic[6] rules. The heuristic rules suggest to group functional dominant flows, separate branching flows and functions with conversion-transmission functionality into modules (see Figure 23).

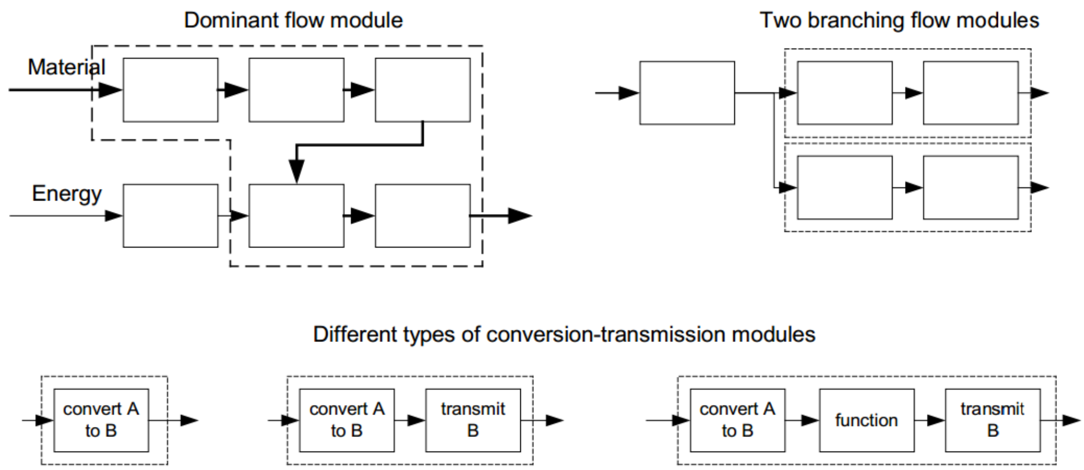


Figure 23: Three heuristic rules according to Stone (1997), figure according to Blackenfelt (2001, p. 57)

Even though, there is no theoretical proof why the heuristic rules should identify an improved product architecture, other studies further developed and empirically verified the usefulness of the method (Kurtadikar et al., 2004; Day, Stone and Lough, 2010). The heuristic method for identifying modules has been later used by Chandrasekaran et al. (2004) to develop design templates for product-platform-focused design in multi-product development. With this method, heuristically identified modules are classified based on their re-occurrence in a set of considered products with the help of a module-product matrix. Another example for an extension of the heuristic rules can be found in the work of Zamirowski and Otto (1999) who add two portfolio related modularity rules to Stone's (1997) heuristic rules.

3.1.2 Functional-physical analysis

Addressing physical-functional relations when optimising product architectures has the aim of making part of the product functional-independent. This means that the relation

6 The Oxford Online Dictionary defines 'heuristic' as "enabling a person to discover or learn something for themselves" (Oxford Dictionaries Online, 2011) whereas Stone et al. (2000, p. 14) define module heuristics as "A method of examination in which the designer uses a set of steps, empirical in nature, yet proven scientifically valid, to identify modules in a design problem."

between the functional domain and the physical design domain is mapped and optimised based on modularity principles. Functional independence of modules follows the definition of modularity which was shown earlier in this work and the principles of axiomatic design (Suh 2001).

Without developing a designated method, Stake (2000, p. 100–102) uses the direct relation of product properties to physical solutions. “Product properties” can be seen as an intermediate state between design properties and functional requirements and “physical solutions” are considered as components. Jiao and Tseng (1999) introduce a huge methodology for product family architecture (PFA) development. Their holistic methodology generates functional, technical and physical views of the product architecture. For the actual architecture design process, a matrix-based clustering algorithm (Kusiak and Chow, 1987; Newcomb, Bras and Rosen, 1996; Tseng and Jiao, 1997) is used to cluster functional-technical relationships into functional independent modules. The importance for a preceding thorough analysis of customer needs and functional requirements as base for the PFA is stressed and focused by the research group of Du et al. (2005, 2001).

In a similar approach, Goepfert (1998) develops a methodology which establishes the product architecture based on the visualisation of the mapping between product functions and product components with the purpose to achieve functional independent modules. In addition, the visualisation of the product architecture helps to align the structure of the modules to the organisational concept or vice versa (Goepfert and Steinbrecher, 2000).

3.1.3 Physical interactions between system elements

Components of a product can be grouped into modules in a way that interactions predominantly occur inside modules and that there are only few defined interactions between modules. Only few interactions between modules allows for efficient standardisation of interfaces and therefore for module combinability. Moreover, if there are only few standardised interfaces between modules, it is less likely that a change to one module has an impact on other modules or to other parts of the product.

As the first research stream within this category, graph-based methods identify modules based on interactions between components or subassemblies of technical systems. These methods are applied to visually or algorithmically establish product architectures. The graphs show the components and the dependencies between them. Afterwards, the strength of dependency is rated so that components with strong dependencies can be grouped into modules (Kusiak and Huang (1996); Shamsuzzoha (2011)).

Another research stream establishes product architectures with the help of matrices. Steward (1981) introduces a design structure matrix (DSM) to manage the design of complex systems. Pimmler and Eppinger (1994) use the DSM to define product architectures by analysing physical interactions between components of a decomposed product. The ideal product architecture designed with the help of the method contains a high degree of intra-module interactions and a low degree of inter-module interactions (see Figure 24).

An algorithm is applied to support the method (Kusiak and Chow, 1987; Pimmler, 1994; Liu, 2008, 2006, Yu, Yassine and Goldberg, 2003, 2007). In a similar approach, Huang and Kusiak (1998) apply two matrices to optimize product architectures. The first interaction matrix shows physical interdependencies between the components of a product. The second suitability matrix is a component-component matrix as well and shows whether it is desirable to integrate two components into the same module or not. Even though DSM methods have been constantly adapted and applied (Eppinger and Browning, 2012), it is interesting to note that a number of these references are 20 years old, yet organisations are still struggling with very mixed portfolios of products and all the attendant inefficiencies.

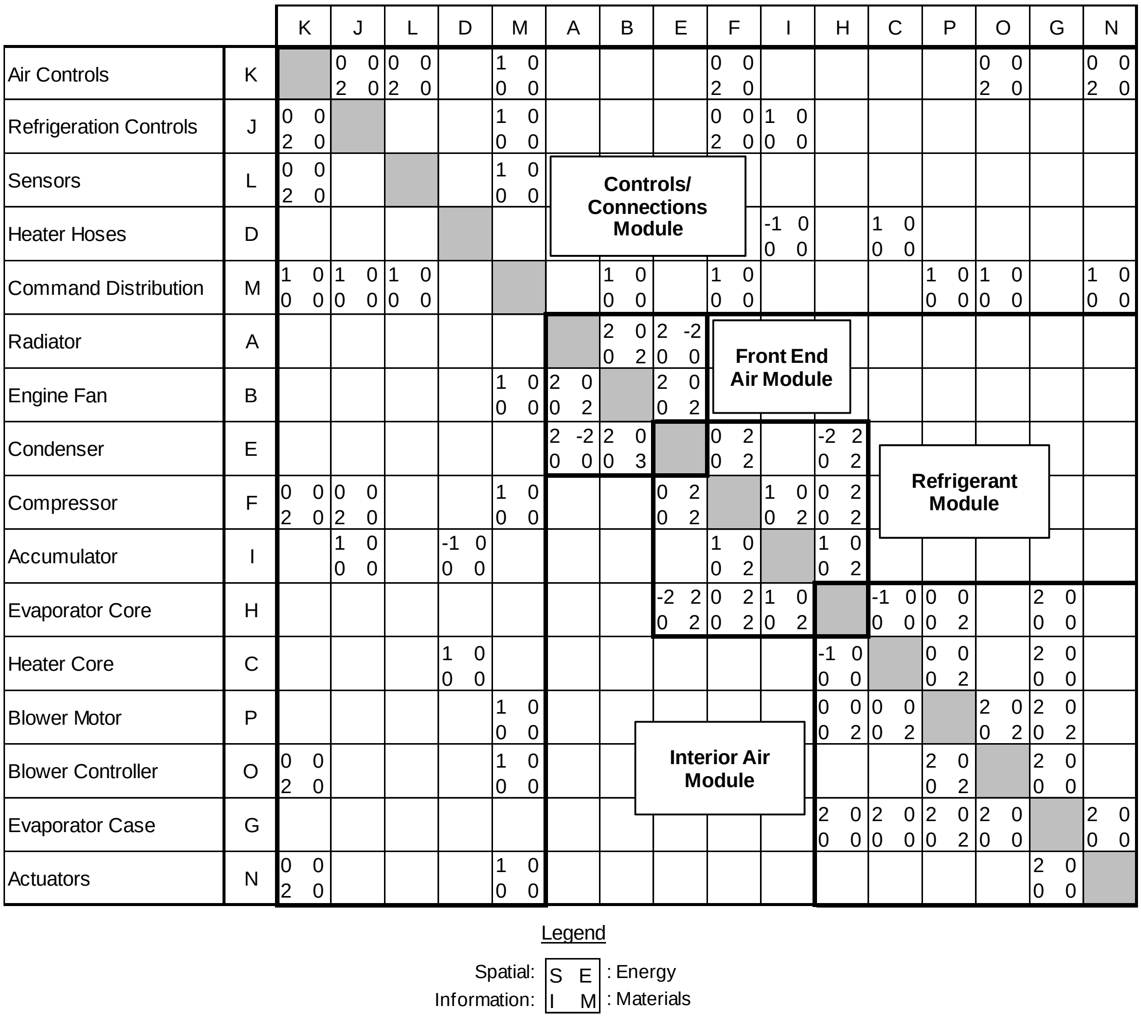


Figure 24: Design Structure Matrix of a climate control system (Pimmler and Eppinger, 1994)

Helmer et al. (2008, p. 648) argue that the quality of the resulting product architecture of DSM techniques is unsatisfactory. They claim that the information type required for product architecture definition is not clearly defined, unavailable or of poor quality to come to the right results. Moreover, the researchers claim that just focusing on the minimisation of inter-module interfaces and the maximisation of intra-module interfaces does not always lead to improved product architectures. Therefore they develop an extended DSM approach which covers other relevant aspects like diligent information collection, consideration of other domains for the DSM, post algorithm phase correction and improving the results, also with regard to constrains and technical feasibility. These additional factors have to be determined case-dependent and highly depend on individual engineering knowledge. Other works integrate other perspectives such as functional cost aspects into product architecture creation with the help of interaction matrices (Shan and Chen, 2009; Xu et al., 2006; Alizon, Shooter and Thevenot, 2006).

Designing the product architecture based on interaction between system elements helps to identify detrimental and desired interfaces as well as a product structure with maximised intra-module and minimised inter-module interactions. The minimised inter-module interactions can be realised with well defined interfaces. The analysis of interactions and the focus on interfaces between elements allows for efficient standardisation of module interfaces. Thus, the resulting product architecture is less sensible to change because a change to a certain module is less likely to affect another module and high combinability is provided due to standardised interfaces.

3.1.4 Addressing strategic reasons

Not all researchers regard the factors mentioned above in previous sections as appropriate input factors to modularise products. For instance, it is claimed that functional elements have another purpose (e.g. starting point for a technical system) rather than serving as an input factor to set up a modular system. Thus, it is suggested that "the functional structure does not directly show the modular structure in the design of new products that include a physical assembly" (Lehtonen, 2007, p. 67) Moreover, Lehtonen (2007, p. 96) claims that "the key issues for the division of the module structure arise from the business environment and the production environment, and the relations emerging from the technical implementation ought to be examined only in regard to these product requirements". Therefore, this section now turns to the examination of methods that establish architectures driven by strategic business factors.

- Product life cycle considerations

Many researchers establish product architectures with factors that address various life cycle issues. In all presented research works of this section, it is claimed that the product architecture is a great lever to better achieve the goals of life cycle engineering. This may include the following perspectives: engineering, manufacturing, testing, assembly, distribution, operation, service, maintenance, reuse, recycling and disposal.

Based on these claims, researchers have developed methods to enhance the life cycle performance of products. To achieve this, components are grouped into modules if they have similar or the same life cycle characteristics like the same material for recycling, the same reuse intend, or the same maintenance interval (Gu and Sosale, 1999; Newcomb, Bras and

Rosen, 1996). Gu et al. (1997) develop a method which establishes modular product architectures based on the same principles. However, they also add constraints like the maximum number of modules and refine the identified modules based on “manual” engineering knowledge.

Yu et al. (2011) cluster modules based on “modular driving forces”. In their view, modular driving forces are life cycle motivations to group components into modules. Besides functional and structural considerations, components are grouped based on three main motivations: a) similarity in component lifetime (e.g. for maintenance, upgrade or end-of life scenarios), b) similarity in material compatibility (e.g. for reuse, recycling and disposal) and c) ease of disassembly for recycling and other end-of-life intents. In a similar manner, Ji et al. (2013) use ten input factors for green modular design. The factors can be divided into three different categories: functional similarity (e.g. functional compatibility), structural similarity (e.g. component connection pattern) and material reuse similarity (e.g. material environment impact).

The optimisation of the product architecture based on life cycle viewpoints is limited by the constraints of already existing components. To overcome this shortcoming, Coulter et al. (1998) develop a method for the identification and elimination of limiting factors for life cycle modularity. Based on the matrices and metrics of Newcomb et al. (1996), possible changes to the design are identified and evaluated. For instance, materials of different components in the same module can be changed so that they are compatible for recycling. Therefore, components can be efficiently redesigned to improve life cycle modularity (Coulter et al., 1998).

- Module drivers: reasons for grouping parts into modules

Erixon (1996, 1998) develops the five-step methodology “Modular Function Deployment” (MFD) to derive modular product architectures. Module clustering is based on “module drivers” which are the company- and product-strategic reasons why parts should be grouped into modules. To support this step, a module indication matrix (MIM, see Figure 25) is applied to relate module drivers to technical solutions. The methodology is verified in practice and further developed without changing the core of the holistic method (Erixon et al. 1996; Nilsson & Erixon 1998; Nilsson 2010; Ericsson & Erixon 1999).

Module drivers	Technical solutions (TS)	TS 1	TS2	TS 3	TS 4	TS 5	TS 6
Development	Carry-over	3					3
	Tecnology push		9				
	Planned product change			3	9	9	
Product variants	Technical specification	9	3				
	Styling						9
Production	Common unit			9			
	Process/organisation				9	3	
Quality	Separate testing						
Purchasing	Supplier available	3		9			
After sales	Service/maintenance			3			3
	Upgrading				3		1
	Recycling	1					
	Sum of drivers	16	12	24	21	12	16
	Module candidates						

Figure 25: Module Indication Matrix (MIM) indicating potential modules based on module drivers (Blackenfelt, 2001, p. 59)

MFD establishes product architectures mainly based on module drivers during the product life cycle. The module drivers reflecting the company functions R&D, product management, production, quality, purchasing and after sales are as follows (Ericsson and Erixon, 1999):

- Carryover: Components should be grouped because they are together potential candidates for carryover from one product to the other. These modules are tried to kept stable over the whole life cycle of the modular system.
- Technology Evolution: Components should be grouped because they might be replaced at the same time by other or more advanced technologies.
- Planned Product Changes: Components should be grouped because they undergo a planned change (e.g. for customer, development, cost or production reasons) at the same time.
- Different Specification: Components that contribute to product variability should be grouped into as few as possible modules.
- Styling: Components that are strongly influenced by fast changing trends should be grouped into the same module.
- Common Unit: Components that can be used in a large number or in all products in parallel should be grouped into the same module.
- Organisation / Processes: Components that use the same production processes should be grouped into the same module.

- Separate Testing: Components that can be tested together separately should be grouped into the same module.
- Availability from Supplier: Clusters of components that are available from suppliers as final modules should be considered for supply from external manufacturers.
- Service and Maintenance: Components that have similar service and maintenance characteristics should be grouped into the same module.
- Upgrading: Components that are together potential candidates for upgrades should be defined as a separate module.
- Recycling: Components that can be recycled together should be grouped into the same module.

However, much more other factors influencing the product architecture are considered by the method. Firstly, there is the strong link to current and future customer demands and technological trends. Secondly, there is the strong link to feasibility of the modular system in development and production. Thirdly, there are the impact of the platform on various costs (system, production, development) and quality factors (e.g. quality, lead time, sales and after sales). Moreover, the optimum number of modules is defined by correlating it to the assembly time and indicated with the square root of the number of components (Erixon, 1998).

Stake (2000) and Blackenfelt & Stake (1999) extend the research about MFD. Stake (2000) develops a software tool to automatically generate product architectures by applying dendrograms and clustering analyses based on a product management map by Nilsson and Erixon (1998). Another aspect of Stake's (2000) research classifies the module drivers of Erixon (1998) and relates them to the different strategy alternatives (i.e. differentiation, cost leadership, focus on niches) according to Porter (1992) which a company can pursue.

In later works, Borjesson (2009) extends the MFD methodology by incorporating various other aspects while leaving the module driver concept as core of the methodology. Borjesson (2009) claims that module drivers alone do not always lead to better modules. Instead, modules have to be improved by manual engineering work in the classical MFD methodology. Therefore he introduces "convergence properties" which are four additional factors that are used during module generation (Borjesson, 2009):

- Optional product properties
- Geometrical properties
- Functional flow properties
- Module driver compatibility

Figure 26 shows a matrix-based "Product Management Map" (PMM) that is adopted from Quality Function Deployment methodology for modularisation purposes. The PMM guides the MFD methodology from customer requirements over product properties and technical solutions toward clustering of modules with the MIM. Figure 26 also shows the extended elements of Borjesson (2009).

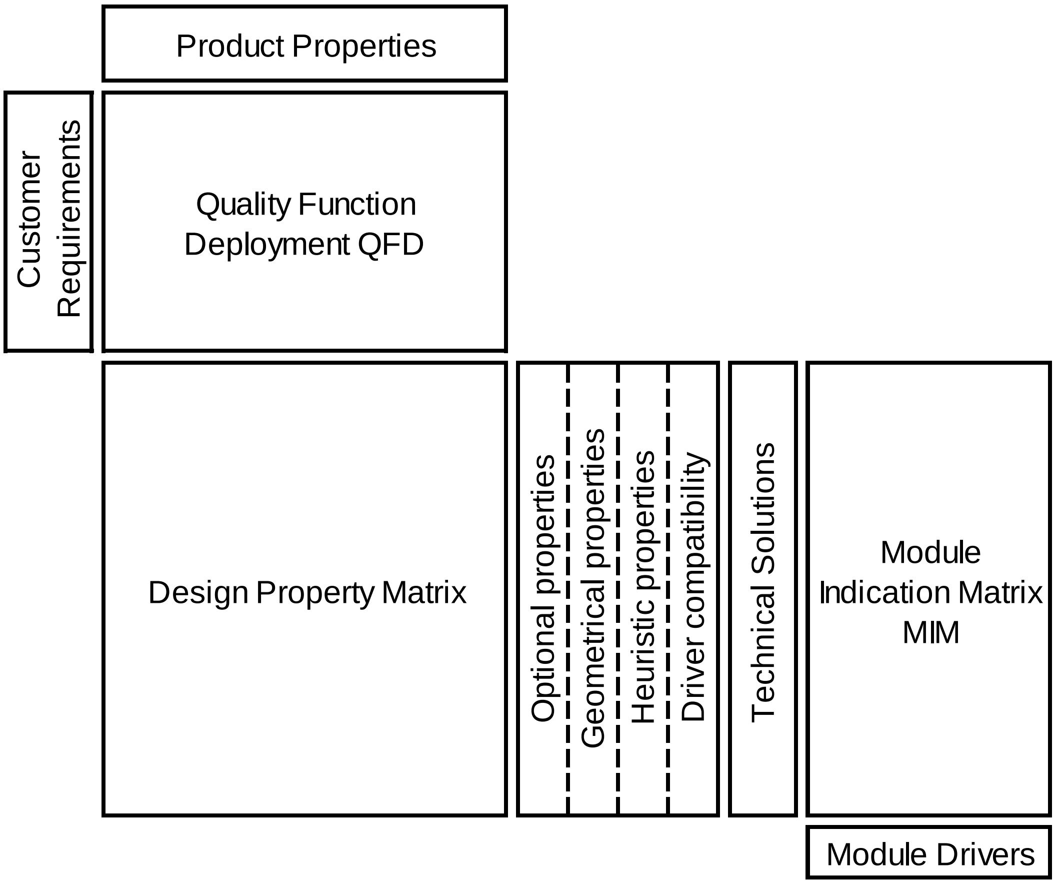


Figure 26: “Product Management Map” to support the MFD methodology (Borjesson, 2009)

3.1.5 Combined and integrated approaches to establish product architectures

There are a large number of studies in literature that combine various above mentioned approaches and add new elements for product architecture creation. Moreover, combined methodologies have in common that they integrate various factors from different stages of the product life cycle or value chain to improve product architectures. This section shows a brief overview of the factors that are considered in combination or in addition to the factors of above mentioned methods. This section will not go into detailed description how these methods work.

- Combining functional and physical elements

Meehan et al. (2007) support design for reuse with modular design. Therefore, they introduce a multi-viewpoint modularisation method. The method starts with module creation on functional level. Subsequently, during the design process, modules are also created on working-principle level and on solution level. Mapping between modules on functional, working-principle and solution level shall support designers to introduce modular solutions that foster design for reuse. Figure 27 shows that the method assists the designer in module clustering from an abstract toward a more concrete level during the design process.

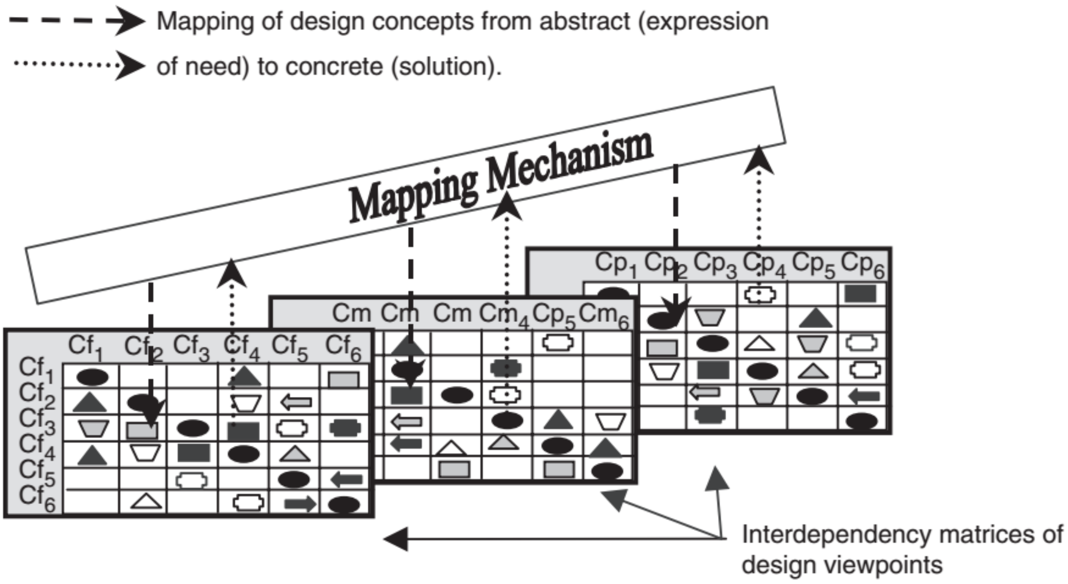


Figure 27: Product architecting on functional, working-principle and solution level during the design process (Meehan, Duffy and Whitfield, 2007, p. 150)

- Combining physical and strategic factors

Early approaches combine technical aspects of the DSM with strategic aspects of the module driver concept (Lange, 1998; Lanner and Malmqvist, 1996). Others combine technical DSM aspects with internally conflicting strategic factors (e.g. stability vs. instability during life cycle of modular system, intended reuse vs. development, commonality vs. variety, carry over vs. change and make vs. buy) (Blackenfelt, 2001, 2000). Strategic reasons from different company perspectives are combined with a functional-physical module interface graph to establish product architectures (Blees, Henry and Krause, 2009, 2008; Blees and Krause, 2008).

- Incorporating DFA and DFM methodologies

The methodology of Salhieh and Kamrani (1999) analyses the similarity of components based on front-end information which is derived from market needs. Physical similarity (e.g. design specifications) between components is considered for module grouping as well as the feasibility of the created product architecture. The new element here is that in later studies, Kamrani and Salhieh (2002) extend the modular design approach by incorporating DFA and DFM methodologies into later stages of their modular design process.

Emmatty and Sarmah (2012) combine a modularisation methodology that accompanies the whole design process with a design knowledge database, a platform-component approach and Design for Manufacturing and Assembly (DFMA). The methodology incorporates constant feedback cycles between DFMA, conceptual product architecture phase and the requirements phase.

- Managing the trade-off between common and variety platform elements

Improved product architectures are achieved by managing the trade-off between using common elements for platform-based design and individually designed elements. After defining all product variants of the product family which share the same product architecture, the product variants derived from the platform are evaluated against individually designed product variants (Gonzalez-Zugasti, Otto and Baker, 2000). More detailed descriptions measure the impact of different architecture alternatives on the trade-off between variant-driven complexity cost and direct cost (Schuh and Jonas, 1997; Schuh and Schwenk, 2001). The trade-off between standardised modules and optimized single products is also solved with the use of algorithms and mathematical models which reflect the performance of different module scenarios by considering constraints of the modular architecture, various types of cost and the price potential of different scenarios (Fujita, 2002; Fujita, Sakaguchi and Akagi, 1999; Fujita and Yoshida, 2004). Yigit et al. (2002) consider the trade-off between quality and performance loss and increased configurability in production from modular architectures.

The right platform strategy for different market needs based on deep market understanding, price-performance targets of platform products, and the platform roadmap is considered by Meyer and Lehnerd (1997). Simpson et al. (2001) makes the technical link between market segments, platform specification, and the division of the platform into scalable parts.

De Weck et al. (2004) came up with an approach that goes beyond single platform improvement. The approach determines the right number of platforms to cover different market segments. This approach is based on a maximised profit function for a product family that is based on market data, competitor data and on estimated variant and capital investment (e.g. factories, dies and R&D) cost.

- Considering effects on production and logistic processes

Architecture-relevant relationships between the entities of the functional product hierarchy (e.g. welded joint, bolted connection, adhesive bonded joint or contact surface) are improved by applying optimisation measures. The optimisation measures can be grouped into three fields which are used as levers: rearrangement or reduction of assembly steps, definition of subassemblies and reduction of assembly parts through integration (Bäßler, 1987; Dahl, 1990).

The optimal fit between product architecture, assembly sequence and product variants can also be achieved by visualizing product variance in combination with the process sequence (Schuh, 1988). Another option to achieve the best fit between the product architecture and the process sequence is by measuring the impact of variant-caused complexity on logistics and manufacturing processes which is directly related to the shape of a product-process variant tree (Caesar, 1991).

- Considering uncertainties

Several researchers mention that despite acting amid highly volatile markets, product architectures are still created with static planning. Therefore, they design scenario-robust product architectures by varying uncertain, imprecise and dynamic input- and output factors for the product architecture creation process and analyse how the product architecture looks like under changing circumstances (Moon et al., 2007; Nepal et al., 2008; Schuh et al., 2014; Schuh, Lenders, and Bender, 2009).

Based on the work of Jordan and Graves (1995) about manufacturing flexibility and product-plant-configuration and the work of Connors (1996) about multi-attribute trade off analysis, Kidd (1998) develops a method for the review of different platform strategies. Therefore, he connects platform attributes and platform uncertainties of different platform scenarios to economic platform performance metrics (Kidd, 1998). Future platforms with stable product architecture need good anticipation of future trends, risks and market preferences during the early product creation phase. This need is addressed with the proposed simulations in the study.

- Integrating PLC or strategic, functional and physical aspects

Recent research streams continue to integrate more and more factors for modularisation by integrating factors such as customer requirements, product programme planning, functional requirements, technical dependencies and various strategic requirements (Jonas, Gebhardt and Krause, 2012; Koppenhagen, 2004; Sand, Gu and Watson, 2002, 2001; Ulrich and Eppinger, 2012).

Koeppen (2008) remarks that the strategic, PLC, functional and technical aspects which are considered for product architecture creation all rely on qualitative engineering judgement which is to some extend subjective, not repeatable and know-how dependent. For this reason, she develops formulas to describe and quantify the reasons why components should be coupled into modules. The mathematical formulation of modularization reasons, covers strategic, PLC, functional and technical aspects.

Simpson et al. (2012) develop an integrated approach for product family design. The method is based on proper analysis of product requirements. Afterwards, different support tools (e.g. metrics, matrices, graphs) are applied to balance the trade-off between product planning, variability planning and commonality planning. Other highly integrated and development process-accompanying approaches have also been presented by Thumm & Göhlich (2015) and Pakkanen et al. (2015).

Marshall and Leaney (2002) present a framework for systems engineering which includes a methodology for product modularisation. The framework brings modularisation into the broader range of system engineering which respects the requirement for a holistic view on the topic of modularisation (see Figure 28).

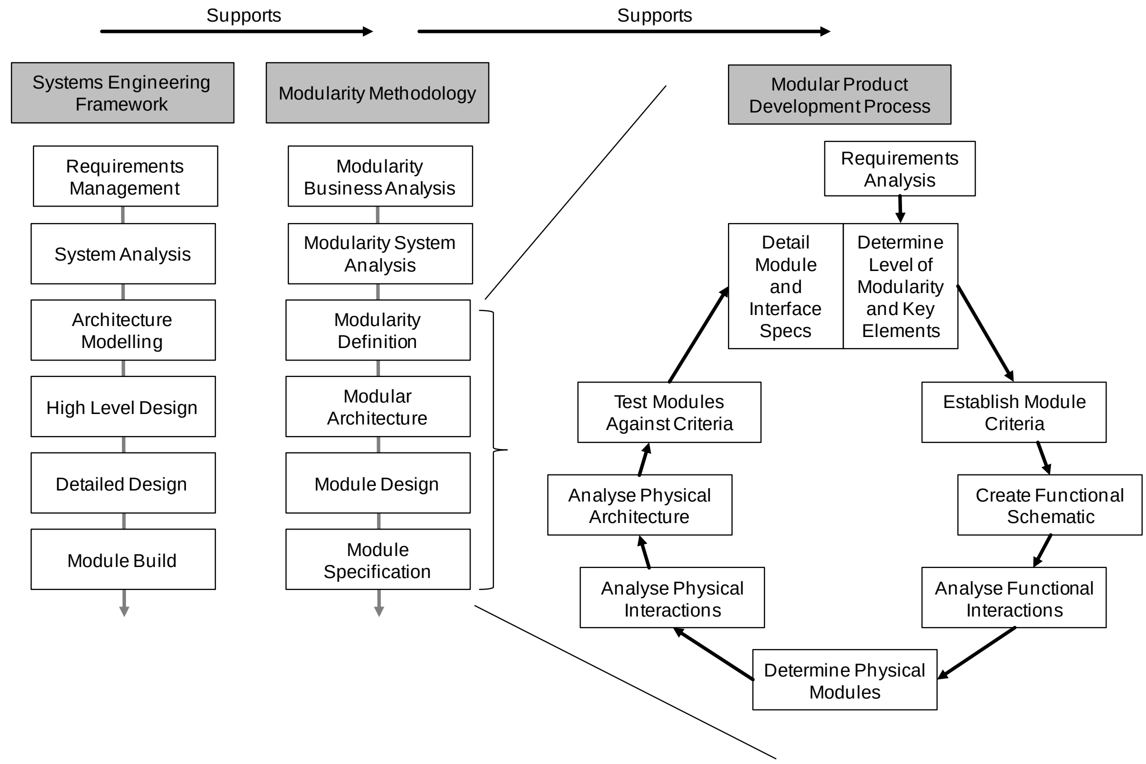


Figure 28: Modular product development process in the context of the system engineering framework (Marshall and Leaney, 2002, p. 296–297)

Applying various tools along the product development process to enhance modularity, commonality and variety has become increasingly attractive. Some researchers combine elements of Design for Variety (DfV[7], e.g. based on Martin & Ishii (2002) and Martin (1999)) with strategical module clustering and other accompanying information during the design process (Blees, Kipp and Krause, 2010; Kipp, Blees and Krause, 2010; Kipp and Krause, 2008). Krause et al. (2014) and Kruse et al. (2015) further develop their work and introduce an integrated method toolkit for modular product families that can be applied during different phases of product development. The toolkit consists of eight visual tools that aim at following aspects:

- Design for optimised variety of modules and products
- Modularity for different product life cycle phases
- Product program planning as input for modularisation
- Development of modular product programs

Schuh et al. (2007) draw upon a QFD-based matrix-approach to handle their modularisation method. The methodology starts with requirements from various sources and relates them to technical functions. Afterwards, different product architecture alternatives are established based on the relations of technical functions to physical components. Finally,

[7] DfV, Definition by Kipp and Krause (2008, p. 426): "...possibilities of design and product architecture, which minimise the costs of the development and production of variant products."

physical components or modules of a certain product architecture alternative are related to each other in order to define interfaces between them on a conceptual level. Figure 29 shows the matrices of the holistic modularisation methodology.

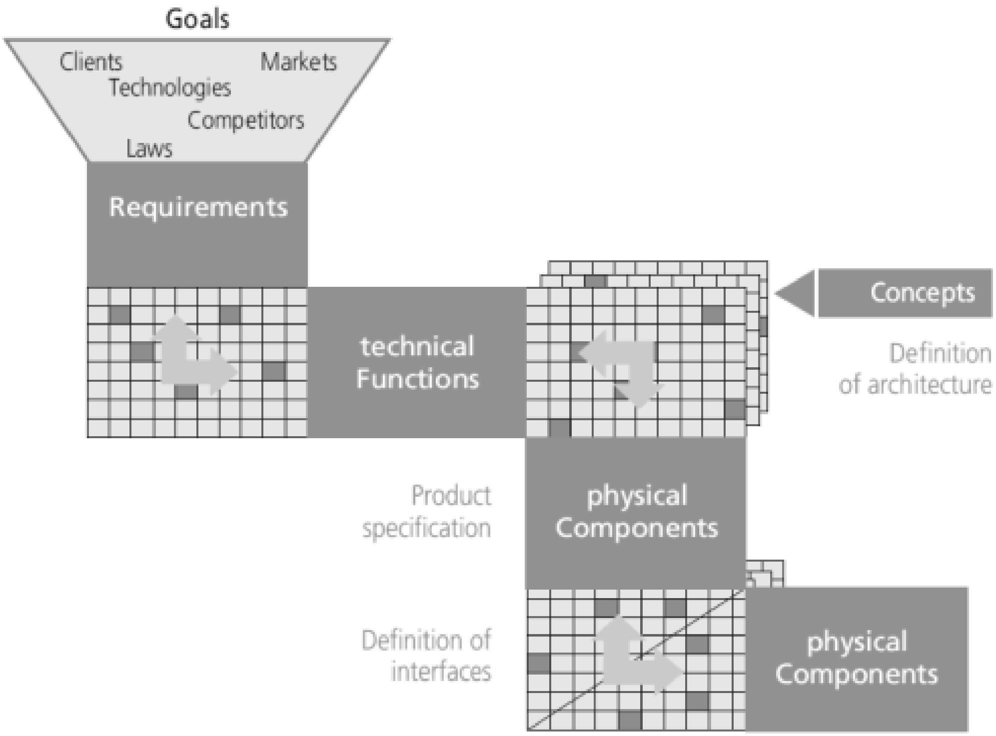


Figure 29: Matrix-based approach to handle the modularisation method of Schuh et al. (2007), figure according to Schuh et al. (2007, p. 29)

Based on a study on the definition of product platforms (Kristjansson et al. 2004) and on an analysis of influencing factors which should be considered during the creation of a platform strategy (Kristjansson & Hildre 2004a), a method to review product platforms is developed (Kristjansson & Hildre 2004b; Kristjansson & Hildre 2004c; Kristjansson 2005). The developed method assesses the alignment of platforms to the company's strategy (i.e. differentiation, cost leadership, focus). The aspects that are assessed comprise a) platform strategy, b) internal side effects, c) external side effects, d) match of platform with products, e) dynamics of the market, f) competition situation and g) the competency of the company regarding the platform. The results of platform assessment evolve during discussions that run in parallel to the platform development process. The assessment of the platform remains on a high and managerial level. Because the ratings are based on experience, estimations and gut feeling, the results do not reflect a real state but can only be seen as a current approximation to the platform's performance. The method depends on the opinion of different individuals. This makes it possible that different individuals rate the performance of a platform totally different in the same field due to missing facts that should ideally be used to back the reasoning for a certain rating.

3.1.6 Overview: methods

The literature review about modularisation methods shows how engineers are supported to establish product architectures during the engineering design process. The previous sections have presented 58 methods from 89 research publications that are eligible candidates to support engineers in establishing modular systems, with a variety of key points. It is the purpose of this overview section to give a compressed taxonomy of how work from the literature supports engineers during modularisation activities.

During *module clustering*, most of researchers draw upon graph-based, matrix-based or mathematical support. For instance, graph-based approaches help to visually subdivide a set of functions into modules, matrix-based approaches are used to identify similarity patterns between components and mathematical models help to balance the trade-off between different estimated effects of modularisation. There are also numerous researchers who apply a mix of different means for product architecture representations or who apply metrics or tables. Figure 30 shows the distribution how the identified modularisation methods support engineers during product architecture design.

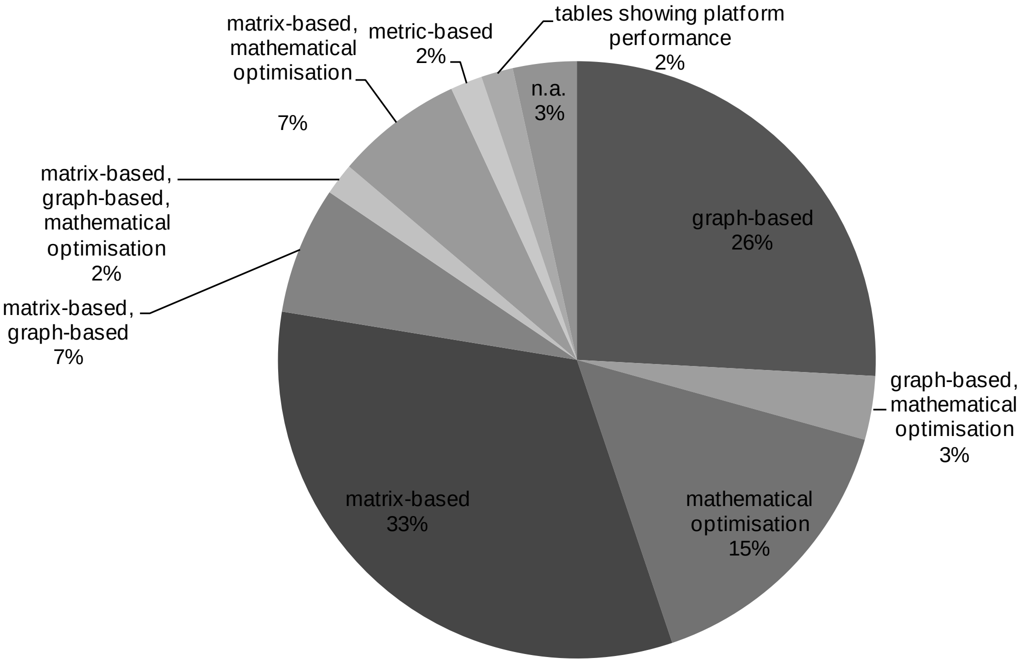


Figure 30: Distribution of how the presented 58 methods support engineers through product architecture representation

The methods have in common that they aim at organising a system into modules. This can either be achieved bottom-up or top-down (Alizon et al., 2007; Liu, Wong and Lee, 2010; Simpson et al., 2014). In detail, this means that they either help to group a set of elements (e.g. functional, physical) into modules **(bottom-up approach)** or that they subdivide a product or product portfolio into modules with the help of different input factors **(top-down approach)**.

The literature review identifies factors that can be used to establish modular systems. The input factors can be seen as reasons and guiding principles for product architecture modification. Therefore, a modular product architecture is claimed to contribute significantly to the performance of different company areas.

What was found in literature can either be divided into methods considering *abstract factors, strategic factors, or integrated approaches* which integrate a mixture of various factors (see Figure 31). The derivation of data for Figure 31 is given in Appendix A.

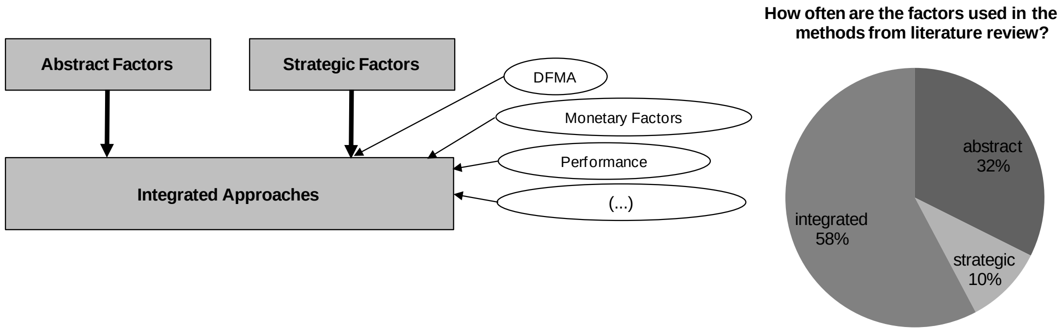


Figure 31: Factors that are applied to establish modular product architectures

Abstract factors are physical interactions between elements, functional structures and functional-physical relations. They are used to create an improved product architecture based on the abstract definition of "good design" or of "modular product architecture" which was defined earlier in this work.

Other methods improve the product architecture based on more practice-related, less abstract factors. Among these reasons are strategic reasons for modularity which contain factors from the whole product life cycle and value stream. Strategic factors can be applied generally with the help of module drivers as well as with the help of more detailed product life cycle factors.

Integrated (holistic) approaches consider various factors and are therefore further classified. Integrated platform-based methods consider the trade-off between commonality and individuality. This trade-off can be managed by directly comparing different platform architecture alternatives against cost, revenue, performance, quality and other characteristics of the product specification.

Holistic consideration of production and logistic processes takes into account if various measures affecting the product architecture are beneficial or disadvantageous for logistics and manufacturing. Other holistic methods combine the principles of variability management and modularisation. These approaches strive for on optimum balance between internal complexity, external variety and the cause-effect relationship between them. Further holistic methods combine a huge amount of strategic, functional and physical factors.

Several holistic methods were found which have distinct features in addition to more “conventional” methods. Among them are integrated methods considering the optimum degree of modularity. On the base of other known main input factors, different architecture types are created and evaluated or the level of modularity is planned prior to applying the methodology. Another distinct feature is the consideration of uncertainties which makes clear that the expectation for factors which are used during early product creation phase could rely on an unstable base during the life cycle of the product architecture.

Besides the main input factors which are used for the actual product architecture generation process, researchers use side input factors which are used prior to or after the actual product architecture creation process. Side factors with origin before the actual product architecture creation process usually consider market data or functional issues. Moreover, some researchers stress the importance of alignment of general business objectives with product architecture improvement objectives prior to undertaking the architecture creation process.

Side input factors which are considered after the actual architecture creation process come from various fields. Usually constraints, feasibility and impact of the product architecture are considered after the actual architecture creation. For instance, findings after DFMA methodology, production ramp-up or testing is used to create an iterative feedback loop to the conceptual product architecture creation phase.

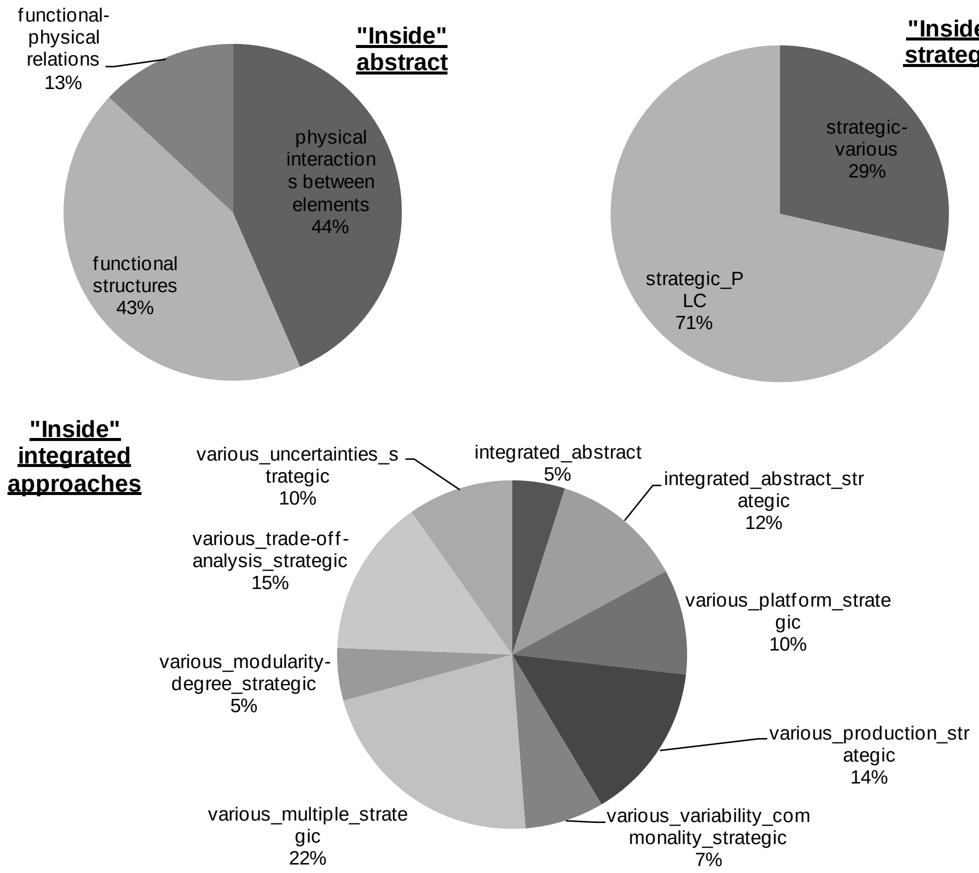


Figure 32: Percentage of factors used by “abstract” methods, “strategic” methods and “integrated” approaches

Figure 32 gives a detailed overview at the percentage of different factors that are applied “inside” abstract, strategic or integrated approaches. The derivation of data and more detailed overview are given in Appendix A: Overview of modularisation methods.

Recently developed product architecture design methods (see Section 3.1.5) tend to integrate more and more factors. Researchers also put considerable effort behind developing sophisticated algorithms for automated architecture design. This requires immense information input. In addition, it is also stated that the outputs of the methods can only be achieved with the input of manual engineering knowledge which cannot be captured with any methodology.

Moreover, methods alone are not capable to deliver optimised product architectures. It was found out that if applying different methods to the same problem delivers considerably different product architectures (Holtta and Salonen, 2003). This result is unsatisfactory and can be explained with the complex relationship between applied factors and the enormous amount of vague estimations and predictions that have to be undertaken during product architecture design.

Nevertheless, the validation of the methods in literature shows that the application of modularisation methods promises good outcomes. However, these outcomes are not for free. Methods have to be implemented and applied with a certain effort, especially if the method is new to its users. Another reason for methodological scepticism can be the delay between applying the method with certain amount of effort and delivering results such as cost savings (Lindemann 2006, p.59). No study has been identified that gives evidence to a case where a modularisation method has been applied in daily industrial practice. Most of the modularisation methods have been validated on a small sample size, relatively isolated from industrial practice. However, reality in industry is not that neat. Rather, there are diverse and unforeseen variabilities in practice. Consequently, the transfer of modularisation support from academia into daily industrial practice with enormous time and cost pressure could be prone to failure if the underlying overall issue of transition towards modularisation is not properly understood.

A major part of this section has dealt with factors that are used in literature to establish modular systems. In order to make a first step towards understanding the overall issue of transitioning towards modularisation, the next section will move on to bring the presented modularisation methods and their applied factors into the wider range of a company's development process life cycles.

3.2 Context of modularisation approaches in the product life cycle

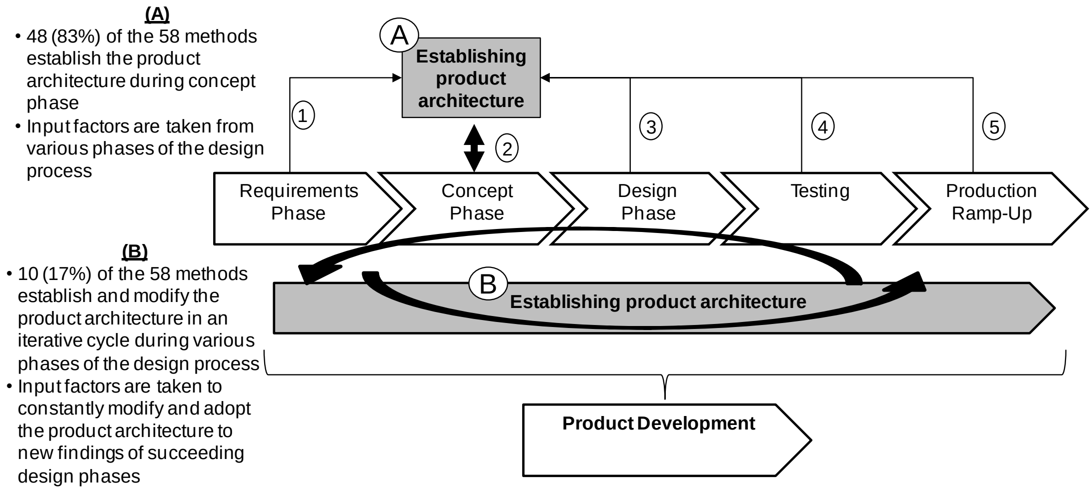
It is the purpose of this section to bring modularisation into the context of the development life cycle. It will be shown where modular systems are suggested to be established in the product life cycle by the presented modularisation methods. Furthermore, it will be shown which aspects of the product architecture life cycle are covered by the literature review, and equally important, which aspects are not covered by the 58 modularisation methods that were identified. These are analysed in some detail.

The detailed taxonomy of modularisation methods is given in Appendix A: Overview of modularisation methods. There, it is also described how the data and graphs of this section were derived. The following paragraphs will summarise and visualise the results of bringing modularisation methods into the wider context of the product life cycle.

3.2.1 Context in the product development process

Figure 33 has been created to show two different alternatives as to how product architectures are established during the development process. The figure has been created based on the information provided in Appendix A.

The first alternative, alternative (A), establishes product architectures during the concept phase of product development. This alternative is chosen by the majority (approximately 83 %) of modularisation methods of the literature review. Input factors that are used to establish the product architecture are either taken directly from concept phase or from various other phases of the engineering design process.



Id phase	Name phase	Times considered in 58 methods	Percentage
1	Requirements Phase	22	38 %
2	Concept Phase	52	90 %
3	Design Phase	27	47 %
4	Testing	12	21 %
5	Production Ramp-up	21	36 %

Figure 33: Consideration of product development phases during modularisation

The second alternative, alternative (B), is chosen in around 17 % of the 58 identified modularisation methods. These methods start to establish a preliminary product architecture quite early in the development process and establish a constant feedback loop between different design phases in order to constantly modify product architecture through newly acquired findings of succeeding phases.

Both of the presented alternatives clearly have their own advantages and disadvantages (Kusiak, 1995, p. 261). For instance, if the product architecture is early fixed during concept phase, constraints that arise later during other phases of the development process will undermine the product architecture. Moreover, establishing the product architecture early requires engineers to make vague estimations on factors that arise during succeeding phases. On the other hand, if the product architecture is constantly modified, designers of (sub-) modules and components encounter serious problems if they cannot rely on fixed functionality or interfaces (i.e. the results of previous phases). Therefore, it has to be stated that the classification of modularisation methods is neither black nor white, nor that clearly cut. In practice, users of the methods have to find a situation-specific balance between the alternatives.

One of the most important points of this section is the coverage of phases during product development. The table in Figure 33 shows that all phases of new product development which are shown in the same figure are sufficiently covered by the identified modularisation methods. This is a result of the finding that researchers have integrated more and more factors to establish architectures (as shown in section 3.1.6), and in parallel, researchers have integrated more and more product development phases during modularisation since recent years. For instance, the phase with the smallest coverage, which is "testing", is still considered by at least 12 modularisation methods. It can be concluded that whichever phase during product development shall be considered or improved by modularisation, there are a significant set of methods available. Consequently, there is actually sufficient appropriate available support in this area. Hence the focus of the next section will turn to the overall life cycle.

3.2.2 Context of modularisation in the overall life cycle

This section goes beyond the product development process and brings the modularisation methods into the context of the overall life cycle phases of a company. Figure 34 shows all phases that are considered by the 58 methods of the literature review.

The methods cover different overall life cycle phases for two reasons. Firstly, factors from different life cycle phases are considered for product architecture improvement. Secondly, the performance of considered life cycle phases shall be improved by modularisation itself.

The depicted phases that could be identified in the literature review cover the aspects of (1) product development, (2) the product life cycle (excluding (1) product development), (3) the value stream, (4) financial functions, (5) organisational aspects, (6) phase-overarching aspects that cannot be directly assigned to a phase and (7) the evolution of the modular system beyond its development (see Figure 34).

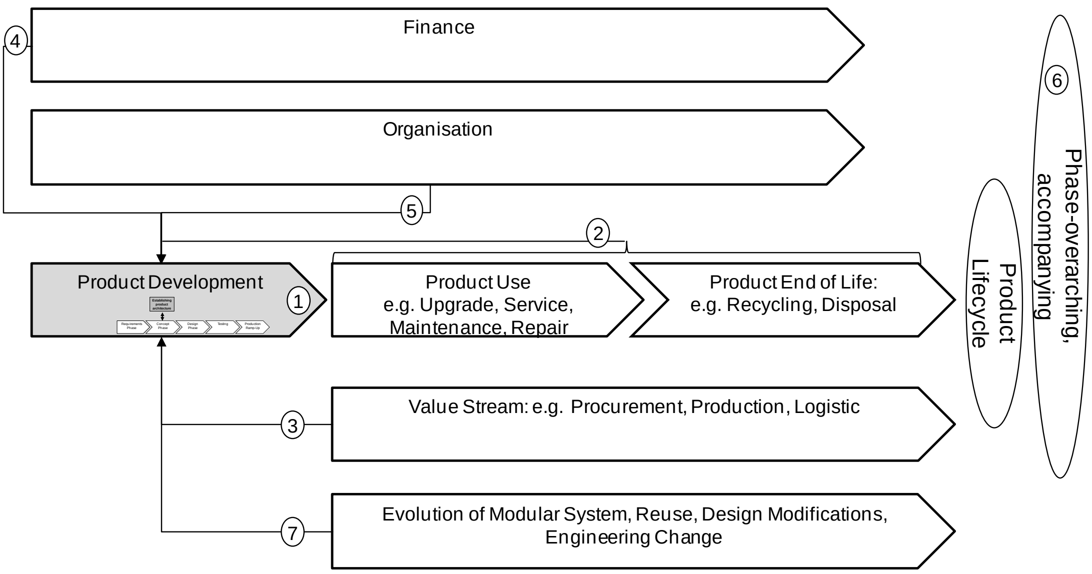


Figure 34: Life cycle phases considered by modularisation methods

Figure 35 shows the coverage of each phase by indicating how often a phase is considered by the 58 methods of the literature review. Most of the modularisation methods consider aspects that have their origin in the product development process, the product life cycle or the value stream. Less frequently, but still significantly covered are the aspects of financial and organisational processes. Phase-overarching or accompanying aspects are considered by 4 % of the methods. This category comprises aspects that could not be directly assigned to one or several of the listed phases.

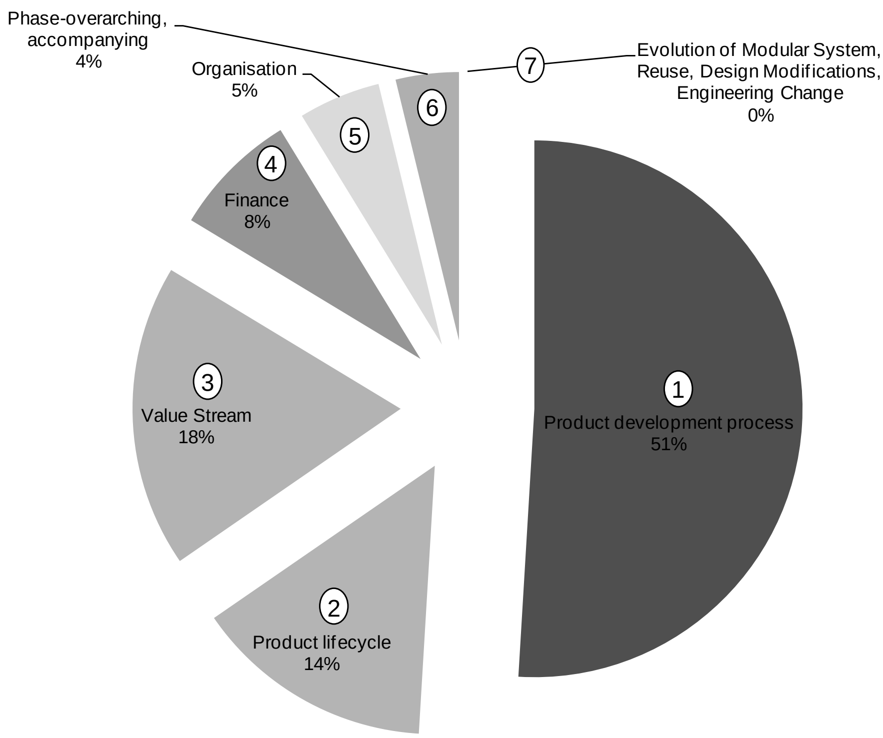


Figure 35: Life cycle phases that are covered by the modularisation methods from literature review

All aspects related to the category of (7) evolution, reuse, design modification and engineering change of the modular system or its elements are not covered by the identified 58 methods. These aspects have been totally neglected by previous scholars who do research in the field of modular product architecture design (see Figure 35).

Several conclusions can be drawn from the finding that phase (7) of Figure 34 and of Figure 35 has been neglected by previous researchers. Firstly, handling the aspects of this phase could be overly simple and self-explaining. Such a situation would not require extensive research about the topic. Secondly, it could be derived that it is totally unimportant and not relevant to consider how a modular system evolves after it is planned and established during a new product development process. Thirdly, it could be concluded from the findings of this section that the focus of current research is misleading. It has to be questioned if the focus on single new product development projects that start with a blank sheet of paper really reflects the needs of industry for modular system development.

A recent study in industry concludes that the third possible conclusion of this chapter is the only valid one (Arnoscht, 2011). Arnoscht (2011) found out that it is not sufficient to plan the product architecture during a single product development process. Rather, many problems arise exactly after this phase, namely, during phase (7) evolution, reuse, modification and change of the overall modular system (see Figure 36). It can be thus concluded that dealing with this phase is neither simple nor unimportant. In contrary, if a company wants to transition from single product development towards development of modular systems, it is vital consider how a modular system can be evolved and how it can be maintained over a prolonged period.

Other previous studies showed that commonality in platforms tends to decline over the platform life cycle. Moreover, actual commonality in implemented platforms and derivative products is significantly lower than the commonality that was originally planned during product family planning (Boas, 2008; Montano, 2011; Munk, 2011). This problem is denoted as “platform divergence” and is seen as a main future issue to be remedied in this field (Simpson et al., 2014).

In summary, if a company wants to make the overall transition towards modular system development, it has to ensure that the product architecture across different products is *kept stable* over time. Such stability can be achieved by taking the evolution of the modular system across a wider range of products together with its reuse, design modifications and change into consideration. This is an under-researched topic which needs further investigation in order to avoid significant problems during overall modularisation transition.

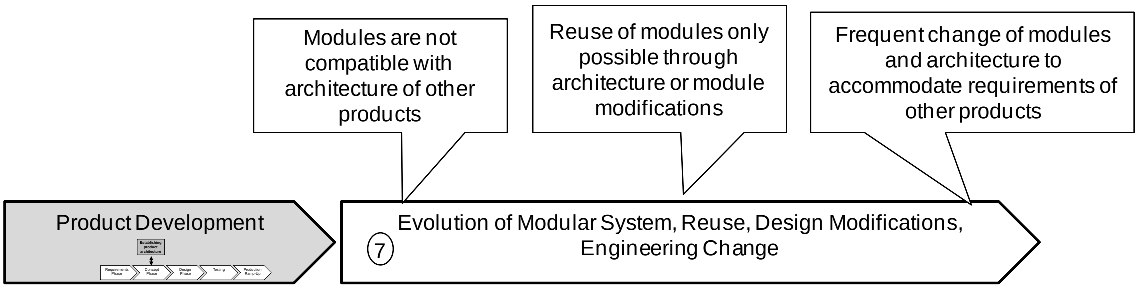


Figure 36: Issues during transition from single product development towards modular system development (Arnoscht, 2011, p. 208)

So far, this research has presented what modular systems are, how they are suggested to be established by academia and what the problems are from a theoretical perspective. The next section will now move on to show the practical significance of modular systems in industry.

3.3 Product architectures in industry

Modularisation is not a new concept, either to academia, or to industry. However, during recent years, the dimension of the number of products that are derived from platforms and modular systems has dramatically increased. Examples for product architecture improvement and modularisation in various industries can be found throughout the literature. Most cases report about success stories with the introduction of modular products or modular systems. However, drawbacks encountered by industry will also be presented in this chapter.

3.3.1 Modular architectures in different industries

In his literature review about modular systems, Arnoscht (2011, p. 25) goes back until 3000 BC and lists Egyptian hieroglyphics as first modular system. According to his review, the first technical modular system can be dated back to 300 BC with modular bricks that possess 1 : 1,5 : 3 relationship in dimension.

Up to modern times, the progress of technology has been accompanied by the progress of modular systems. Lehtonen(2007, p. 25–26) presents how modularisation of submarines was used to improve organisation of production during the 1940s. He states that this concept that was new during those days is still applied in shipbuilding industry today.

In addition, Lehtonen (2007, p. 26) gives an example from railway industry in the 1970s. The example includes diesel locomotives that incorporate improved life cycle characteristics like facilitated service through modularisation.

Rothwell and Gardiner (1983, p. 165) show an example how well defined interfaces between the jet engine module evolved together with the stretchable platform of the Boeing 757 in the 1970s and 1980s.

Baldwin and Clark (2000, p. 6–9) state that the product architecture of computers transformed from being integral to modular, starting with IBM's System/360 in 1964. The evolution of the whole computer industry from integral designs towards modular designs mainly took place in the 1980s and 1990s and is claimed to be one of *the* driving forces that reshaped computer industry.

With the Sony Walkman from the 1980s, Sony managed to be steps ahead of competitors by constantly and quickly introducing more than 250 different models in the US. Core modules and platforms are seen as reason for this success in pace, variety and flexibility (Sanderson and Uzumeri, 1995).

The panel meter of Nippondenso Co. Ltd is an example how a modular product architecture can help automotive suppliers to improve combinability of their components. 288 different panel meters could be efficiently manufactured by combining six components with 17 predefined component variants. In addition, Nippondenso Co. Ltd reduced production variety with the redesign of its products like the 250 varieties of its alternators (Whitney, 1993).

Huge investments in the redesign of a product line concerning automation, cost and value creation paid off for Black and Decker. Foundation for the success was the consequent planning and re-use of standardised components that were already optimised for manufacturing. With this strategy, the complexity driving iteration loop between design to function, design for manufacturability and design to cost for each product was eliminated. Amid high standardisation across products in a family, Black and Decker managed to keep high variety with universal electric motors by just adjusting the stack length (Lehnerd 1987).

Feitzinger and Lee (1997) see in modular product design a precondition for modular production and supply design with standardised processes, postponing and process resequencing in order to get agile production and supply. In the case of modular redesign of generic DeskJet Printers from HP which enabled new supply strategies, "the total manufacturing, shipping, and inventory costs dropped by 25%" (Feitzinger and Lee, 1997, p. 118).

Ericsson and Erixon (1999) report about successful introduction of modular product architectures at Scania, Volvo, Atlas Copco Controls (control units for machine industry), VBG Ltd. (truck and trailer industry), and Sepson (winches for heavy duty vehicles). According to the authors, the major goal with modularisation in these companies is a broader offer to the customer while reducing the part number count, assembly time, lead time in production and engineering, investment, purchasing cost, quality losses and testing time (Ericsson and Erixon, 1999).

Examples for modular systems can be found in almost all industries. Beyond the presented cases so far, further examples reach from modular watches (Ulrich and Eppinger, 2012, p. 189), modular gearboxes (ZF Friedrichshafen AG, 2012), modular conveyors, modular trams (Adolph, 2005; Pahl et al., 2007, p. 510–515), electromechanical control systems (Bathelt et al., 2003), modular software (Kuhlemann, 2006), modular axle units (ZF Corporate Communications, 2010), agricultural machinery (Mayer de Ávila and Borsato, 2014), modular spacecrafts (Gonzalez-Zugasti, Otto and Baker, 2000) to modular smartphones (Kazi, 2015; Khedekar, 2016; Phonebloks, 2016). Even medical industry like hip implant manufacturers use the principles of modular design (Hips For You, 2009; Paul Byrd Law Firm, 2013). Although modularity in this industry is not without risk, the possibilities to generate variety are enormous. Hence, there are more than 35 000 different artificial joints registered in Germany's artificial joint registry (Deutsches Ärzteblatt, 2013)[8].

Most examples for product architecture strategies are reported on product family level. However, other promising examples such as the Black and Decker radical redesign program shows that such a program may go beyond product family level. Scania or Volkswagen (see section below) show that transitioning towards modular system development is implemented and aligned throughout the whole company and across different brands. In the case of Renault-Nissan, General Motors together with PSA (i.e. Peugeot and Citroen) or Daimler with Nissan (Mitteldeutsche Zeitung, 2014), modular systems are even established across company borders (Focus Online, 2013; Handelsblatt, 2012a).

[8] The modularisation aim of the listed examples may not mainly be on complexity reduction. This could be slightly different compared to the main focus of this work. For example, upgradeability and reduction of waste seem to be one of the main goals of modular smartphone projects currently (Kazi, 2015; Khedekar, 2016; Phonebloks, 2016).

3.3.2 Modular systems in automotive industry

Among all the examples from various industries, the dimension of modular systems in automotive industry remains the most prominent one.

Honda's way to serve totally different demands in the US, Japan and Europe was a flexible and stretchable platform for the world market. Even though Honda had to invest large sums in the flexible platform, it could cut cost by 20 % and $1,200 per car compared to single product development. The Honda Accord could be developed for $600 million compared to the competitive model Ford Taurus with $2.8 billion (BusinessWeek, 1997).

Scania's modular system which evolved over many years targets to serve current and future customers, with individual and optimised products, in parallel with a limited and controlled number of parts and interfaces (Scania, 2008, 2000; Scania Group, 2011). Synergies that are created between products are claimed to reduce the number of components by 50 %. This lowers the cost for research and development by 30 % - 50 %, production by about 10 % and sales and service by about 30 %. For instance, 85 % of the components in a tourist coach chassis are shared with a truck chassis (Scania, 2009).

In general, automotive industry seems to expect extremely high complexity entailed by new technologies and higher competition in future (Krepper, 2011). Moreover, the industry tends to be present in more and more markets while serving cars for each class type. For this reason, car manufacturers are striving to replace their platform-driven architectures with module-driven product architectures (see Sections 2.3 and 2.4) as they regard this as a more flexible approach with more potential for cross-model commonalities. Besides volume-driven cost effects in purchasing and production, they expect advantages throughout the whole value stream. These expectations have led to a "hype" about modular systems at car manufacturers. Just to name a few, Tata Motor wants to introduce a new modular system, the "Code X4", to launch hatchbacks, sedans and multipurpose vehicles (MVPs) from the same modular system (The Economic Times, 2013). Renault-Nissan intends to build cars around a Common Module Family, the so called CMF, (The Economic Times, 2013). Peugeot and Citroen want to derive cars from their new Efficient Modular Platform EMP2 (PSA Peugeot Citroen, 2013). Volvo plans to launch new car generations from the Scalable Product Architecture, SPA, (Gomoll, 2013).

Modular strategy of Mercedes

Mercedes invests enormous sums (3 billion Euro for organisational restructuring alone, without initial platform investment) to restructure their organisation around the introduction of four modular systems for vehicles and one modular system for power-trains (MPA). In this course, manufacturing will also be reordered from relatively autonomous production facilities towards production facilities that are guided by the standardised product architecture (Daimler, 2014). The four vehicle platforms are intended to cover all of Mercedes car models (Daimler, 2014):

- the rear-wheel drive architecture (MRA) for the S-, E-, and C-Class
- the front-wheel drive architecture (MFA) for the A-, B-, CLA- and GLA-Class
- the architecture for SUVs (MHA) for the M-, R-, GL- and G-Class
- the architecture for sports cars (MSA) for the SL- and SLK-Class

With the new modular strategy, Daimler plans to launch 40 new car models, including 12 new car models without predecessors until 2020 (Daimler, 2014). Daimler claims that the cost savings with this new approach would be huge and that the launch of a multitude of additional vehicle variants would not be possible without the extended modularisation approach (CARSCOOPS, 2014).

Platform strategy of BMW

BMW is reducing its platforms to just two across different models and brands (with MINI and Rolls-Royce). The platform consolidation is expected to support BMW desire to remain profitable during unstable periods while being able to subsequently launch new model variants. With this strategy, BMW plans to reach unit sales of two million and more from 2016 on (Vijayenthiran, 2015). The two platform architectures of BMW are as follows (Boeriu, 2015; Kurylko, 2014):

- The Cluster Architecture (CLAR, initially the 35up) will equip rear-wheel drive cars like the 3-, 5-, 6-, 7-Series or the X3, X4, X5, X6 and X7.
- The UKL platform will be the base for smaller front-wheel drive cars like the 1- or 2-series. The same platform will also be shared with the MINI.

All-wheel drive variants of BMW will draw upon the CLAR and the UKL platform (Kurylko, 2014).

Volkswagen Group’s modularisation transition

James Scoltock, the editor of the “Automotive Engineer” states that “when modularity is mentioned during discussions at the various events we attend, conversation is soon directed to Volkswagen’s MQB architecture” (Scoltock, 2014). In fact, Volkswagen Group’s (also referred to as Volkswagen within this section) undertaking seems to be cutting-edge modular system development.

Volkswagen expects that the modular strategy makes it possible to reduce development and production costs by 20 %. Together with other effects in the value chain, this would reduce unit costs and one-off expenditure by 20 %. Moreover, engineering hours per vehicle are claimed to be reduced about approximately 30 % (Krepper, 2011; Pötsch, 2011, p. 18). In turn, this leads to a decreased time to market for new model variants (e.g. new Golf VII) by about 30 % (Autobild.de, 2013).

The former Volkswagen head of development Dr. Ulrich Hackenberg stated that the modular strategy enables much higher flexibility as conventional platform strategies. According to his estimation, over 60 % - 70 % of a car can be defined as common modules which can be used for saloon cars as well as for off-road vehicles. For instance the new A1 of Audi will share the common base of the VW-Polo class. Cars can be varied in width, length, height, wheel dimension and wheelbase. Common modules are amongst others power train, air conditioning system, seats, door locking systems and window lifter. The flexible modules permit to use different gas types as well as the accommodation of batteries for hybrid- and electric cars (Krepper, 2011).

Moreover, the Modular Petrol Engine System (MOB) and the Modular Diesel System (MDB) are aligned to the modular systems of vehicles. As the different engine variations are mounted at the same angle and at the same interface, the "exhaust line, drive shafts and transmission location" can be efficiently standardised. This is claimed to reduce "the number of engine and transmission variations in the Group's MQB system by nearly 90 %" (Volkswagen Group, 2015).

The technical concept behind Volkswagen's intensified modularisation strategy is the transition from rigid underbody platforms that generated synergies mainly between the models of a certain value class towards modular systems. The modules of a modular system are seen as the main ingredient for synergies inside a value class and between different value classes (Pötsch, 2011, p. 18). The underlying concept is depicted in Figure 37.

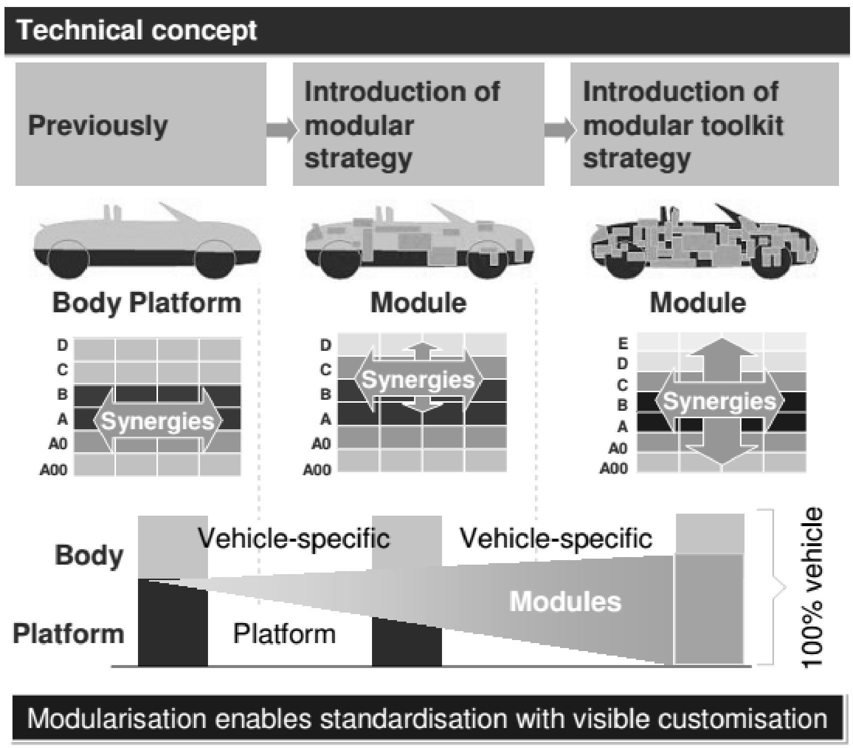


Figure 37: Technical concept of Volkswagen's competitive advantage from its modular system strategy (Pötsch, 2011, p. 18)

Remarkably Volkswagen Group aims to derive almost their complete range of car models (with sales volumes of more than ten million cars in total in 2014) from only four modular systems (Volkswagen, 2014). The term "modular system" used within this work was formerly referred to as “modular matrix” and is now referred to as “modular toolkit” by Volkswagen. Volkswagen’s four modular systems are as follows (Volkswagen Group, 2015):

- New Small Family (NSF): This modular system will be used for ultra-compact cars like the VW Up!, the Seat Mii and the Skoda Citigo (Autobild.de, 2013).
- Modular Transverse Toolkit (MQB): The MQB is used for small and medium sized vehicles with transverse engine mounting position. The MQB shall serve more than 40 model variants by 2018 and it shall equip more than 4 million units by 2016. Examples for models that can be equipped with modules from the MQB are VW Polo, VW Golf, VW Passat, VW Scirocco, VW Touran, VW Caddy, VW Tiguan, Audi A3, Audi TT, Seat Leon, Skoda Octavia, Skoda Superb (Autobild.de, 2013) and partly the Skoda Fabia (Skodaportal, 2014; WORLDCARFANS, 2014).
- Modular Longitudinal Toolkit (MLB): The MLB is used for bigger sized cars with longitudinal engine mounting position. Examples for modules that are capable to use modules from the MLB are the Audi A4, Audi A5, Audi Q5, Audi A6, Audi A7, Audi A8 and the Porsche Macan (Autobild.de, 2013).
- Modular Standard Toolkit (MSB): This modular system is intended to boost synergies between high-end sports cars or premium brands like Lamborghini and Porsche (Autobild.de, 2013).

The interplay between the different modular systems and how they contribute to Volkswagen’s vehicle classes is shown in Figure 38.

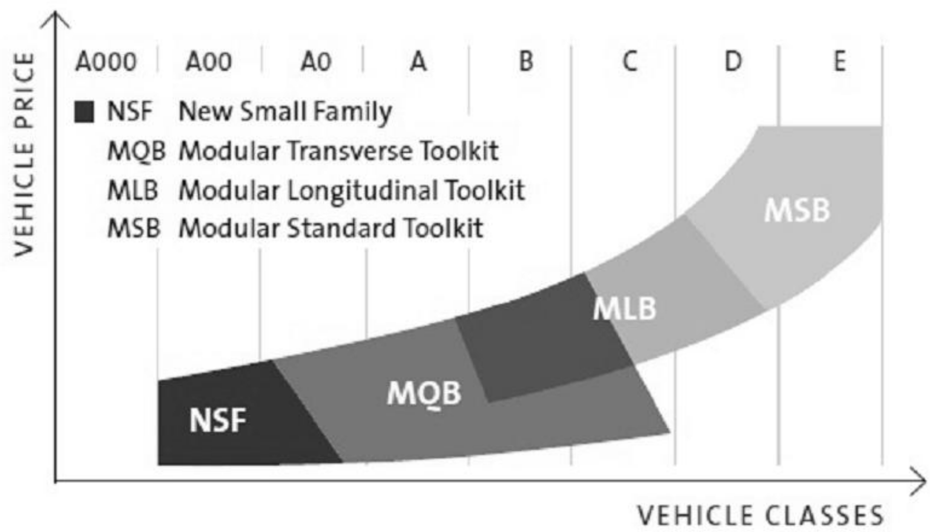


Figure 38: Modular systems for passenger cars at Volkswagen (Volkswagen Group, 2015)

In order to bring on the one hand the benefits of standardised vehicles down to production and on the other hand to being able to manufacture flexible solutions, Volkswagen establishes a strong link between the modular system for products and a so called modular system for production (MPB), (Volkswagen Group, 2015). It is claimed that the standard installation of the MPB is capable to deliver 30 vehicles per hour of the same model. Nevertheless, the MPB can be flexibly extended with pre-defined equipment at predefined positions so that it is possible to manufacture 60 different vehicles of any model and of any brand per hour (Volkswagen, 2012; Volkswagen Group, 2012).

3.3.3 Problems during modularisation transition in industry

The reported transition of industry towards modular systems with a new dimension is just at the beginning. In 2015, major car companies are either at an advanced planning stage or at the early implementation state of their new modular platforms. It can be assumed that the immense reuse of modules, explosion of car variants, ambitious growth strategy, reorganisation and industry-wide collaboration even among competitors that is based on the possibilities of new modular systems will radically change automotive industry. Moreover, other industries already jumped on the "modular" path, paved by the promises of new modular system pioneers. Up to now, it is impossible to foresee whether these undertakings will be successful or if some, if not more, of the risky ventures will make a crash-landing.

Nearly all cases from industry only report about the positive effects of higher variety and commonality and the resulting impact on company processes (Persson and Åhlström, 2006). The most frequently mentioned disadvantage of "platforming" and "product architecting" is the high initial effort and investment (Feitzinger & Lee 1997; Lehnerd 1987).

Camuffo (2002) discusses advantages and disadvantages while analysing Fiat's World Car, the Palio. Indeed, worldwide standards enabled by modularisation, make it possible to amortise development costs much earlier due to repetition. Know-how can be accumulated and used in every new site by replicating the same organisational concepts and working methods. Therefore, production capacity can be increased simply by reproducing the same module in parallel and standard working practices can be shifted to all process, service and subassembly suppliers (Camuffo, 2002, p. 24).

However, the case study showed that the promising detailed benefits are not that easy to implement as such broad-scope strategies increase project complexity considerably. This increase began when Fiat started to enter countries with significantly different requirements. Much more had to be considered than the required product customisation. Local adjustments have to be made in technologies, organizational structures and management practices, especially in existing and overtaken sites (Camuffo 2002, p.25).

> *There, in fact, the strong commitment to global optimization and cross-country standardisation has been challenged by the peculiarities of local competition, institutional constraints and cost factors. For example, local content constraints and tough price competition by other local and global OEMs (Tata, Daewoo, Ford, etc.) have pushed Fiat managers toward a major customization and nationalization of Siena. (Camuffo 2002, p.25)*

Camuffo (2002, p. 26–27) further observes that “modularization is a complex, slow and controversial process” which “will negatively affect OEMs' capability to differentiate and characterize their vehicles’ and brand identity vis a vis competitors”. Moreover, he advises not to underestimate the complex link of modularisation to “different systems and variables” like product design, technology, manufacturing equipment and other company areas.

Even the modular world of the forerunner of modular systems, Volkswagen Group, is not always that neat. With the intention to follow the modular system strategy of their passenger car unit, Volkswagen wanted to bring their utility vehicle brands VW, MAN and Scania onto the same platform. However, after Volkswagen had invested 500 million Euros into the project, it encountered serious problems so that it had to stop the project. This means that a common modular system for utility vehicles is currently not progressing, and to date, such endeavours can be seen as having failed (Handelsblatt, 2012b).

Even more, although the modular system strategy for passenger cars of Volkswagen is still at an early stage, it is quite obvious that the transition does not run as smoothly as expected. So far, positive effects of the modular system have not been fully realised. In contrary, due to problems launching the new Golf and the Passat from the MQB, Volkswagen had to invest 300 million Euros additionally! This has led to a nervous atmosphere and job changes at the top management level (Freitag, 2014).

Even worse, Martin Winterkorn, former CEO of the Volkswagen Group, conceded that *the transition toward MQB has not led to the expected simplification, but that the transition has been afflicted with unforeseen problems*. Without doubt in the modular strategy itself, he called the efforts of overcoming the obstacles of the “*bumpy transition”* a continuous, strenuous challenge (Automobil Produktion, 2014). Thus, the modular system of Volkswagen remains one of the main issues of the vehicle manufacturer (Spiegel Online, 2015).

3.4 Summary and knowledge gap

Based on the logic and the promising theoretical benefits of modular systems (see Chapter 2), many researchers have attempted to develop support methods for product architecture design. Thus, it has been shown that the product architecture is an important part of the general engineering design process. In initial phases of the design process, the product architecture is an abstract construct. Afterwards, the product architecture becomes more concrete from phase to phase, until the conceptual product architecture is transferred into the bill of materials and into the description of bill of material items. This is the point where the product architecture influences the performance of different company functions.

Modularisation methods influence the way that elements of design artefacts are clustered – either in a more abstract state in the early design process or in a more concrete state in later design stages or for product redesign purposes. The applied methods from literature can be characterised according to the goal that they are attempting to pursue. Methods that have the goal to change the degree of modularity either lead the product architecture into a more integral or modular architecture by drawing upon principles of modularity. Other methods attempt to improve diverse strategic effects of the product architecture or its influence on PLC or value stream issues.

In parallel to academia, it seems that some industries are on the verge of deriving more products than ever before from common modular product architectures in the coming years. Given the high investments and the high expectations that are put on new modular systems in practice, it can be assumed that the *outcome of modularisation transition* is one of the most critical contributors to the competitiveness of companies pursuing a modularisation strategy.

However, much of the research up to now has sought to answer what the logic behind product architectures is, what benefits can be expected of modular systems and how product architectures can be established.

The literature review has shown and, thus, it reinforces the claim in the introduction to this work, that it is not yet clear how companies can be supported to tackle the overall issue of transitioning towards modular system development. As was pointed out in the introduction of this thesis, this major gap can be further detailed into following points:

- As was shown in the introduction of this work and in Sections 3.1.6 & 3.3.3 of the literature review, there is urgent need to show how modular systems can be implemented into industrial practice. Even though modularisation seems to work well on a blank sheet of paper, transfer to industry, with associated legacy issues, are fraught with diverse additional challenges.
  Moreover, it has been shown that modularisation has been studied “mostly in static situations”. No “longitudinal studies” have been applied to capture the reality that “no system is really static”, that “products change, processes evolve, organizations adapt, and innovations appear, and all of these changes are accelerating” (Fixson, 2007, p. 98).
- Although considerable amount of research has been undertaken on how to establish modular product architectures, the introduction and Sections 3.2 & 3.3.3 indicate strong need to research solutions how product architectures can be kept stable over time without eroding, diverging or breaking apart. It is the understanding of this work, that it is not sufficient to stop “architecting” after establishing the product architecture in the context of a new product development process. Rather, reality presents an environment where the majority of designs evolve iteratively from past designs.

Having discussed the state of the art in the literature and in industry which clearly identify important deficiencies of current research, the thesis will in the next chapter move on to show how the knowledge gap can be closed.

4 Research direction and methodology

This chapter discusses in more detail the research aim to help sharpen and focus the research direction of this study (see Section 4.1). In addition, a research methodology for the work is developed and is outlined in detail in Section 4.2. Section 4.3 presents the research activities applied for this research work. The overall approach to data collection is described in Section 4.4. An overview of how the data that was collected within the case companies contributes to the contents of this thesis is presented in Section 4.5. A discussion of the outcomes and the research deliverables of the work is then included in Section 4.6 (see Figure 39).

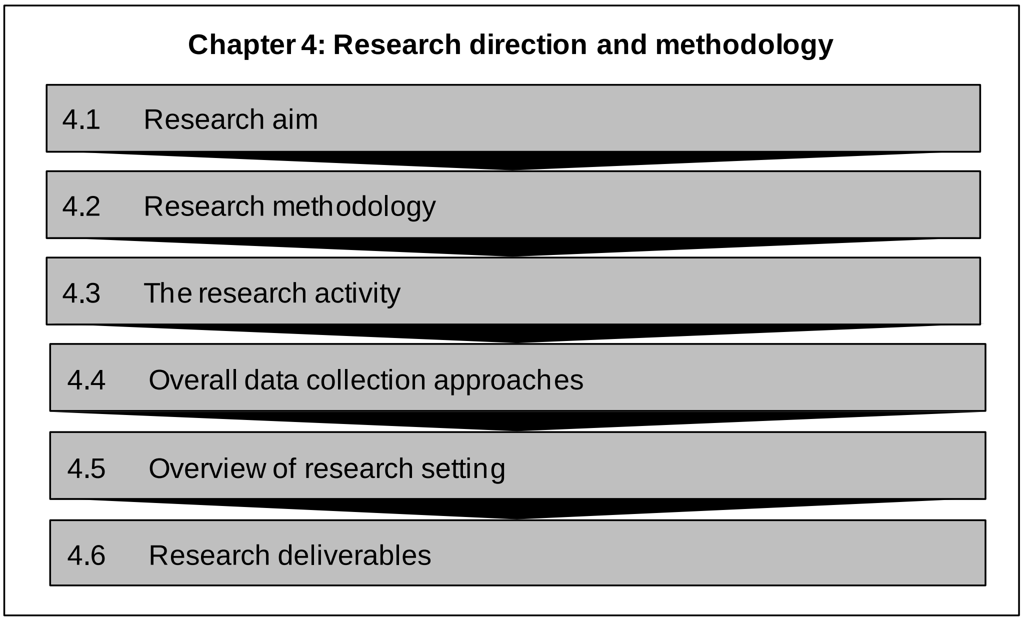


Figure 39: Elements of chapter 4

4.1 Research aim

As previously introduced the overall aim of this research is: *To identify and test critical issues and important factors associated with support for the transition towards modular system development with stable product architectures.*

Based on these findings, it is proposed to develop engineering design support for the transition.

Having outlined in the introduction what the research questions and research objectives of this work are, as shown in overview in Table 1 which covers objective, research activity and proposed deliverables. Here the research objectives are considerably expanded and are then used to drive the methodology.

In order to achieve the research aim, this research work will be divided into two parts which examine four main research questions:

- Part 1: Establishing a deep understanding of transitioning towards modular system development in industry:
    a) What are the vital elements that have to be considered for transitioning towards modular system development?
- Part 2: Developing support for the transition in industry based on the findings of part 1:
    b) Does a modularisation assessment framework support companies in making the transition? What is an appropriate modularisation assessment framework?
    c) Does the assessment of product architectures support companies in making the transition? What are appropriate metrics to assess product architectures during the transition?
    d) Does the provision of product architecture information in standard IT-Systems support companies in making the transition? What is an appropriate approach for the IT-integration of product architecture information?

The detailed research objectives that were derived from above research questions and from above mentioned aim are as follows:

RO 1: To identify and test vital elements for modularisation transition

a) RO 1a: To identify critical issues that companies encounter during transitioning towards modular system development

b) RO 1b: To establish important factors that must be in place for transitioning towards modular system development

c) RO 1c: To identify and test support for transitioning towards modularisation

RO 2: To develop a modularisation assessment framework for companies that transition towards modular system development

a) RO 2a: To identify an appropriate assessment framework for modularisation transition

b) RO 2b: To develop and test a modularisation assessment framework for modularisation transition in industry

RO 3: To develop metrics for transitioning towards modular system development

a) RO 3a: To derive requirements for modularisation metrics applied in industry

b) RO 3b: To find out use and limits of existing modularisation metrics

c) RO 3c: To develop and test metrics for modularisation transition in industry

RO 4: To develop an approach for provision of modularisation information in companies

a) RO 4a: To identify requirements for provision of modularisation information

b) RO 4b: To identify relevant information for modularisation transition

c) RO 4c: To develop and test an approach for integration of modularisation information into standard industrial IT-systems

The next section will show how the research objectives will be achieved by considering existing research, scientific rigour and involvement of industrial practitioners.

4.2 Research methodology

As stated earlier, it is the purpose of this work to study the overall issue of transitioning towards modularisation with a specific focus on implementation into industrial practice and on prolonged stability of the modular system architecture. In order to capture the overall issue, it is claimed that a longitudinal field study in industry with multiple research methods is the most appropriate approach to bring light onto the underrepresented research area.

The next section describes the research methodology of this research work by presenting the overall research framework and the research context.

4.2.1 Design Research Methodology (DRM)

The purpose of engineering design research is "generating knowledge about design and for design" (Horvath 2001 cited by Blessing & Chakrabarti 2009, p.5). In other words, research in engineering design is about understanding design (descriptive research), and developing support for design (prescriptive research) in order to improve design in terms of processes and created artefacts. "Design research must be scientific in order for the results to have validity in some generic, theoretical as well as practical sense" (Blessing and Chakrabarti, 2009, p. 9). Blessing and Chakrabarti (2009, p. 6) identified three main issues in past and current engineering design research: lack of overview of existing research, lack of use of results in practice and lack of scientific rigour. In order to address the three issues and to systematically develop and validate knowledge, Blessing and Chakrabarti (2009) introduce a "Design Research Methodology". Design Research Methodology (DRM) shall guide researchers in engineering design to achieve results which are indeed practically and scientifically valid.

The methodological framework consists of four stages which are closely linked: Research clarification, descriptive study I, prescriptive study and descriptive study II (Blessing and Chakrabarti, 2009, p. 14–38):

1. The main purpose of the Research Clarification stage is the identification of research questions and objectives based on background information about the existing and de-

sired situation. Moreover, the clarification phase determines a focus on what to find out during the descriptive study I.

2. The Descriptive Study I aims at a deep understanding of the existing situation and the detailed factors that influence the research goal and serves as input for the effective development of support during the prescriptive study. Findings that impact the development of support are also described here. During this stage, overall and measurable success criteria are identified and brought into relation to each other. This step is closely linked with the creation of the impact model in the prescriptive study.
3. The Prescriptive Study first ensures appropriate support, by selecting relevant factors from descriptive study I by bringing them into a cause-effect relationship model with overall success criteria. Based on this impact model, the intended support is developed and documented.
4. The developed support from the prescriptive study is evaluated during Descriptive Study II. This is either done by evaluation of the application or the achievement of identified success criteria.

It is not possible for every research project to cover every stage of the research framework in detail. For this reason, Blessing and Chakrabarti (2009, p. 18–19) identify seven different research project types that can be selected based on the research question, the required research coverage of a certain and the available resources. According to the seven research types, a research stage stage can either be “review-based”, “comprehensive”, “initial”, or in certain situations be omitted. A review based study focuses solely on the review of literature. A comprehensive study includes both, a literature review and own results produced by the researcher. An initial study closes the research project by showing the impacts of the research and making the research results usable by others. Figure 40 gives an overview of the research framework and the existing research types.

It has to be pointed out that the Design Research Methodology is a model that does not exactly reflect reality and considers the iterations and stages that are worked through in parallel. Theoretically, the starting point for the research could also be in any stage of the Design Research Method Framework (Blessing and Chakrabarti, 2009, p. 17).

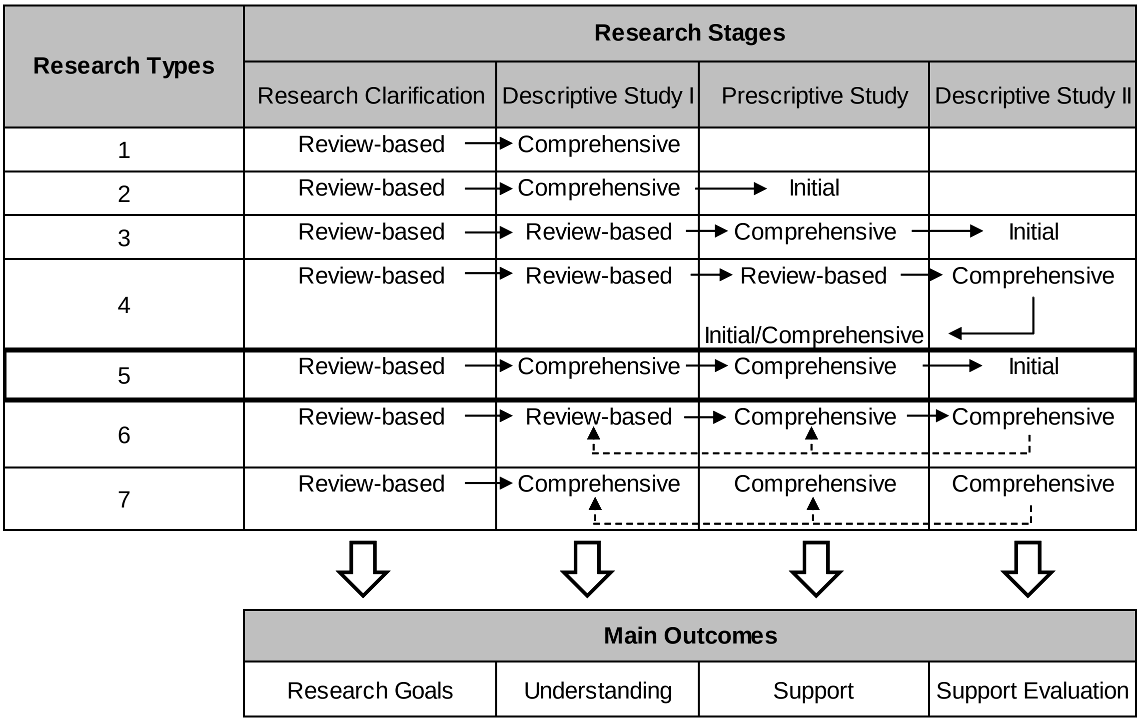


Figure 40: DRM research framework and suggested research types (Blessing and Chakrabarti, 2009)

4.2.2 Research framework of this study

As marked out in Figure 40, the research methodology of this work is essentially type five. It starts with a review-based research clarification, which is followed by a comprehensive descriptive study I, a comprehensive prescriptive study and is closed by an initial descriptive study II. Following research activities are done in each research stage:

- Research Clarification: Identifying problems of industrial practitioners

This research stage starts with a problem definition of increased complexity in manufacturing companies, based on literature (see Section 1.2) and based on the problems of the collaborating case companies. This is followed by a literature review about product architectures as means to improve complexity (see Chapter 2). The literature review also reveals the state of the art in product architecture improvement and a preliminary research direction through identified gaps (see Chapter 3). The defined focus of this research work (see Section 4.1) is based on a comparison between what is available in literature and initial findings identified during the case study (see Section 4.3 and Appendix B). The results of the research clarification lead to research about the overall issue of modularisation transition in industry.

- Descriptive Study I: Generating a deep understanding about modularisation transition

This research stage comprehensively focuses on the identification of issues, important factors and support that are relevant for modularisation transition. Relevant factors come from different aspects. Literature in each aspect is analysed for important factors, but as there was no satisfying coverage in literature (see Chapter 3 and Section 5.1), a longitudinal case study is done to identify those factors that really matter in industry. Data collection for the case study is done through a mixed method approach comprising observer approach, participant-observer approach, document analysis, experiments, surveys, interviews, discussions, workshops and action research. There will be a more detailed description of research methodology used for this research stage in Chapters 4.4, 4.5 and 5.2. The research results of this stage are issues that companies encounter during modularisation transition, important factors that must be in place for transition and identified and tested support (see Chapter 5). These results are used as input for the development of support in the next research stage.

- Prescriptive Study: Modularisation assessment framework, metrics, information provision

Identified issues, important factors and use and limits of existing support are brought into a support framework for modular system development in Section 5.5. This can be seen as starting point for the development of detailed support for industry: modularisation assessment framework (see Chapter 6), modularisation metrics (see Chapter 7) and modularisation information provision (see Chapter 8). The research methodology for each support is described in detail in Sections 4.3, 4.4, 4.5, 5, 6, 7 and 8. Each detailed type of support is iteratively developed by checking it against requirements from literature and the requirements from engineers and engineering managers in industry. The output is evaluated within the next research phase.

- Descriptive Study II: Validation of modularisation transition support

The purpose of this research stage is to show how far the developed design support (i.e. modularisation assessment framework, modularisation metrics and modularisation information provision) can be applied to support engineers and engineering managers during modularisation transition of their organisation. It is seen as evaluation criteria that this research helps to remove issues during modularisation transition. The focus in this stage is on the evaluation of application (e.g. usability, applicability, and relevancy) rather than on evaluation of actual success (e.g. cost savings through modularisation). Reason for this focus is the time delay between modularisation transition and its effects in the organisation. Detailed description of validation of each type of support is given in corresponding chapters 6-8.

Figure 41 shows an overview of content and link between the above described research phases, corresponding research objectives and deliverables.

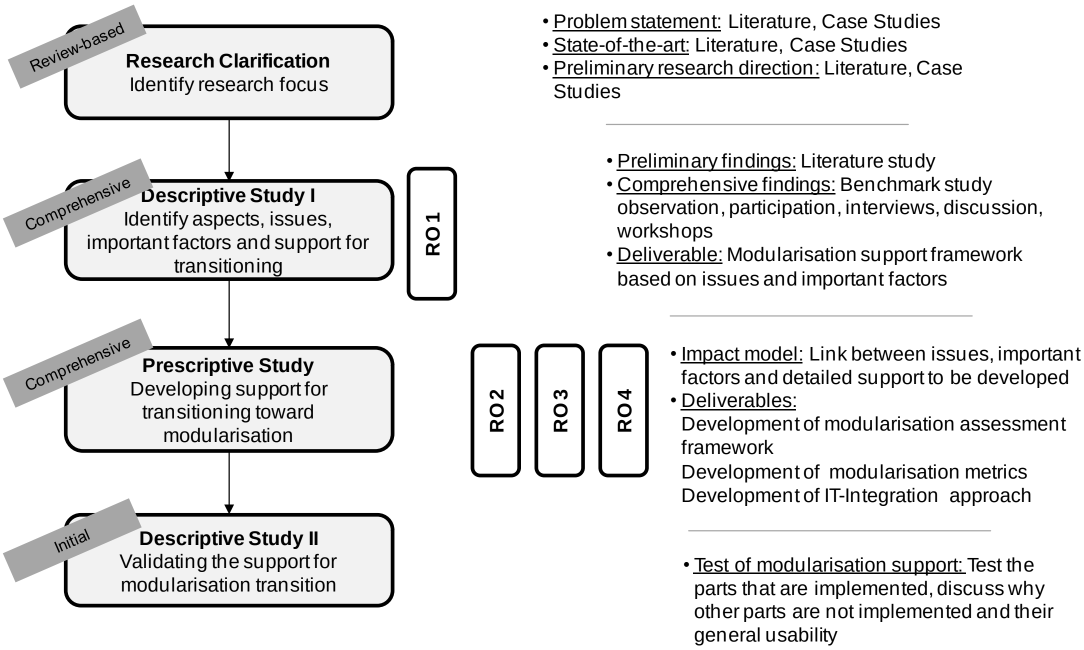


Figure 41: Overview of research stages and research objectives

4.3 The research activity

There are different elements of research activity associated with this research. This spanned from end of 2010 until end of 2015 (see Figure 43) and covered 30 international industrial cases. These are summarised below.

4.3.1 Case study approach

In theory, there are two different approaches how to learn more about modularisation transition. First, it is possible to study many companies transitioning towards modularisation. This “cross-case” approach had to be equipped with a statistical significant sample size (Gerring, 2007). Such an approach is used if a broad understanding about the widespread of a phenomenon across a population has to be generated (Flyvbjerg, 2011). Second, it is possible to learn more about modularisation by studying a small sample size transitioning toward modularisation. Such a so-called “case study” is chosen if intensive investigations have to be carried out (Gerring, 2007), if “how” and “why” questions have to be answered (Yin, 1994) and if research has to be done within real-life context (Robson, 1993). As the distinction between “case study” and “cross-case study” is a matter of degree (Gerring, 2007), a case study approach was chosen for the purpose of this work with a main primary case and several supporting secondary cases, that is, “multiple case studies” (Robson, 1993) with few well-selected cases. Other reasons why the case study approach was chosen for this research are:

- It helps to generate in-depth knowledge of the whole by focusing on a key part (Gerring, 2007).

- It helps to generate understanding about context, process and causal mechanisms (Flyvbjerg, 2011).
- It helps to generate high conceptual validity (Flyvbjerg, 2011).
- Case studies typically combine different data collection methods (Eisenhardt, 1989; Robson, 1993).

Consequently, a case study approach is the most appropriate way for this research enquiry. However, it can be argued that deriving research findings from a case study is causing problems when it comes to generalising the findings (Flyvbjerg, 2011). These concerns can be removed by application of different research methods, by analysing the research findings in the context of existing theory, by discussing the findings with multiple investigators, by publishing research results, by presenting research results at conferences and by triangulating the research findings with other case companies (Eisenhardt, 1989).

For confidentiality reasons, it is not possible to name the collaborating industrial partners. Moreover, for the same reason, details that could be used for identification were requested to be removed. Nevertheless, a short characterisation is given below and in Appendix B.

4.3.2 Primary case company

The primary case company is an international manufacturer for heating, ventilation and air conditioning (HVAC) appliances. The products are diverse and comprise HVAC products like controls, conventional and condensing oil and gas technology, solar systems, heat pumps, biomass heating systems, combined heat and power generation, ventilation, radiators, buffer tanks and air conditioning for residential, commercial and industrial purposes. Products are sold individually or combined in order to sell them as a complete system.

Products of multiple brands are developed and produced in different sites worldwide. The different brands evolved through mergers and buyouts from historically different companies.

The selected case company inherits all complexity drivers of Chapter 1. The need to serve more markets with diverse technologies and more product features on competitive cost made the management board come to the decision to implement complexity reducing measures. Transitioning towards modular system development has been seen as strong lever to decrease internal complexity while increasing variety offered to the customer. In order to achieve this, the central engineering department was given the task of implementing appropriate measures for the company-wide change towards modular system development. This department is the main research site of this research work. The central research site allowed getting valuable insights into different development projects across the globe in addition to the overall modularisation transition project.

4.3.3 Secondary cases

The primary case company is part of a larger group which comprises various central departments and various independent organisational entities from different industries like packaging technology, household appliances, industrial products (e.g. drives and controls) or automotive. The relationship to these companies made it possible to get insights into totally different secondary cases for benchmarking and triangulation reasons. Moreover, the primary case company collaborated with several different consultancies with academic and industrial background during the study. This allowed obtaining fine insights into state of the art approaches. Regular participation at research conferences was used to establish connections to researchers from other companies (e.g. from vehicle industry) with the same or similar research problem. For the triangulation of findings, a further manufacturer from systems engineering was consulted. This company develops equipment for connected safety-critical systems.

It is argued that the selected research cases presented in this section build a scientifically proper base for collecting and analysing data with different methods in order to solve the stated research problem.

4.3.4 Longitudinal research study

The previous paragraphs described how the required depth for this research enquiry can be achieved. This section adds a further dimension: time. Like done by most researchers in this field, it would be possible to go into a company to study a "snapshot" of how the company is designing modular systems or if a certain method is applicable in industry. Such a research design would be pure cross-sectional research where measurement is done once for a case, where the measurement of each research subject applies to a single period and where the measurement of each research subject for each case occurs within a "sufficiently narrow span of time" so "that the measurements may be regarded as contemporaneous" (Menard, 2002). However, in order to answer the research questions, the transition towards modular system development is seen as a dynamic process with different periods, iterations and an appropriately long span of time. For instance, distinct periods could be the periods before and after a certain type of intervention like crafting the plan of the modular system or implementing a specific type of modularisation support. For such a case, it is suggested that a longitudinal field study has to be conducted. A longitudinal field study is a study in the field that collects data at "two or more distinct periods, for those distinct periods, on the same set of cases and variables in each period" (Menard, 2002). Longitudinal research has the purpose to reflect a) "the experience of individuals as they age or pass through successive stages" and "to estimate the parameters, efficiently and without bias, of any dynamic process" (Menard, 2002). Figure 43 shows the relation on a high level between different research periods in above mentioned cases within the time span of this longitudinal field study approach.

In order to being able to triangulate, to evaluate and to generalise findings from the cases mentioned in the previous section, a variation from "classical" longitudinal research like it is applied in social sciences has been designed. In addition to the primary case company, different secondary cases were studied in different distinct periods. That also allows reflecting on a longer time span. For instance, if a company that started modularisation transition ten years ago is asked to share experience on modularisation issues retrospectively, a period of ten years can be covered within a five year research project. This variation is seen as valid from a longitudinal research methodology viewpoint (Menard, 2002):

> *It is possible, for example, to have a revolving sample in which subsamples may be dropped for one period, then re-included in the sample in a subsequent period. It is also possible to have a panel design in which cases are dropped, without replacement, after they meet some criterion (e.g., age 21). This latter design would result in a monotonically decreasing sample size that could pose problems for analysis of data from later years of the study (unless the design were further modified by replenishing the sample with new respondents from younger cohorts).*

According to this definition, it is scientifically valid to drop and replenish cases if they can be related to the same criteria (e.g. three years after start of transition) and periods (e.g. maintenance period of modular system). Consequently, the research approach of this study also includes a "Revolving Panel Design" (Menard, 2002):

> *Revolving panel designs collect data on a sample of cases either retrospectively or prospectively for some sequence of measurement periods, then drop some subjects and replace them with new subjects. The revolving panel design may reduce problems of panel mortality and repeated measurement in prospective studies (...) or problems of extended recall periods in retrospective studies. Retention of a particular set of cases over several measurement periods allows short-term measurement of change on the individual or case level, short-term analysis of intra-cohort developmental change, and panel analysis. Replacement of the subsample that is dropped in a measurement period with a new but comparable subsample of cases permits analysis of long-term patterns of aggregate change.*

In fact, this study includes a "classical" longitudinal study of the primary case company with a "Revolving Panel Design" which enriches collected data from secondary cases.

In terms of data collection, a longitudinal case study approach can be seen as a family of different methods which have to be applied according to the context of the case and research question (Gerring, 2007; Menard, 2002; Pettigrew, 1990; Robson, 1993). Those methods will be further handled in the next section.

4.4 Overall data collection approaches

Due to different characteristics of the research phases, each main research phase that is concerned with research results is described separately. For Research Clarification phase, there was no dedicated data collection method applied. The research focus as result of this research phase evolved exploratory through findings of literature studies and from research methods that were applied in Descriptive Study I. For instance, the initial research focus was revised and refined after interviews and participant observation. Descriptive Study II (i.e. research evaluation) is described for each research result or support type separately.

4.4.1 Data collection for Descriptive Study I

This section describes the basic rationale of how data was collected for the Descriptive Study I which is described in Chapter 5 in more detail.

A multi data collection strategy was chosen for this research phase. "Multimethod Research" is useful for triangulation of research findings that were derived from different research methods. This view from different perspectives improves the validity of overall research findings. For instance, if research findings from different methods are contradicting, then the validity of each finding is poor. In such a case, further investigations have to be made until there is reliable agreement between research findings from different methods (Brewer and Hunter, 2006). In the case of this research, further methods were applied when first research findings were challenged. For instance, in-depth interviews were conducted when document or field note analysis revealed unsatisfactory answers to certain research questions. Moreover, multimethod research proved particularly useful for the purpose of this research when it came to applying the most suitable research method for a specific situation. By applying multiple methods, a research problem can be attacked with "an arsenal of methods that have nonoverlapping weaknesses in addition to their complementary strengths" (Brewer and Hunter, 2006).

The applied data collection methods can be divided into different groups (Brewer and Hunter, 2006):

Firstly, in order to achieve *naturalistic and realistic research* findings, the researcher has to collect data in a *natural field setting* (Robson, 1993) from "indigenous inhabitants" (Brewer and Hunter, 2006). Thus, it is possible for the researcher to get increasing awareness and understanding of emerging theory of the natural flow of events in the field (Brewer and Hunter, 2006). Data collection methods that were applied to achieve realism comprise a participant-observer approach (e.g. attending meetings, workshops or collaborative work session) and in-depth, semi-structured interviews (e.g. interviewing a specialist, making interviews in a focus group setting or organising workshops with experts from different fields). The strength of these complementing data collection methods is that they provide detailed information to previously unknown issues and that they allow insights into contexts and interdependencies. Moreover, these methods elicit experiences, opinions, feelings and urge the researcher to directly resolve contradictions. On the other hand, these methods have to weakness that they are time-consuming and require a disciplined, diligent researcher. In addition, it is maybe the main weakness of these data collection methods that they require considerable effort on objectivity in order to overcome inherent subjectivity (Brewer and Hunter, 2006; Gubrium and Holstein, 2001; Mack et al., 2005; Robson, 1993).

Secondly, in contrast to naturalistic field work, survey research focuses on general questions that have to be answered with a statistical relevant sample size. Data collection methods that were used to achieve generalisability include (semi-)structured interviews (e.g. in order to sort out what industrial benchmark partners are doing) and questionnaires (e.g. to collect requirements for modularisation metrics) (Brewer and Hunter, 2006). It is the strength of these data collection methods that they are very helpful to produce objective, commonly agreed results. However, this strength has to be paid with less ability to produce deep insights. Another flaw of survey research is its dependency on motivation and understanding of participants (Gillham, 2000a, 2000b).

Thirdly, complementary to the first two data collection categories which require involvement of others, nonreactive research focuses either on unobtrusive or indirect observation of unaware research subjects like people, artefacts, archives or any other naturally occurring data sources (Brewer and Hunter, 2006). For the purpose of this study, product assessments and document analyses have been carried out. This allows generating evidence without manipulation or abnormal behaviour of research subjects (Brewer and Hunter, 2006). However, nonreactive research highly depends on correct interpretation of the researcher.

Finally, the last category is data collection through experiments. Experiments require control over events and measure “the effects of manipulating one variable on another variable”. Experiments are mainly used for hypothesis testing (Robson, 1993). Even though, this whole research work and its elements can be seen as some kind of “experiment”, lab-like environments with controlled and separate variables could only scarcely be achieved in the course of this research project. This is an intended contrast to already existing support. However, it was possible to conduct experiments to test developed modularisation support, for example on isolated methods, products, IT-Tools or on separate development projects. The result was that hypotheses were constantly and circularly created, tested, revised and refined during the course of this research.

To sum up, the use of multiple data collection methodology ensures to bring a high level of trustworthiness into the research. One data source, especially in the case of a longitudinal case study, can be checked against the other data source. This provides that contradictions in research information can be gauged out while preventing wrong conclusions (Brewer and Hunter, 2006).

A more detailed description of actual data collection and qualitative analysis of data for this research phase is given in Section 5.2.

4.4.2 Data collection for Prescriptive Study and Descriptive Study II

Design support for transitioning towards modular system development is based on data from the previous section, Descriptive Study I. In the course of this work, the development

(Prescriptive Study) and evaluation of support (Descriptive Study II) is combined as it is assumed that this research phase incorporates a constant feedback cycle between action and evaluation.

The theoretical foundation of this research phase is based on two different streams of thought. The first stream is grounded in action research from social sciences. It is one goal of action research to solve a problem in practice (e.g. with an organisation) by participating in an iterative change process with iterative cycles of intervention and evaluation (Billies et al., 2010; McIntyre, 2008). Beginning at Research Clarification and Descriptive Study I, action research starts with questioning a particular issue in order to come to new insights, questions and perspectives to a particular problem. For instance, this research work started with developing another modularisation method before it became evident that, first, the underlying problem is different than actually assumed and, second, that the responsive support needs to go beyond another modularisation method. Moreover, the understanding of this research phase also poses the requirements for the development and evaluation of the actual support. The various phases of action research are interconnected with each other in a spiral of a) questioning a particular issue, b) reflecting upon the issue, c) investigating the issue, d) developing an action plan, e) implementing the plan and f) refining the implemented support evaluation (Billies et al., 2010; McIntyre, 2008). For the purpose of this work, the spiral of these phases result in validated support for modularisation transition. The phases of action research are similar to the phases of the second considered stream.

The second stream of thought for support development and evaluation comes from engineering design research. It comprises following phases (Blessing and Chakrabarti, 2009):

- Task clarification: This phase establishes the requirements for the support.
- Conceptualisation: This phase generates different concept variants. The concept that is further pursued is selected based on most promising fulfilment of requirements.
- Elaboration: The elaboration phase transfers the support concept into a detailed description of the support.
- Realisation: During the realisation step, the support is developed and an intervention takes places to apply the support in practice.
- Support evaluation: This activity takes place throughout support development and in addition after support development. For instance, support can be evaluated by validating it against established requirements through expert interviews and test in a real application environment.

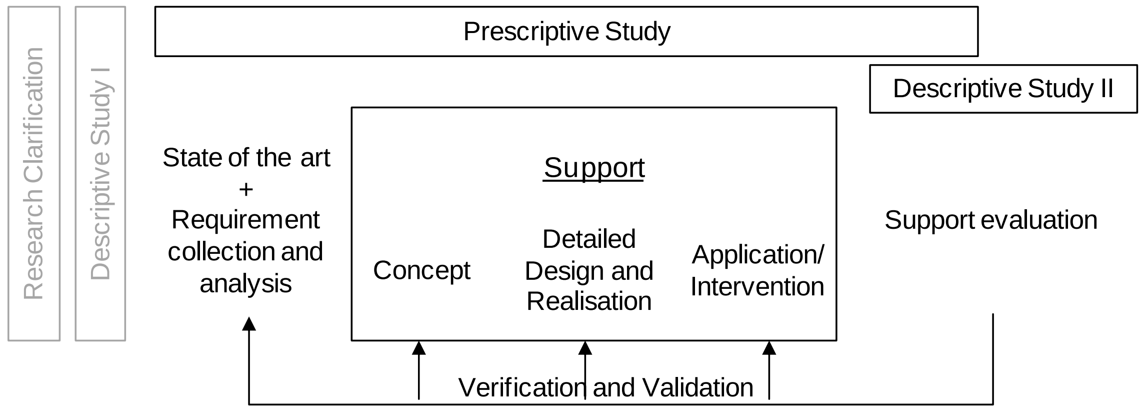


Figure 42: Support development and evaluation of Chapters 5-8, based on action research (McIntyre, 2008) and DRM (Blessing and Chakrabarti, 2009)

Figure 42 shows how support is developed in the context of this work. Each type of support and, thus, Chapters 5-8 start with the overall objective of the support. This is followed by following sections in each chapter:

- Brief state of the art section
- Requirements for the support type
- Description how the support was developed
- Description of the developed support (main part of each chapter)
- Findings from support evaluation

The detailed research methodology elements of Prescriptive Study and Descriptive Study II depend on the specific situation and the type of support developed. Therefore, context-specific details of the presented phases will be provided separately for each support type in Chapters 5, 6, 7 and 8.

4.5 Overview of research setting

Major parts of this research have been conducted in industry. The case study draws upon two different types of cases. The first case is the primary case company which sought to transition toward modular system development. The second type of case draws upon triangulation with different organisations that are either more mature in modularisation transition (mature cohorts) or that are quite at the beginning of transitioning toward modular system development (younger cohorts). More information on the characteristics of the primary and the secondary cases can be found in section 4.3 and in Appendix B.

The primary case company was studied by participating and by doing interventions in two different kinds of projects:

- Central Department:
  It was the goal of the central engineering department to implement and to accompany

the transition toward modular system development across different business units of the company. The research work in the central engineering department started with analysing the needs of the company for modularisation transition. After investigating different available scenarios how the transition toward modularisation can be made were investigated, a decision was prepared if modularisation transition pays off for the company at all. As the analysis came to the conclusion to further pursue modularisation transition, a plan was set up how the transition toward modularisation can be made. In order to study actual modularisation transition, research took place by observing, participating and intervening in activities guiding the company toward modular system development.

- Development Projects:
  In parallel to the overall perspective on modularisation transition in the central department, the research closely analysed and supported two different projects (preliminary project and pilot project) from different business units in developing modular systems. These studies gave real insights into the needs and issues engineering designers face during modularisation transition in their daily work and how this connects to the implementation activities of the central department. Moreover, during more mature research phases, developed support could be directly tested in those development projects.

Secondary cases were mainly used to get more mature insights into modularisation transition prior to starting in the primary case company, to constantly triangulate findings and to evaluate findings in younger secondary cohorts.

Based on the insights from the primary case and the secondary cases, actual research deliverables were developed. In order to come to RO1, the input of all cases was taken to analyse and condense vital elements for modularisation transition (see Chapter 5). RO2 concerns the development of a modularisation assessment framework which is based on the analysis of the primary and secondary cases. The modularisation assessment framework was developed and evaluated in the context of the primary case company, but validation took also place in a secondary case (see Chapter 6). Modularisation metrics of RO3 were developed based on the overall industrial case study. The modularisation metrics were applied, tested and evaluated in the projects of the primary case company (see Chapter 7). In order to being able to effectively support the modularisation assessment framework of RO2 and the generation of modularisation metrics of RO3, RO4 deals with the provision of modularisation information in a company's standard IT-systems. In parallel to the work on RO4, interventions, application and evaluation took place in the primary case company (see Chapter 8).

Figure 43 shows an overview how the contributions (see Chapters 5, 6, 7 and 8) of this research work have been derived and evaluated in the context of industrial cases.

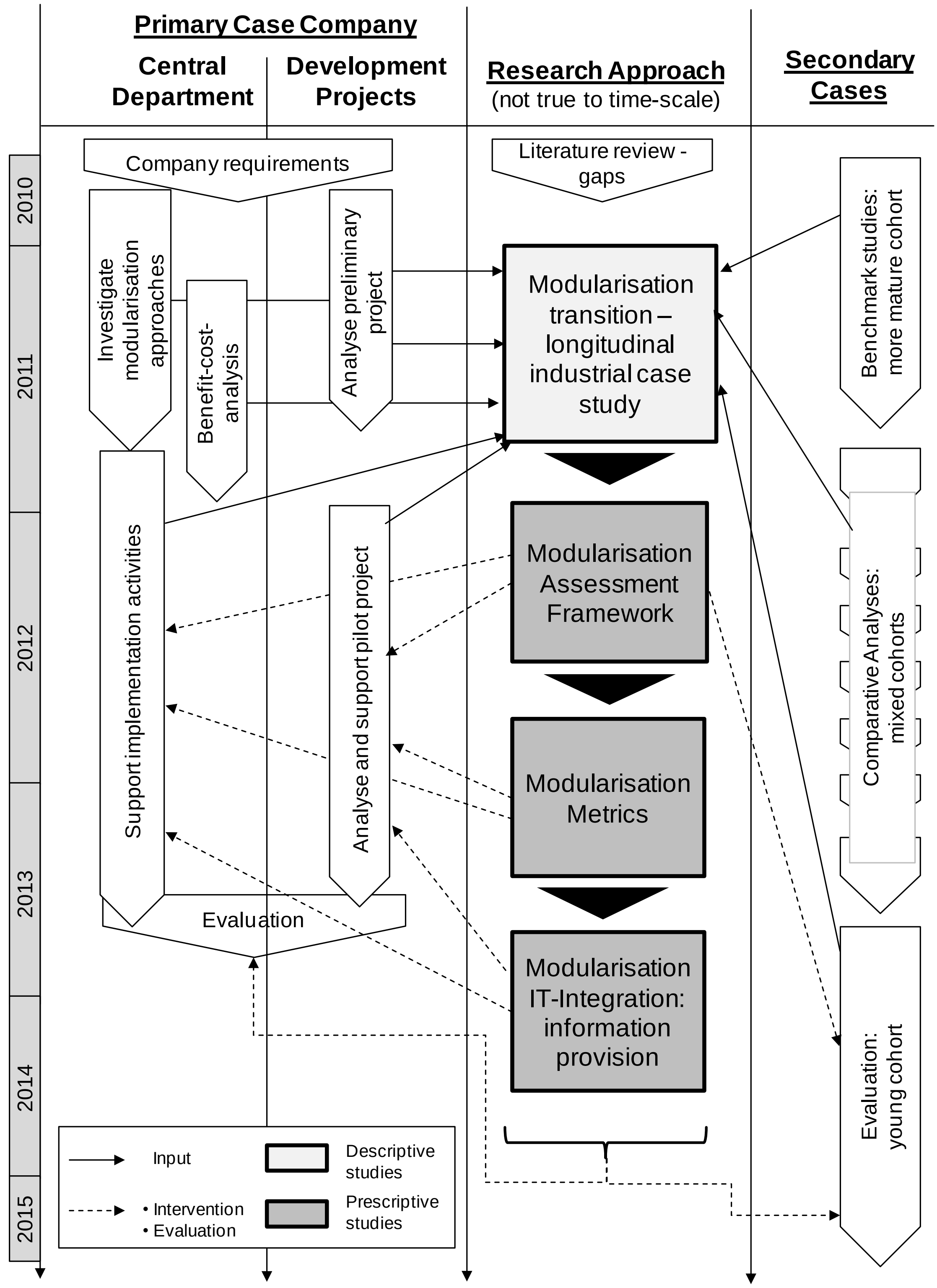


Figure 43: Overview of research contribution from Chapters 5, 6, 7 and 8 and its relation to the longitudinal case study in industry

4.6 Research deliverables

It is the goal of this research work to generate novel contributions for both, academia and industry. Therefore, this section describes what the contributions that initially were mainly developed for industry of this work are, where it can be found within this thesis and how they relate to academic research publications that were generated in the course of this work.

RO1 seeks to identify and test vital elements for transitioning toward modular system development in industry. The sub-objectives of this descriptive research phase deal with problems that companies encounter during modularisation transition, important factors that have to be established for smooth transition and available support that can be applied in order to remedy the identified issues. In sum, it is the goal to understand how companies can make the transition. Chapter 5 presents an *overall support framework for modularisation transition* which takes the results of RO1 into account.

Chapter 6 handles RO2 and presents a *modularisation assessment framework* that guides companies during the transition process taking into account the findings of Chapter 5.

Chapter 7 shows the results of RO3 by providing *modularisation metrics* that can be used in conjunction with the modularisation audit to control the performance of modular systems during and after transitioning.

Finally, Chapter 8 seeks to achieve RO4 by presenting *an approach for provision of relevant information* about modular systems during the transition process. This *IT-integration approach* can be seen as direct support for the derivation of modularisation metrics, the modularisation assessment framework and a vital contributor for overall transition toward modular system development.

The deliverables of this thesis in the context of modular system development are shown in Figure 44.

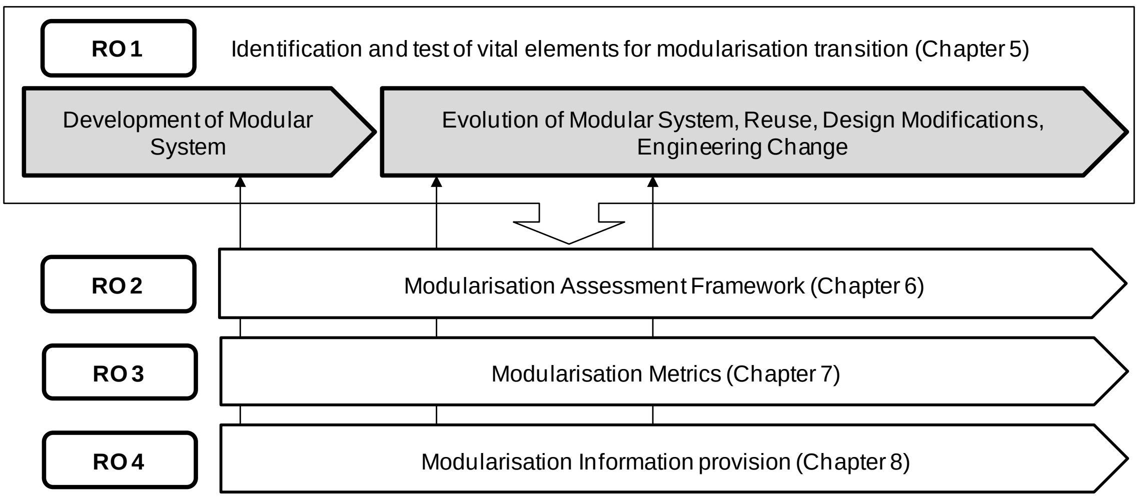


Figure 44: Relation between research objectives and deliverables of this research work

5 Modularisation transition – a longitudinal case study in industry

As described in the previous chapters, to date it is unclear how companies can be supported in making the transition toward modular system development within existing products. The review of literature about modularisation support in Chapter 3 indicates that contemporary research in the field mainly focuses on the concept phase of the engineering design process without giving answers how these concepts can be transferred into a working modular system. In order to close this gap, more knowledge about modularisation transition is needed. Thus, this thesis chapter will present an overview how companies can make the transition toward modular system development.

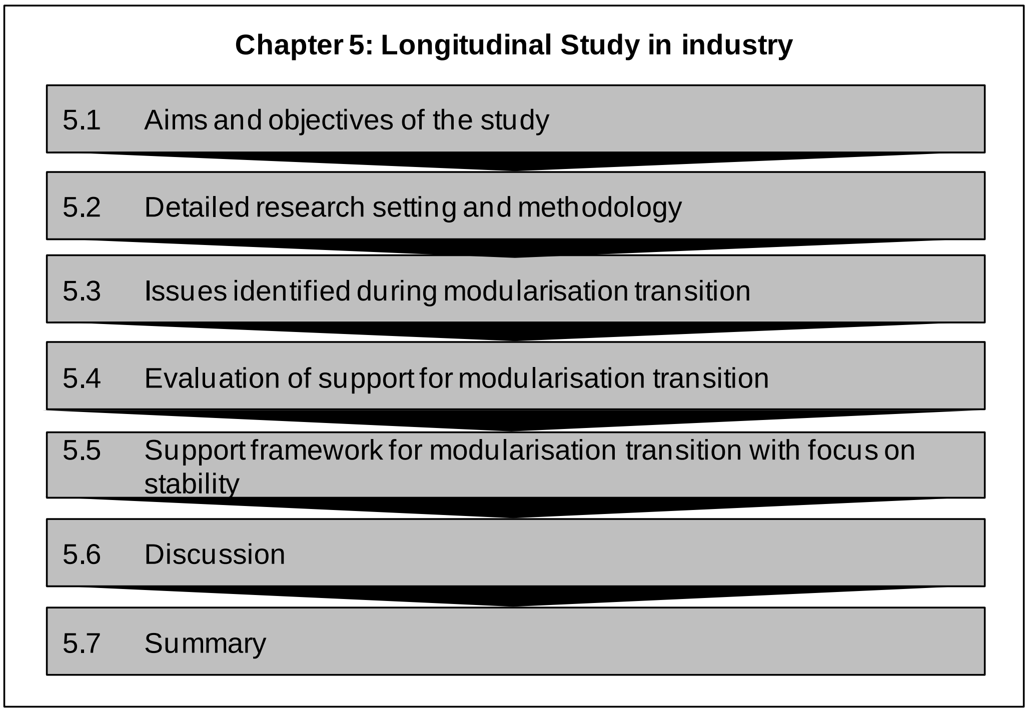


Figure 45: Elements of Chapter 5

Before presenting the results of the industrial case study in order to answer research question RQ1, a brief overview of the remaining sections of this chapter is provided. Section 5.1 gives the aims and objectives of the study. The detailed research setting and methodology is given in Section 5.2. Issues that were identified during modularisation transition are presented in Section 5.3. Use and limits of support for modularisation transition have been evaluated in Section 5.4. Based on these findings, a support framework for modularisation transition with stable product architectures is presented in Section 5.5. The results of this chapter are discussed in Section 5.6 and Section 5.7 summarises this chapter. Figure 45 summarises the elements of this chapter.

5.1 Aims and objectives of the study

The central question of this research is how companies can be supported in transitioning toward modular system development. Thus, as outlined in Chapter 1 and Chapter 4, it is the first research objective of this thesis, RO1, to identify vital elements for modularisation transition. RO1 can be further subdivided into objecting following research deliverables:

- Identified issues during modularisation transition
- Use and limits of support for modularisation transition
- Well-reasoned starting-point for research about support for modularisation transition

With particular respect to these points, this section also briefly reviews literature that supports achieving RO1.

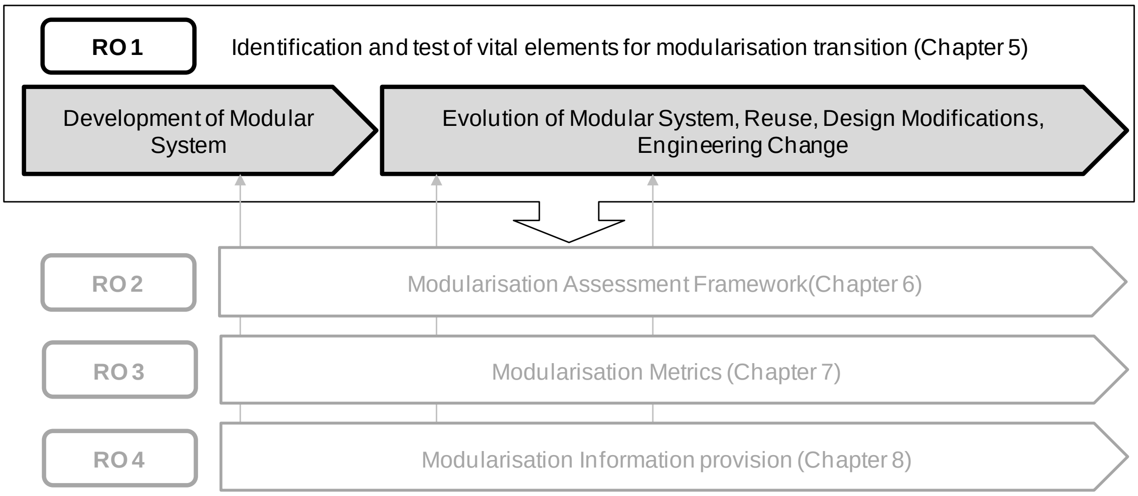


Figure 46: Relation of this chapter to overall context of this work

Besides the modularisation support that was presented in Chapter 3 as a foundation for this research thesis, there are several particular studies available that are of particular interest for the purpose of this chapter, three are discussed in detail below.

- Challenges in platform design from case studies in industry

Chao and Ishii (2004) report about the constant threat that a platform project is "killed" before its completion. The threat comprises synchronisation problems between development teams, high initial investments, lack of resources, unstable market requirements and lack of vision and commitment. Nobelius and Sundgren (2002) studied six manufacturers from different industries on managerial issues in parts sharing between different development projects. The identified issues are of organisational, technological, strategic, cost related and support system related nature. Gudmundsson et al. (2004) studied two standardisation projects in industry with the finding that one project failed and the other project did not have any positive effects on the organisation. Some of the major root causes for this failure were the lack of time, resources, motivation and stringent implementation programme. Persson and Åhlström (2006) analysed modularisation-specific platform development at Volvo Car Corporation. The researchers conclude that managers have to deal with the appropriate degree of modularity, balance the trade-off between requirements of different modularisation stakeholders and the coordination of processes within a modular organisation. Skold and Karlsson (2007) identified three multimanagerial challenges during multibranded platform development. The first challenge deals with the commonality of architectural elements, the second challenge deals with brand differentiation and the third challenge is based on a corporate management level and deals with the trade-off between commonality and brand differentiation. Arnoscht (2011, p. 184) points out that problems encountered during modular system development arise from lack of change management, lack of knowledge, lack of leadership and missing basic conditions (e.g. overly ambitious expectations).

- Implementing platforms in industry

Muffatto (1999) and Muffatto & Roveda (2000) report about implications of platform implementation on development process management and organisational settings. Shibata and Kodama (2013) come to similar implications at Mabuchi Motor Company. Kraus (2005, p. 148) addresses processes, products, organisation, managers, employees and information as important aspects to consider during platform implementation. Karandikar and Nidamarthi (2007) developed a framework to implement a platform strategy at ABB automation. The framework is built around the PDCA-cycle and touches the areas of platform components, change management, work processes and enabling IT technology. Ponn (2015) identified three main points of action for platform development at Hilti Entwicklungsgesellschaft mbH: a) creating transparency and visualisation of correlations in the portfolio, b) promoting synchronisation between product and module development and c) supporting decision making by managing the conflicts of interests between the project and the portfolio perspective.

- Implementing product architectures in industry

In a case study with six manufacturers, Lau (2011) identifies seven success factors for managing modular architectures in *production* design: identified product advantage, designated design rules, definition of modules, system integration, technological maturity, internal communication and stakeholder involvement. Lalande (2013) addresses process, measurement and organisational issues during vehicle modularisation by introducing four critical success factors: reference architecture, generic product definition, execution model and a commodity framework. Kreimeyer (2014) gives insights into the work of the central product architecture department at MAN Truck & Bus AG and presents organisational changes that have to be done when implementing architecture in industry. Collaborating with a platform research group at the Technical University of Denmark (DTU), Nielsen (2010) studied the ongoing process of developing modular platforms and updating existing ones at LEGO and Grundfos. He clearly divides platform and product development while stressing the need for aligning both processes. Therefore, platform development is split into a design and a maintenance organisation. Finally, Nielsen (2010, p. 145–146) calls for more research about the transition period, implementation and maintaining platforms. In a parallel study at DTU, Munk (2011) concludes that "between half and two thirds of the platforms do however do not achieve the expected effects, despite that they do deliver some effects". This is reported mainly due to lack of using platform elements (Munk, 2011, p. 169).

5.2 Detailed research setting and methodology

In order to avoid "reinventing the wheel", the qualitative study which is described in this chapter starts where other researchers seem to have stopped:

- The methods described in Chapter 3 were only very briefly, if at all, validated in industry. Consequently, this research focuses its studies on two methods from the modularisation support from Chapter 3 in some detail. In addition, it looks at other modularisation support methods in less detail. As a result, this chapter will deal with application of modularisation methods in product development projects in industry over a prolonged period.
- Based on first initial studies in other adjacent fields (see second part of Section 5.1), the longitudinal study researches modularisation transition activities in primary and secondary case companies in order to present the basic rationale to develop support for modularisation transition.

The next paragraphs will now describe how data was collected and analysed for this study.

5.2.1 Data collection

As was pointed out in the description of the overall research methodology in Chapter 4, data was collected in a primary case company (primary cases) that was studied constantly during a prolonged period and from secondary cases that were consulted on a "need-based frequency". Following the guidelines for descriptive study methods of DRM (Blessing and Chakrabarti, 2009) and the research methodological examples of Nobelius & Sundgren (2002), Lau (2011) and Hales (1987), following Table 9 has been created to characterise and provide the theoretical underpinning or research structure of the qualitative study of this work.

Table 9: Characterising qualitative study for modularisation transition

Characteristic	Description
Aim, Research questions	Better understanding of transitioning toward modularisation in industry. Questions concerning influencing factors on the transition project and newly developed support for such a transition have to be answered.
Nature of the study	This is an in-depth case study in order to find out what was going on in the case projects and to understand how the transition can be made. In addition to the case study, numerous benchmark partners, i.e. secondary cases, were analysed in order to get a broader, deeper and more generalisable understanding. For characterisation of cases, see Appendix B.
Theoretical basis	- systematic engineering design process with its interfaces to adjacent areas - literature on modularisation, product platforms, product architectures and product families with focus on support and case studies
Unit of analysis	Company-wide transition from single product development toward product development based on modular systems.
Data collection and recording	Participant observation and action-based interventionist approach using research notes, research logbook, meeting notes, survey data (e.g. semi-structured interviews, questionnaires), company documents, data derived from IT-Systems and experiments
Role of researcher	The researcher was research engineer for modular product development, participant in the overall transition project, observer in the overall transition project (with central view on everything that is going on) and supporter of the primary and secondary cases.
Duration	The time on site in the primary case company covers more than 34 months full-time from Nov. 2010 to Sep. 2013 (> 5600 h on site in the primary case company solely for data collection without analysis or theory building), followed by a period of detailed analysis, further development of support and the involvement of younger cohorts for triangulation of research findings which lasted approx. 27 months, part-time until 2015.
Continuation	The work on the project for data collection was during the main engagement full-time 5 days per week (> 45h/week in average) followed by a part-time phase for data analysis and refinement (approximately 15-20 h/week in average).

Characteristic	Description
Time constraint	Time constraints were set by the company in terms of funding issues. However, data could be collected from even before the study started (modularisation activities started in Jan 2010) and after the engagement ended (delivering and discussing results, interviews, discussions, observation, action research, joint publications, retrieving information) until 2015
Observed process	All processes and adjacent processes that were related to the transition toward modular system development. In terms of time, this means that the research starts with initial modularisation activities and commences until the design of first concept appliances and pilot project with the first launch of modular products.
Setting	The major part of the study took place in a large company as primary case in an industrial environment (no labroratory). To complement the picture, other industrial partners were consulted as secondary cases (see Appendix B for characterisation)
Task	The research task and problem is real, relevant and derived from industry as well as literature. The research project was one of the major strategic projects of the primary case company.
Number of cases	One in-depth case study of the primary case company, comprising one modularisation transition project in the central department and two projects in development projects for modular systems. In addition, 27 secondary cases were analysed.
Case size	The primary case project affected the whole company at different sites and in different functional departments.
Participants	Mainly engineers and engineering managers (from HW, SW and Systems Engineering), but general managers and other company functions like manufacturing, purchasing, controlling, product management and sales were also involved where appropriate.
Object	The project involved the whole range of heating, ventilation and air conditioning systems that is needed for complete heating or cooling water and air in households or in industry. Moreover, by collaborating with other companies insights from other products such as automotive equipment, household equipment, heavy machines or power tools were studied.
Characterization of cases and products	- primary case: full HVAC (see Sections 1.8, 4 and Appendix B) - secondary cases: (see Sections 1.8, 4 and Appendix B)

Characteristic	Description
Coding and Analysis method	Notes were collected continuously. During the analysis phase of the research, the notes and the documents were coded based on the findings that can be related to a certain research question. For data analysis, this was done in a database which allows deriving cause-effect relationships as well as quantitative and qualitative findings.
Validation method	- iterative steps, mainly with evaluation reviews by interviews, workshops, presentations and meetings to verify subsequent steps - expert opinion (e.g. by consultants, experienced engineers, managers or other researchers) - comparison with literature and logical reasoning - implementation and application - validation against requirements and specifications - validation with the help of "modularisation users"
Notes	It must be noted that the relation between cases and research questions is of an intricate n:n relation and explorative which makes data analysis more complex than laboratory research or case studies with a single case. Consequently, a result of a single case study within this work is not exactly the same as the answer for a research question.

Several primary and secondary cases from industry were studied during the overall qualitative study. Each of these cases has been further characterised (see Appendix B). There was one central primary case company with a study in the central engineering department, dealing mainly with gas boilers, a study in a heat pump modularisation project and a further study in a modularisation project for stoves and heating inserts. In addition to these three primary cases, 27 secondary cases were studied. The secondary cases are from different industries and had following purpose:

- Mature cohorts: Mature cohorts were used as benchmark partners in order to obtain expertise about modularisation transition for the primary case company.
- Mixed cohorts: These cohorts were approached on a need-based frequency in order to answer specific questions and to refine and validate findings throughout the whole study.
- Young cohorts: This category of cohort was used mainly to refine findings, for triangulation and for validation of research findings.

A more detailed characterisation of the cases for the qualitative analysis can be found in Sections 1.8, 4 and Appendix B where each case is listed and classified. There it is also defined what kind of research method had been applied in detail to collect data for the purpose of this chapter.

Having defined the framework for data collection, the next paragraph will deal with the analysis of the collected data.

5.2.2 Data analysis

The data collected in the field is mainly presented on notes of the researcher and on formal and informal documents. These notes and documents were processed systematically by considering research guidelines about qualitative data analysis and coding of large unstructured sets of field data (Gibbs, 2007; Mack et al., 2005; Saldana, 2013). The five steps are listed below.

1. Finding categories and codes

Categories as a special type of code (Gibbs, 2007, p. 39) are used to classify the content of field data in order to have the possibility to link those contents later on. The categories identified for this research work are given in Figure 47. In order to identify relevant aspects and categories for modularisation transition, a preliminary study that linked theory and industrial practice of modularisation was used (Heilemann et al., 2012). The identified categories are as follows:

a) tbl_life_cycle_phase: the life cycle phase to which a certain issue or support can be assigned to (e.g. requirements phase, logistics, product use), this category also has been applied for the classification of modularisation methods from theory in Section 3.2.2.

b) tbl_aspect_classification: the aspect that a certain issue or support concerned (e.g. process, organisation, IT, evaluation, implementation of modularisation)

c) tbl_sorting_classification: the broader context a certain modularisation topic could also be related to (e.g. change management issues, standardisation, financial considerations)

d) tbl_company_function: the company function a certain content is relevant to (e.g. top management, product management, product development, purchasing)

e) tbl_scope_classification: it is important to differentiate whether a certain content is related to a single product/project focus or to a broad modular system scope

f) tbl_modular_system_phase: especially for support with a broader modular system scope, it was interesting to see to which phase a certain topic could be related to (e.g. planning modular system or maintaining modular system)

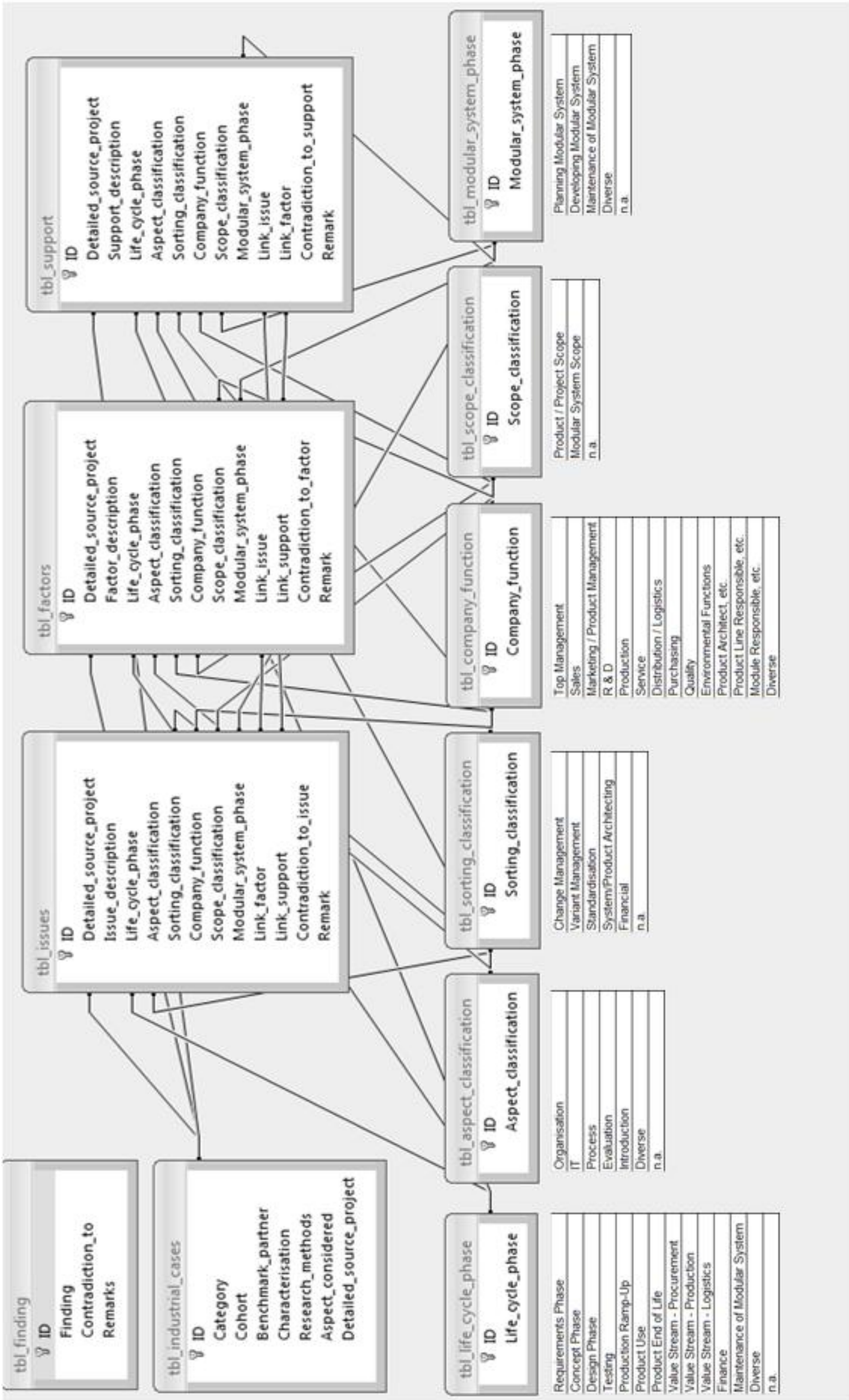


Figure 47: Screenshot of relational database with tables representing data sets, codes and their links amongst each other

2. Transferring field data into a coding database

During this step, all field notes and documents were scrutinised for potential issues (tbl_issues), important factors (tbl_factors) and support (tbl_support) for modularisation transition. After identification, the relevant content was indexed and transferred into a coding database.

3. Coding data sets in the database

The relational coding database was set up in MS Access. The database was built around the principles of NVivo software for qualitative research, i. e. being able to establish links between content and codes. This setup proved to be helpful in identifying common themes, contradictions (e.g. Contradiction_to) and concepts while still being able to trace each content back to its source (tbl_industrial case).

4. Analysing coded data

The database provides the functionality to export coded data sets into MS Excel for further analysis. For instance, this could be used to visualise the rather abstract coded data. Such visualisations served as input for the next step.

5. Building concepts

Given direct experience from the field, documents from the mixed method approach and categorised and visualised information from the coding database was used to establish concepts about modularisation transition, in particular, to achieve research objective RO1. For instance, issues of a certain category could be further grouped into issue clusters in order to identify recurring themes across different cases.

A compressed view on tables, fields and relations in the coding database is provided by Figure 47.

5.3 Issues identified during modularisation transition

The qualitative study using the setting and methodology of Section 5.2 revealed 166 issues in the primary case company and in secondary cases during modularisation transition. These issues are further analysed and presented within this section. A detailed list of issues is presented in Appendix D.

After initial explorative analyses and experiments, it was decided to build these issues around coding category a) "life cycle phases" (see Section 5.2) for the further course of this work. Thus, the categorisation of the issues can be directly compared with the identical categories applied for modularisation support from literature in Section 3.2.2.

Figure 48 shows the distribution of the identified 166 issues along different phases of the complete life cycle, including value stream activities like production and logistics. It can be seen from the same figure that the majority of issues is related to "diverse" phases of the life cycle.

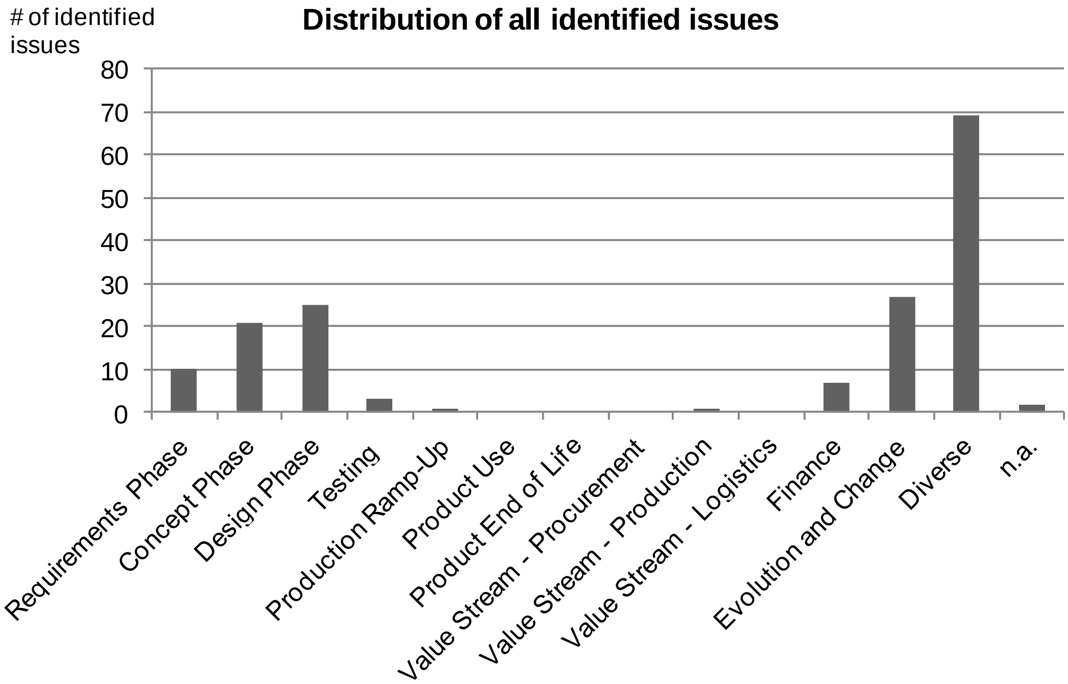


Figure 48: Distribution of all identified issues during modularisation transition

As the emphasis of this work is on engineering design processes, Figure 49 shows a more detailed excerpt with relevant development life cycle related activities for modularisation transition. Figure 49 shows that the requirements, concept, design and evolution & change[9] phase are the most problematic phases for modularisation transition from an engineering design perspective, provided that issues which can be assigned to diverse phases are considered separately.

The next sections will go through each *development* life cycle phase from requirements phase towards evolution & change of the modular system (see the phases of Figure 49) and analyse what the issues are and where they may arise in transitioning companies. In addition to the investigation of development life cycle phases, issues with "diverse" life cycle phases from Figure 48, that may concern implementation issues or the like, have also been investigated and are presented afterwards.

Although the diagrams of Figure 48 and Figure 49 are of quantitative representation, they have been interpreted in a qualitative way. Therefore, following cautionary notes have to be considered:

- It is the purpose of Figure 48, Figure 49 and the interpretations thereof to generate an overview of database results and to serve as starting point for concept building.
- It is not the purpose of Figure 48 and Figure 49 to provide detailed quantitative or numerical analyses.

[9] The evolution and change phase is a phase that occurs in situations of transition as there may be a number of intermediate phases before full modularisation is achieved.

- The “severity” of issues cannot be compared by their quantitative occurrence given in Figure 48 and Figure 49.
- The results of the qualitative analysis have to be considered in relation to the characteristics of the case study they were derived from.
- For concept building, the 166 issues from the database have been thematically combined in order to come up with compressed and manageable chunks of issues (see Appendix D). Thus, the presentation of issues in the following sections will solely be of qualitative and descriptive nature.

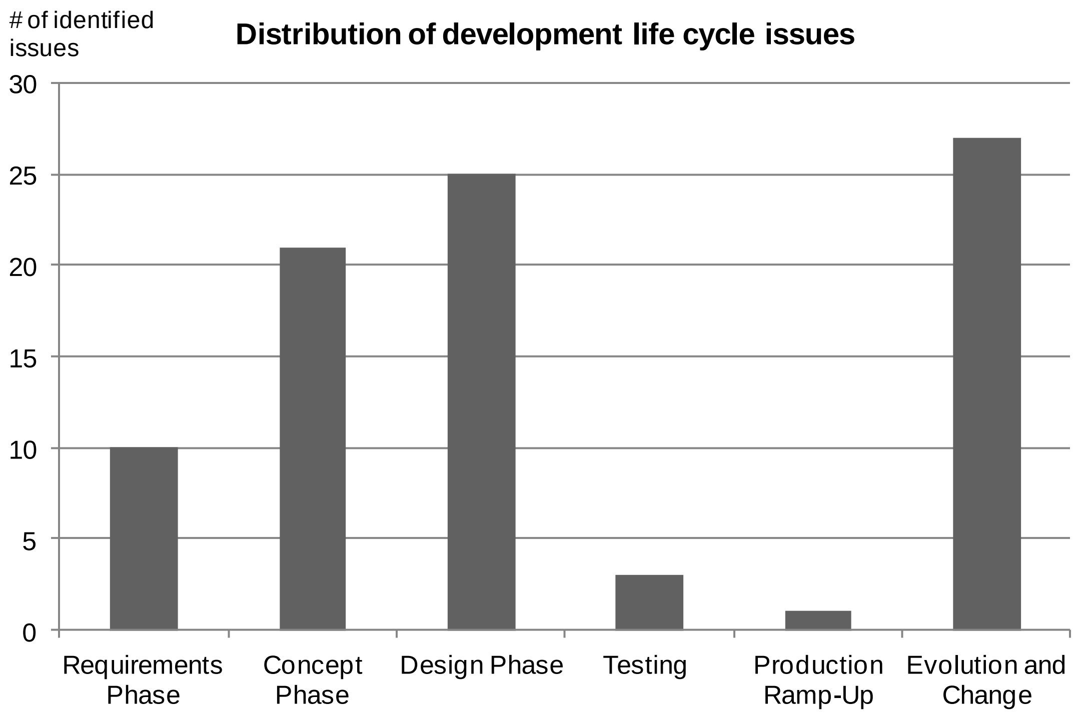


Figure 49: Distribution of development life cycle related issues during modularisation transition

5.3.1 Issues during development life cycle phases

- Issues during requirements phase

The issues that were encountered during modularisation transition in the requirements phase are manifold. The studied case companies started modularisation transition in this phase, assuming that either a series of workshops on modularisation or applying an extended stage-gate process is sufficient to define requirements for the newly created modular system with extended scope. For this reason, product managers and engineers had to handle much more requirements for parallel variant and future products. In addition, as the transition had to be done within existing products, the involved roles had to quickly bring first products from the modular system onto the market.

Compared to single product development, engineers had to deal with following challenges:

- Much more requirements to be derived, analysed and fixed
- Many yet unknown and uncertain requirements due to planning products to be derived from the modular system several development life cycles ahead
- Extreme pressure to quickly launch first product variants derived from the modular system onto the market
- The succeeding development process phase requires a stable and fixed base of requirements in order to create a common modular reference architecture for a broad product portfolio

However, the project teams that were used to single product development settings had extreme difficulties to master the challenges from above for several reasons.

Firstly, although some additional time was allocated to the project for extended requirements engineering, the project teams simply had not enough time and resources to master these challenges.

Secondly, there was the constant struggle between the need to fix requirements for the next development phase and the inability to fix requirements in time due to lack of knowledge and transparency. On the one hand, it was not possible to leave too many requirements open at this stage, because this would have been a poor input for the design of a stable product architecture. On the other hand, fixing requirements just so they are fixed is like a gamble that could lead, in turn, to many changes and an unstable architecture later on. Moreover, project teams felt that in order to make stable and fixed decisions about requirements, additional market analyses would be required. However, there was not enough time and resources to run another series of market analyses, key user experiences and the like.

These issues threatened the stability of the yet to be developed modular system in two different ways: a) constant forces or tensions that try to pull the project back to a single product development approach with narrow scope of requirements (e.g. statements like “at least for the products for the next trade fair”) and b) constant forces that plead for proceeding with a large amount of unstable requirements that are already expected to be updated, added or deleted during later stages. Moreover, there were also cases where too much unplanned time was spent on this phase with the result that later phases were deprived of resources.

- Issues during concept phase

This is the phase where the product architecture is suggested to be established. Compared to single product development, where it is not necessarily required that a *formal* reference architecture is established, modularisation transition requires formal architecting and explicit description. All companies under study made their architectures explicit during this step. However, the way how companies established their product architectures varied vastly. The first controversy encountered was whether a formal method for product architecting is needed or not. Some companies solely relied on engineering expertise without method whereas other companies rather tried to apply a supportive modularisation method. Surprisingly, no matter which approach had been taken, the concrete step of establishing the architecture was not related to main issues of this phase.

The *first issue cluster* that all companies encountered affects the major task of this phase. It is the goal of the concept phase to convert requirements into stable specifications of the architecture and of the product portfolio. These specifications are the input for the succeeding phases embodiment and detailed design (design phase). During single product development, the scope of the specifications to be created is quite straightforward and manageable. However, during modularisation transition the scope and required detail level gets largely extended. Firstly, the specifications have to cover much more parallel and future needs. Secondly, the specifications have to be more detailed than in single product development. This is because the specifications have to be passed on to module developers that have less knowledge about the details of the overall architecture and portfolio. As a consequence, the resulting issue was that engineers did not have enough time and knowledge to create detailed specifications for the whole modular system.

The *second issue cluster* was that the input to this phase is technically solution neutral and the output of this phase should ideally have a fixed decision on what technical solutions to implement with the modular system. For instance, it is required to define interfaces and space requirements for modules. However, as the modular system has to be developed anticipating several years ahead, engineers had lots of difficulties to come up with well-reasoned decisions that are necessary for further realisation of the modular system. This technical knowledge could have been built up with further studies, concept appliances, simulations or experiments, but there was not enough time allocated to this phase.

Beyond difficulties to decide for the right technical solution, there were many time consuming discussions during this phase about the appropriate level of modularity, granularity of the modules, standardisation degree of modules and size ranges of modules. The trade-off between the statement that these discussions were actually very fruitful and the statement that there is no time for these discussions due to market pressure has been present everywhere.

Moreover, there are two forces during this phase that pull the conceptual modular reference architecture towards a more local integral architecture: Firstly, project team members previously used to work on local architectures so that they might not be sufficiently knowledgeable and motivated to work on a common modular reference architecture. Secondly, in order to rule out different concepts, comparisons of products are mostly done on single product level. In such cases, direct costs of optimised integral single products are of advantage compared to direct cost of common modular products. Cost savings of modular products through, for instance, synergy effects are not considered by these approaches. Thus, local architectures have been mainly favoured over global architectures.

Modular reference architectures have been set up with either engineering expertise or with methods, were quite straightforward to determine or some cases have been "already given". Those companies that set up the product architecture with the help of a method encountered following problems during this phase:

- Engineers felt that they spent too much time on filing matrices or graphs while losing the touch to the overall picture of modularisation. For instance, 20 engineers discussed which column to mark with a certain cross in a matrix.
- After filing matrices, graphs and running through algorithms, the resulting benefit and value of that work did not seem to outweigh the effort behind this task. The outcome of the architecture with a certain method seemed either quite obvious or required massive rework by experienced engineers later on.

The issue behind these points is that the study did not reveal evidence if a product architecture method helps to establish better product architectures. There are some indications that methods might be supportive, but there are also facts that show that the benefits of a modularisation method might not always outweigh its resource consumption. This is particularly critical as architecting itself was not seen as main issue during this phase within existing products. Due to its criticality and relevancy in literature, modularisation methods will be further scrutinised in section 5.4. It can be concluded that the main issue of this phase is not "finding" the right architecture, but finding enough time, resources, motivation and knowledge to define an extended modular reference architecture.

- Issues during design phase

Given the conceptual specifications for the modular system of the previous phase, the design phase concerns the embodiment and detailed design of the modules. The output of this phase are detailed drawings that are ready to be handed over to production. The specific feature of transitioning within existing products is that engineers have to manage the trade-off between drawings for the release of the first products and still considering the overall modular system for future products. However, that undertaking was prone to failure due to several reasons.

Firstly, engineers worked with full attention on the release of first products. They only considered design details of future and parallel products where it seemed "suitable" for them. During this phase, engineers neither had time, resources nor any motivating intrinsic or extrinsic incentives to do additional work for the overall modular system that was beyond the scope of the current project. Thus, there were no measures in place to stringently pursue the idea of the overall modular system. In contrary, incentives rather favoured pursuing single project objectives than global company objectives.

Secondly, the specifications from concept phase still contained some fuzzy elements that could not be further detailed during embodiment and detailed design. In particular, this was the case for designing modules and interfaces to neighbouring modules which will be detailed only in future projects. In order to properly design products, there was still lack of information and diverse uncertainties in all areas. Even worse, already during this phase, product management started to question and change a bundle of requirements that should actually have been fixed during earlier stages. In consequence, engineers suffered lack of knowledge for detailed design documents, especially for those that concern the interplay with modules that are needed for the overall modular system, but not developed in detail during current projects.

Another point is that the input to this phase is done in a rather informal way through spreadsheets, text files or drawings. The qualitative study revealed that this information is seen between proceeding and succeeding stages as more informative than as binding! Hence, a high amount of uncontrolled deviations from the original product architecture specification could be found during this stage.

These issues resulted in the situation that after design phase, projects rather seemed to fall back into single product development behaviour than considering the overall modular system to be developed. For instance, this was evident in a case company where the plans of the modular system from concept phase were compared with actual PDM data of the first products to be "derived" from the modular system. The two data sources did not match at all.

- Issues during testing phase

During this phase, it was still not clear whether the modular system will work or not. The focus of engineers during this phase was on testing functional characteristics of the first products to be launched from the modular system. However, what could not be tested was the underlying concept of the modular system. This means that only a small percentage of the planned modules were developed at this stage. Thus, it could not be verified if the modules really work together and if planned interchangeability can be achieved later on. The resulting lack of knowledge could have been remedied with additional tests, but there was no time left for such activities beyond the scope of the first products to be launched.

- Issues during production ramp-up

One issue that is directly related to modularisation was identified during this phase. Several cases were identified where the modular structure of engineering design differed from the modular structure of production or service. The resulting different view on modularity is less a problem of early involvement of manufacturing or service, but in fact a problem of handling different structures in different IT-systems throughout the company. Moreover, it gets more difficult to communicate modules if different company functions have different views on a certain module. In sum, this could lead to a lack of knowledge or transparency.

- Issues during evolution and change phase of the modular system

Several major issues were identified during this phase that deals with the maintenance of the product architecture, which is the evolution, reuse, design modification and engineering change of the modular system and its derivative products. The qualitative study revealed that this is the most dangerous phase where the product architecture diverges and the modular system loses its effectiveness.

A constant conflict of interest and trade-off between global goals of modularisation and local goals of derivative product development has been identified. Different product development projects that are actually supposed to derive products from the same modular system and sticking to the specifications of the common modular reference architecture have been frequently inclined to chase single project goals and losing sight of the big modular picture.

The following issues can be seen as the reason why forces that are pulling projects toward the common modular reference architecture are too weak:

- Lack of intrinsic motivation: For engineers it seemed to be easier to generate new module variants than reusing module variants from the modular system. Moreover, it was easier for engineers to deviate from the common modular reference architecture and creating isolated tailored solutions. In addition, engineers literally stuck to their used single product development behaviour. From an emotional point of view, local engineers and managers feared loss of freedom and loss of expertise as some parts of their engineering design activities were prescribed by central modular system development.
- Lack of extrinsic motivation: It was revealed that there are no formal consequences if engineers built a “workaround”, undermining the common modular reference architecture. This might either be a result that there was no one in the company in charge of checking the compliance of artefacts with the modular system in due time. On the other hand, this could have been also a result of traditional evaluation of product development: Usually, projects and products are assessed based on their local goals and not based on their global goals.
- Lack of time and resources: During the transition period, resources were allocated to fulfil the market needs of proximate products. Therefore, engineers developed local solutions to their specific needs on hand instead of sticking to architecture specifications. However, modularisation transition means that designated modules meet the needs of both, proximate and non-proximate products. There was paucity of time and resources to develop the modular system for both needs.
- Lack of collaboration and communication: The transition requires to exchange much more information about requirements, specifications, modules, interfaces and products than within single product development. Though, communication channels were still directed toward the traditional way of working. This made it more difficult for engineers to break up with their old behaviour and communicate site-, country- and market-overarching.
- Lack of knowledge and transparency: With modularisation transition, the scope of products and modules extends vastly which leads to higher complexity engineers have to cope with. For instance, in single projects engineers had a lack of informa-

tion about the overall modular system of what is available, what architectural plans are and to which specifications to stick to.

The following practical implications show how difficult it is to keep the modular system approach during this phase:

Different local sourcing strategies and different level of in-house production depth for the same parts of the product architecture made it difficult to achieve architectural commonality across sites. The same hindering forces for stability were given if rationalisation projects were done on product level. If a local manufacturing site decides to rationalise its single product, e.g. based on DFMA principles or based on best in class products, the outcome might be negative for the site-overarching modular system.

Some of these issues have been already indicated throughout preceding phases. However, their magnitude and effects increased during this phase.

In the section that now follows, those issues that cannot be directly related to single development life cycle phases will be shown.

5.3.2 Issues during diverse life cycle phases

As explained earlier, the described issues lack of intrinsic motivation, lack of extrinsic motivation, lack of time, lack of resources, lack of collaboration, lack of communication, lack of knowledge and lack of transparency have some phase-overarching elements and are also valid for several life cycle phases. This section now moves on to explain issues that are valid for several development life cycle phases.

- Decision making

In order to avoid an unreasonable high amount of iteration loops, it is necessary to fix decisions during modularisation transition. While the effect of constant changes during single product development remains manageable, the effect for changing large modular systems becomes unclear and intricate. However, the qualitative study revealed that, in fact, engineers and managers are highly troubled to fix decisions for the broad modular system. In the end, modularisation transition seems to be like a big gamble or like bedding into the long-term future. Therefore, it gets all the more important to remove uncertainties by providing support for decision makers through a sound base of knowledge and dedicated time and resources for modularisation transition.

- High initial investments and only indirect but delayed benefits

The qualitative study found out that there is need to scale down expectations about benefits of modularisation during transitioning. This was for several reasons:

- Modular system development required high initial investments while the benefits of developing a modular system could not be harvested for at least one development life cycle. Thus, the promised increase in profitability was not related to the present, but to the long-term future. The principle behind this issue is sketched in

Figure 50 (real data behind the scheme has been collected during cost-benefit analyses in the primary case company, but due to confidentiality reasons the graph has been disguised) and was also identified by Blackenfelt (2001, p. 15–16). However, engineering controlling systems in companies were rather adjusted to short-term developments than to sustainable long-term developments. The resulting discrepancy between expected benefits and actual cost led to an atmosphere that was adverse to modularisation transition.

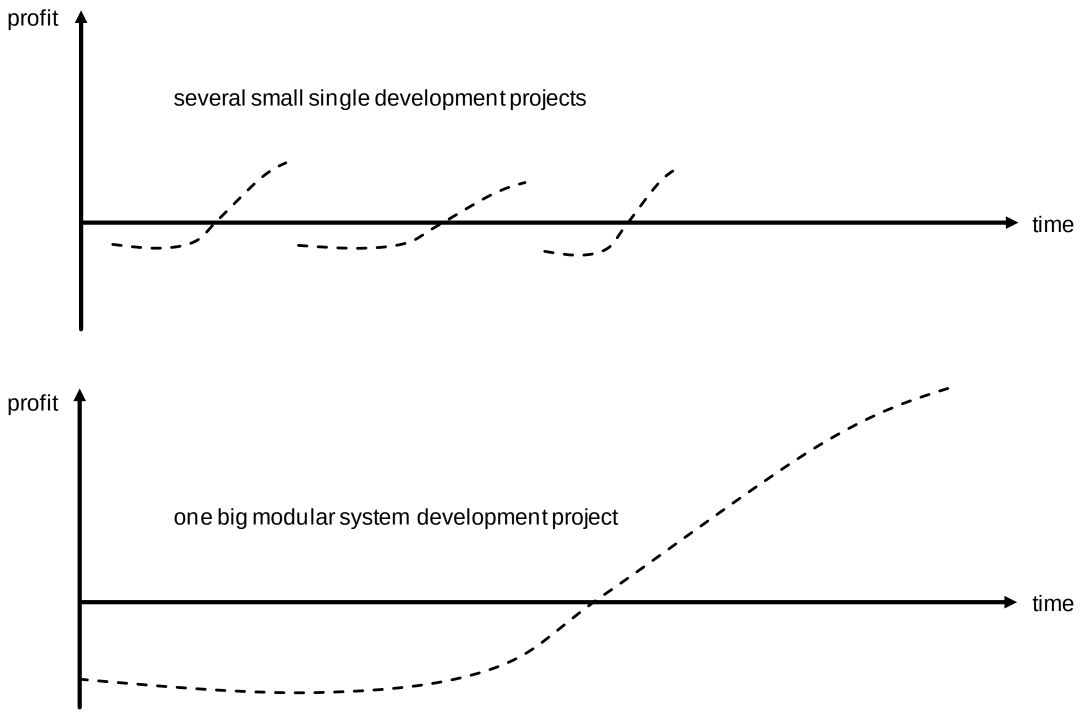


Figure 50: Delayed realisation of benefits in modular system development projects (see lower part of figure) compared to single product development projects (see upper part of figure), schematic representation

- During development projects, short-term goals and goals of single products have always been prioritised compared to the global goals of modularisation. For instance, oversizing modules for multi-purpose applications causes frustration in single product development projects due to lack of direct benefits.
- It is one of the major goals of modular system development to reduce complexity. This complexity reduction was often measured as part number count within case companies. However, the qualitative study showed that the part number count increased over time, with modularisation and without modularisation. Figure 51 shows a simple, and schematic sketch concerning this issue. Usually it is assumed that the part number count increases less dramatically with modularisation than without modularisation. Even worse, some cases were identified where the part number count rose initially faster than without modularisation. This phenomenon during transition was also observed by Nielsen (2010, p. 136). In practice, it is not

possible to directly compare a project with modularisation to exactly the same project without modularisation. The fact that it is unclear how to directly show the benefits of modularisation during transitioning may leave a considerable amount of employees doubtful.

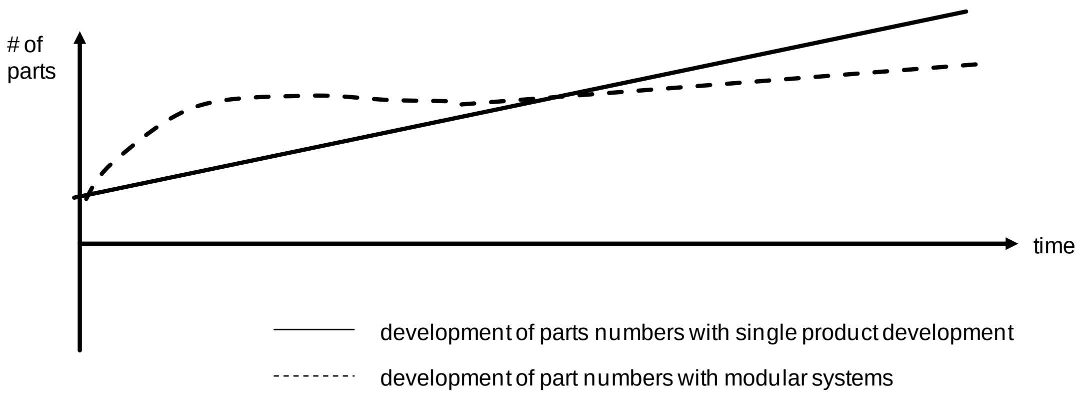


Figure 51: Simplified development of the part number count with and without modularisation, based on the qualitative study, schematic representation based on Nielsen (2010, p. 136)

- Synchronisation between different development projects

Another issue encountered during the study was that there is a lack of synchronisation between those projects developing modules for the modular system and those projects that consume generic modules. For instance, a case was identified where a product development project relied on a generic module from another project, but in the end the project had to develop its own project-specific module. If an overarching module is not fully developed exactly at the point when the product development project needs it, local projects are inclined to go their own way while blaming the modular system for its troubles. This example shows that there has been a lack of overall framework across projects for synchronisation of modular system development.

5.3.3 Summary of issues

The qualitative study has identified a number of issues that primary and secondary cases encountered during modularisation transition. The issues emerged during different development life cycle phases. The requirements phase, the concept phase, the design phase and the maintenance phase of the modular product architecture (i.e. evolution and change of modular system) have been identified as the most critical phases. In summary, following clusters of issues have been identified during the study:

- High initial investments, only indirect and delayed benefits
- Difficulties in decision making during modularisation transition (e.g. through uncertainties, trade-offs and high impact of decisions)

- Lack of intrinsic motivation to pursue the "overall modular system picture"
- Lack of extrinsic motivation to pursue the "overall modular system picture"
- Lack of dedicated time and resources to pursue the "overall modular system picture"
- Lack of knowledge and transparency about the "overall modular system picture"
- Synchronisation problems between projects (i.e. derivative product and modular system) and lack of collaboration and communication

It has to be considered that the clusters of issues might have dependencies among each other. A detailed list of issues from the research database can be found in Appendix D.

During front-end of the development life cycle, the main difficulty is rather on establishing and committing upon requirements and technical specifications that can be used as input for developing the whole future modular system. During later phases of the development life cycle, the main problem is rather on keeping the modular system stable. This means that product development projects develop in compliance with specifications of the modular system and that architecture specifications are constantly controlled and eventually aligned to changing circumstances. However, the study revealed that exactly this has not been happening due to the presented issues. As a result, projects fall back to single product development mode over time due to lack of support mechanisms that remove above mentioned issues. An overview of the issues and how they are related to the modular system life cycle is shown in Figure 55.

In addition to the identified issues, it is important to analyse the use and limits of existing modularisation support and why it failed to solve above mentioned issues. This will be treated in the next section.

5.4 Evaluation of support for modularisation transition

This section presents the evaluation of modularisation transition support which has been conducted during the longitudinal case study. The support of this section is based on the literature review of this thesis in Section 2.6 (development life cycle process models) and Section 3.1 (modularisation methods). Firstly, the usefulness of existing development life cycle models will be investigated in Section 5.4.1. Secondly, findings of the qualitative analysis will be used to analyse some modularisation methods from literature in Section 5.4.2.

5.4.1 Development life cycle models

The literature review in Section 2.6 brought modularisation into the broader context of product development processes. Modularisation has either been seen as the step in new product development which divides the product into modules or as a phase-overarching activity that also prepares engineers for modularisation with appropriate information.

The qualitative study revealed that the development process models like stage-gate models that are used in industry were quite close to that what could be found in literature. However, the models from industry possessed a higher detail level. In sum, three different streams how current process models could assist modularisation transition have been identified in industry. In the course of a project called "modularisation process integration" which was conducted during the longitudinal case study, these different process models were evaluated with engineers, engineering managers and even top management of the primary case company. It was the aim of this evaluation to identify an appropriate process model for modularisation transition to be implemented in the primary case company. Detailed results for each identified model which have been documented throughout the study are given below.

- Classical new product development approach with focus on modularisation

Modularisation can be integrated into the classical new product development process that is well known from literature and industry. Compared to the development of a small range of products, the development of a broad range of products based on modular architectures does not require an adaption of the standard product development process. Rather, the qualitative study found out that emphasis has to be placed on the front-end phases until system structuring. The most significant difference to single product development could be found in the steps "detailed market analysis" and "system structuring" which were covered in much more detail, with broader scope and more stringency. This is also true for the other steps of the requirements and concept phase, but not with that high intensity.

While single product development straight starts with requirements engineering or a short market analysis, during modularisation it is first important to get broad and detailed market knowledge in order to get a stable base for all products under scope. For instance, the market analysis phase includes the definition of target market segments based on strategic positioning within the competitive environment, definition of risks and future trends, prioritised success factors, unique selling points, defined customer value and detailed numbers about forecasts of volumes, prices and target costs.

Requirements engineering does not contain any specific additional steps. However, product managers and engineers have to work on a much broader scope of interdependent and conflicting requirements. The complexity of "requirements overload" requires a clear prioritisation and focus on most relevant requirements to be covered by the modular system.

The allocation of requirements to product functions and solution principles requires more resources than without modularisation. This is because engineers are not used to cover such a broad range of relationships to be considered. Moreover, they are not used to work with product functional analysis in practice and they usually do not provide traceability of requirements to different succeeding phases and projects.

The most significant difference between modularisation and single product development could be found in the step "system structuring". The need to formalise and make the architecture explicit for all future products is only present in the case of modularisation. This requires various additional activities like selecting an architectural concept out of different alternatives, defining the variance of future interfaces and modules and the definition of specifications for future module and product development.

The later phases (i.e. the design, testing and production preparation phase) are not given further attention during modularisation within this model.

A characterisation of the identified difference between modularisation and single product development in the classical new product development model can be found in Figure 52. The figure evolved from a more detailed overview in spreadsheet format which was established during the case study. The upper part of the figure shows the phases of the "traditional" new product development process. The elements along the bottom indicate for which phase an emphasis has been identified with modularisation compared to single product development and which phase has a distinct difference during modularisation transition. For instance, different activities or a different emphasis can only be found until system structuring/embodiment design where the product architecture specification is generated and passed on to other projects. There are no additional elements during later phases.

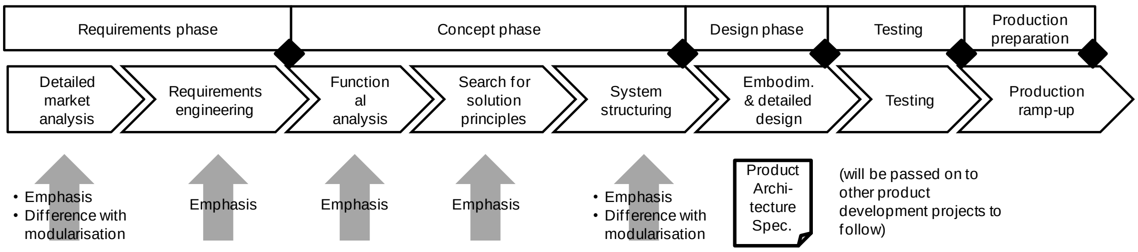


Figure 52: New product development process model applied to purposes of modularisation

The evaluation of this process model for transitioning toward modular system development revealed various advantages and disadvantages which are given in Table 10. The table can be seen as summary of the project "process integration of modularisation" which was mentioned above.

Table 10: Advantages and disadvantages of transitioning toward modular system development within classical new product development process

Advantages	Disadvantages
Relatively low initial investment required (e.g. only small organisational changes or few additional resources during front-end)	Lack of time and resources to consider specifications of multiple projects that eventually have not even started Lack of motivation of employees to work on modules or products beyond the scope of

	their direct product development project Lack of knowledge and transparency about items not to be covered by the current development project (future products cannot be treated in the required detail level) Lack of collaboration and communication across different projects to follow

Given the pros and cons, it has to be concluded that this approach is not suitable to mitigate the issues mentioned in the preceding section and to transition toward modular system development with the scope of multiple products from several different development projects. With this approach, engineers tend to focus on the current development project. Specifications that go beyond that development project are on their own too weak to make succeeding projects sticking to the overall modular architecture. In fact, this approach should only be chosen for a small number of products to be modularised which evolve from the same product development project.

- Classical platform development approach

As a result of the flaw of traditional new process development models, companies introduced the concept of platform development. In this approach, platforms are developed centrally in order to provide shared assets to different derivative development projects.

The platform life cycle usually starts with the anticipation of future market needs and the elaboration of innovation opportunities for derivative products. Based on the platform plan, the concept of the platform together with its embodied design can be developed. Those platform elements that are most likely to be shared across different products are already designed in detail and at least preliminary tested by the platform development team during this phase. The succeeding phase of platform management deals with the introduction of the platform through implementing it into derivative product development projects. Moreover, this phase contains the activity of platform maintenance. The key activity of platform maintenance is to adapt the platform to changing needs of derivative products without losing sight of synergies with other projects. Thus, platform management either deals with the confirmation that derivative products are in line with the platform, that the platform will be adapted with penalties for the derivative project or that out-of-scope needs of derivative projects are rejected after verification. A complete overview of the platform approach is given by Figure 53.

Even with a platform approach, there still need to be considerable resources invested into development of derivative products. Each derivative product or product family has to come up with a promising business case derived from market and sales planning activities. This activity is followed by a bid / acquisition phase or the decision for a strategic market launch with the creation of a detailed product specification. Those specifications that are covered by the platform are realised by reusing the detailed design of platform elements. Unique customer specifications are coordinated with the platform process and after a positive decision from platform development, realised within the derivative product development projects.

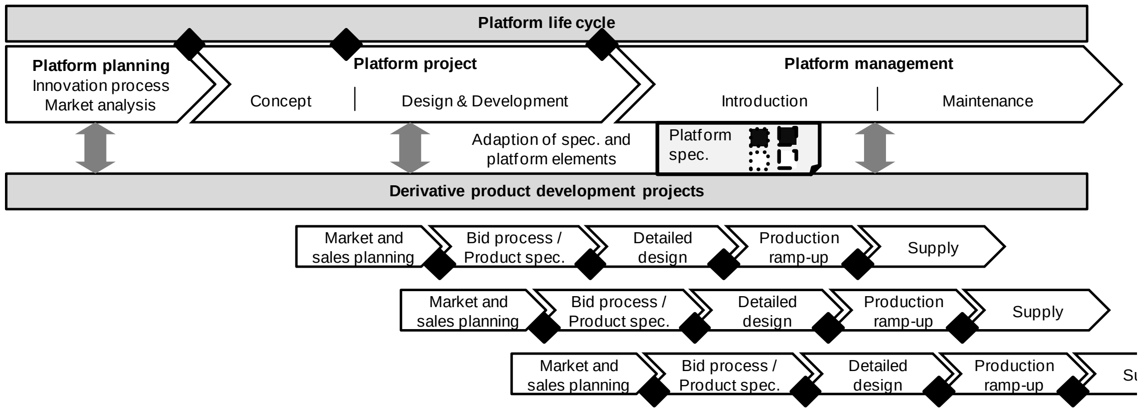


Figure 53: Platform development process with derivative product development

The "modularisation process integration" project mentioned above revealed that the traditional platform approach constitutes the counterpart to the model of new product development by explicitly considering the interplay between common platform elements and several distinct derivate product development projects. Considering the identified issues from the previous section, this approach yields some definite advantages. The central process with dedicated resources leads to appropriate time and motivation for platform developers that are willing to achieve project-overarching commonalities.

However, there are also clear disadvantages of the platform approach like the heavy initial investment and delayed benefits of platform development as illustrated in Figure 50 or Figure 51. The degree of transparency, collaboration and synchronisation between product and platform development was facilitated with this process model compared to traditional new product development. Nevertheless, these issues still seem present for this kind of process approach. This finding was confirmed during other case studies (Ponn, 2015). Another major drawback of the platform development approach can be denoted as the rigidity and inflexibility of the platform itself. In sum, such a traditional platform approach should not be pursued for transitioning toward modular systems within *existing products* which shall yield quick results. An overview of advantages and disadvantages of this approach is given in Table 11. The table can be seen as summary of the project "process integration of modularisation" which was mentioned above.

Table 11: Advantages and disadvantages of transitioning toward modular system development with a traditional platform approach

Advantages	Disadvantages
Appropriate time & resources Appropriate motivation for platform developers	Tremendous initial investments due to change throughout the organisation Launching new products on the market not possible until platform accomplishment and development of first derivative products Revolutionary approach not possible to be undertaken within existing product development Rigidity and inflexibility of platform compared to modular approach
Better than with single product development but still need for considerable improvement: transparency, synchronisation and collaboration between platform, derivative product development and among derivative product development projects	

- Identified common modularisation approach of consultancies

The qualitative study identified a third stream of how the development life cycle could support modularisation transition. During the "modularisation process integration" project, the primary case company also contracted consultancies with expertise in modularisation processes. The approach used by these consultancies has also been evaluated within the primary case company. This approach takes the use of different elements of the traditional new product development approach, of the platform approach and of the systems engineering approach where the product architecture specification is passed on to several different domain engineering projects.

The first two phases of the modular system life cycle have been conducted during several workshops with product development teams. These workshops scoped the complete range of products to be derived from the modular system. First, a detailed market analysis was conducted, followed by extended requirements engineering for the whole modular system. Second, the concept phase dealt with the determination of functions and technical solutions to be realised with the modular system. Then, the common modular reference architecture for all products under scope was derived by applying dedicated modularisation methods (see Section 5.4.2 for their evaluation). After design and verification of the modular architecture, the modular system was planned by providing "rough" specifications about modules, the number of module variants, their realisation, interfaces, and future products to be delivered to customers. Third, these architecture specifications were handed over to derivative product development projects. It was the purpose of derivative product development projects to a) develop products to be launched quickly onto the market and b) to "fill" the modular system by providing reusable modules that are compliant to the specifications of the common modular architecture specifications (i.e. "realisation phase" of the modular system life cycle). Fourth, after several derivative product development life cycles the modular system should be realised or "filled" with the planned amount of module variants from which a large number of products can be derived easily by configuring existing modules. This phase constitutes the "usage phase" of the modular system life cycle.

The distinct characteristics of this approach are as follows:

- Extensive front-loading for full-scope requirements and concept phase until planning of the product architecture, conducted in a series of cross-project workshops
- Distinct product development projects that develop modules for the modular system based on a common product architecture and parallel development of derivative products.
- No further support for actual realisation, usage and update of the modular system from a modular system life cycle perspective

An overview of this kind of modularisation transition approach is given in Figure 54.

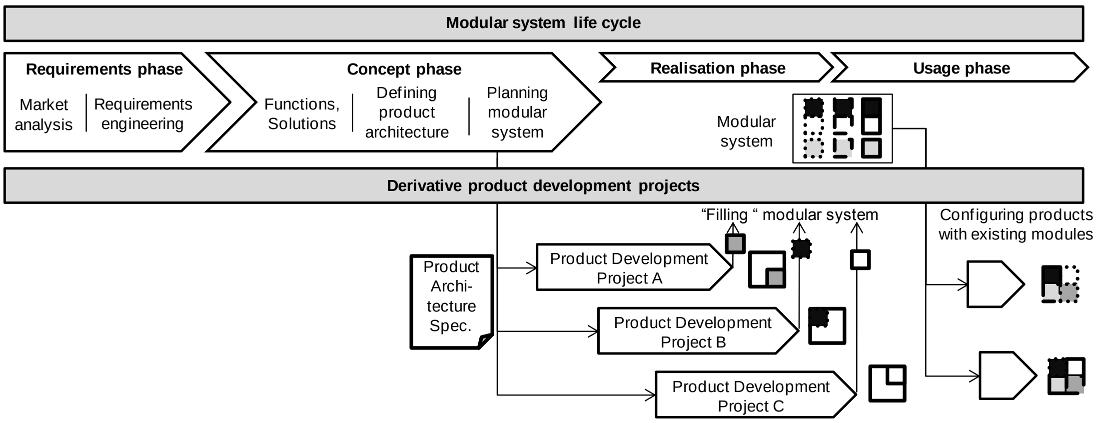


Figure 54: Proposed modular system development process of consultancies analysed during the qualitative study

It is the definite advantage of this approach that it can be pursued with acceptable initial investments within existing products and while still being able to launch new products relatively quickly after initiating the transition. While there are appropriate resources, motivation and transparency dedicated to modularisation until the product architecture specification is set up, the situation changes after the modular system is intended to be filled from independent development projects. During later phases of the modular system life cycle, i.e. during realisation and usage phase, lack of resources, motivation, transparency and synchronisation pull derivative development projects from a modular system perspective toward a single product focus. Consequently, the lack of mechanisms that make single development projects stick to the overall product architecture endanger the stability of the modular system quickly after the product architecture has been established. The root cause for this problem is the lack of support during later stages of the modular system life cycle which remains undefined for this kind of transitioning approach. An overview of advantages and disadvantages of this kind of modularisation support is provided by Table 12.

Table 12: Advantages and disadvantages of transitioning toward modular system development with approach of consultancies

<table>
<tr><th>Advantages</th><th>Disadvantages</th></tr>
<tr><td>Possible to pursue this approach within existing products

Appropriate resources, motivation and transparency between projects until modular product architecture is planned and specified</td><td>Lack of resources, motivation, transparency and synchronisation after common modular reference architecture is handed over to single product development projects

Loss of stability of modular product architecture and breaking apart of modular system due to individualisation of projects quickly after product architecture is established</td></tr>
<tr><td colspan="2">Medium initial investments due to workshop-procedure without changing organisation and process landscape and relatively low disruption of launch of new products.</td></tr>
</table>

The next section will turn to investigate use and limits of modularisation methods that are applied along above mentioned different development life cycles.

5.4.2 Modularisation methods

Modularisation methods were identified by the literature review as major research stream to support modularisation design activities. The qualitative study gave the unique opportunity to analyse modularisation methods from a longitudinal industrial perspective.

While case companies were studied on support for modularisation transition, the question emerged how useful are formal modularisation methods, like those that were presented in the literature review (see Chapter 3.1), for industry during modularisation transition. This question was answered by conducting several multi-perspective studies in the course of the overall qualitative study in industry.

- What is considered as important by industry?

During the prolonged study in the *primary case company*, different modularisation methods were evaluated based on their potential to be applied in industry. This evaluation and selection was done prior to applying a method. During the study, there were different iterative review cycles. The evaluation took place in an evaluation review meeting where presenting engineers or engineering managers, the heads of development of six different business units, the head of central engineering, the head of product management and the vice president development of the company took part or sent a delegate. Each review meeting was prepared by creating a detailed preliminary evaluation matrix with different scores and criteria. The preliminary evaluation matrix evolved by interviewing consultancies, different engineers who used the methods in practice and by participating in development projects that took use of modularisation methods under evaluation. The output of each review meeting was a refined and agreed evaluation matrix. The evaluation matrix cannot be shown in much detail within this work due to confidentiality reasons.

The following table shows how industrial practitioners from the primary case company rate the importance of different evaluation criteria for modularisation methods.

Table 13: Evaluation criteria for modularisation methods and their importance for industrial practitioners

Benefit (expected)	
Criteria and weight	**Sub-criteria and weight**
Benefit (now): 30 %	Complexity reduction (e.g. # of parts): 30 %
Benefit (future): 30 %	Consideration of the market: 10 %
	Optimised product architecture: 10 %
	Product architecture administration: 10 %
Benefit (broadness): 40 %	Consideration of strategic aspects: 10 %
	Consideration of PLC / value stream aspects: 10%
	Flexibility of product architecture: 10 %
	Software support: 10 %
Grand total:	100 %

Effort (expected)	
Criteria and weight	**Sub-criteria and weight**
Effort per project: 50 %	Internal cost / time: 30 %
	External cost / time: 20 %
Effort for implementation: 50%	Process Integration: 15%
	Building up know-how (cost/time): 10 %
	Difficulty to keep architecture stable: 15 %
	Software cost and training effort: 10 %
Grand total:	100 %

Table 13 is divided into two parts. The first part on the left shows the benefits on which modularisation support should focus on. The table shows that short-term, long-term and a broadness of benefits should be addressed. Given existing modularisation methods, it is not surprising that such methods should address complexity reduction, the market, establishing the product architecture, strategic aspects, PLC / value stream aspects, flexibility and software support. All these criteria have been found to be adequately addressed by modularisation methods that were evaluated.

The criterion "product architecture administration" intended to provide architectural stability was weighted equally with all other criteria. However, this criterion was not found to be considered in much detail by traditional modularisation methods that reflect state-of-the-art.

The right hand side of Table 13 shows that industrial practitioners have high concerns about the effort that is needed to apply modularisation methods in practice. It is not surprising that whenever possible, they try to reduce internal effort (e.g. for engineers), external effort (e.g. for consultants), effort to built up know-how, software cost and training effort. Moreover, industrial practitioners emphasise that it is important that the elements of the method can be integrated into existing processes. Another point that also was mentioned on this side and rated with significant 15 % importance by industrial practitioners was that the method provides effective help to keep the architecture stable once it is established. This is a point that has so far not been considered by traditional product architecture methods.

In essence, the surprising point of this study is that unlike traditional modularisation methods where the weight is vastly on establishing the architecture, working on the market and on other factors, industrial practitioners are seeking support on how to efficiently administrate and keep product architectures stable after establishing them. Moreover, practitioners do not want to go through additional modularisation methods, but they want to integrate or change elements of their existing processes.

- Application of methods by industry and application by consultants

The question how useful current methods for modularisation transition are can also be approached by answering if companies with successfully implemented modular systems took use of traditional modularisation methods from, for instance, literature in Chapter 3.

Table 14 gives an overview of the results of the study.

Table 14: Application of modularisation methods of different case study partners

<table>
<tr><th>Method applied</th><th>No method applied</th></tr>
<tr><td>Modularisation consultancies:
6 consultancies were studied, all of them applied modularisation methods for the transition</td><td rowspan="2">Industrial projects or companies not supported by any consultancy:
None of the 13 studied mature, mixed or younger cases without consultancy on board took use of a modularisation method. This proved particularly to be true for companies that were studied as benchmark partners for successful modularisation.</td></tr>
<tr><td>Industrial projects supported by modularisation consultancies:
11 projects were studied, all of them applied modularisation methods for the transition</td></tr>
</table>

Among the 30 cases studied during the course of this work, 17 cases applied modularisation methods whereas 13 cases did not apply modularisation methods. However, if the cases are further characterised whether they involve consultancies or whether they solely apply expertise from industry, following points get obvious from Table 14:

- All 17 consultancies or consultancy-related cases applied modularisation methods.
- None of the 13 cases that solely relied on engineering expertise to establish modular systems applied traditional modularisation methods.

Given these points, it can be summarised that from an industry's point of view, traditional state-of-the-art modularisation methods cannot be seen as precondition for successful modularisation transition as there are highly successful cases that made the transition with other means of support than with the suggested modularisation methods from the literature review.

- Participation in the application of traditional modularisation methods in industrial projects

This study was an in-depth study following a participant-observer approach during the application of two modularisation methods in development projects in industry over a prolonged period. The methods applied were similar to the method of Schuh et al. (2007) and the MFD-method of Erixon (1996) and Nilsson and Erixon (1998). The analysed methods took use of other well-researched elements. These elements included, for instance, guidelines (Ulrich, 1995; Ulrich and Eppinger, 2012), optimisations (Holtta, Tang and Seering, 2003), visualisations (Ericsson and Erixon, 1999; Goepfert, 1998; Pimmler and Eppinger, 1994; Stone, Wood and Crawford, 2000) or evaluations (Ripperda and Krause, 2014).

Both methods have been applied with support of experienced consultants in different projects within the primary case company. The methods have been applied and studied over a period of eight months in average.

The expectation of management concerning both approaches was to create working modular systems for a wide variety of products. In the course of the study, several observations concerning use and limits have been made.

In both projects, most of the time (approximately 75% roughly calculated by applied workshop days which cannot be seen as an exact quantitative figure but as a figure to facilitate understanding) was spent for "preparing" module clustering. This means that this time was spent for market analyses, requirements engineering, functional analysis and selection of technical concepts supported by defined support tools (e.g. QFD matrices, Porter's Five Forces Analysis or Pugh Analyses).

Afterwards, the remaining time was spent on module clustering and conceptually defining modules, interfaces and products to be derived from the modular system. Main supporting tools for module clustering were originally adapted from predominant tools that can be found in literature (see Section 3.1 and Appendix A), namely Design Structure Matrices, Module Driver Matrices, optimisation algorithms and graphs (e.g. functional flows or network diagrams).

Given above activities, the applied methods can be divided into two parts. The first part is not directly related to module clustering. It concerns all steps in the product development process that come before module clustering. This shall ensure that the required scope in those product development steps is provided. In other words, this means that the market analysis covers all markets, requirements, functions and technical solutions under scope of the entire modular system. The second, remaining, part is directly related to module clustering. During this step, the product architecture is set up in a sufficiently broad scope to cover all modules and products to be derived from the modular system. Moreover, the product architecture is made formal and explicit during this part. For both parts of the methods, there was no deviation from what can be found in literature concerning main input factors, side input factors, development phases, architecture representation and improvement (see Section 3.1 and Appendix A).

Thus, the problems and benefits by applying the method encountered during the study can be seen as *use and limit* of existing support for modularisation transition. They evolved following the detailed research methodology of Section 5.2 and are ordered along the development life cycle. The findings concerning this matter are summarised below.

From the study can be derived that the real *use* of applying modularisation methods in the primary case company has been as follows:

- Strong consideration of phases prior to module clustering in the concept phase:
    - The methods helped to gain market knowledge deeply and for all products under scope in a methodical way with documented and agreed decisions. In a conceptual state, the methods helped to sort out those innovations and technologies that are promising candidates for implementation.

- Applying the methods ensured that sufficient attention to front-loading was paid. Activities that were required by the methods during this phase could not be skipped or handled with inappropriate attention by following the method.
- The methods helped to make it clear to engineers and product managers that it is necessary to only focus on most promising requirements and design solutions.
- The methods helped to get a common agreement among engineers and engineering managers of what should be ideally common across different products. Already during applying such a modularisation method, engineers agreed on more than 50 % reduction of complexity through the reduction of the number of applied module variants across different products. However, this could have also been a result of the cleaning up effect of a previously uncontrolled growing module portfolio. Moreover, this agreement was done at a quite early stage and not during the stages where most of the problems occurred to achieve these goals.
- A multidisciplinary team, especially that of design engineers, was involved from the very beginning and, thus, was later able to implement the basic rationale of front-end decisions.

- Useful aspects during module clustering during concept phase:
    - Methodological support during product architecting itself helped to look at the product architecture from different stakeholder's angles and made sure that all aspects (e.g. downstream life cycle and value stream effects) are considered during architecting.
    - The methods helped to make the product architecture formal and explicit (e.g. through visualisation) and to communicate it within the organisation.

On the other hand, applying the modularisation method had several detrimental side effects. The *limits* that have been encountered during the study are as follows:

- Phase-overarching limitations – High resource consumption that was withdrawn from product development projects:
    - Applying the methods consumed high amounts of resources in addition to "ordinary" project tasks from which project resources were withdrawn. This led to lack of resources in other areas and in succeeding phases and decreased the motivation of engineers to work on modularisation.
    - The main time during applying the methods was used for seemingly endless discussions about requirements to be implemented or not, different product architecture alternatives, the trade-off between concept decisions and the constant struggle between single project goals and global company goals with the modular system. Even though, these discussions were helpful to share under-

standing among different engineers, they were time consuming and could only be closed by decisions of top management. It was not possible to link design choices during product architecting directly with feasible cost reduction and other targets of modularisation. Thus, these modularisation discussions did not contribute to available time and motivation of involved engineers.

- Limitations during concept phase – Traditional architecting methods and algorithms (e.g. DSM, MIM, heuristics) only weakly contributed to improved architectures not justifying their time consumption:
    - Applying modularisation methods led to strong resistance among engineers. Engineers doubted that the applied tools for product architecting actually lead to better results and to better product architectures. If teams were not stuck in discussions, they were ticking boxes in matrices or filling templates and the like. During such sessions, the team lost trace of the overall big modular system picture.
    - In accordance with the literature review in Chapter 3, the applied methods took use of functional structures, functional-physical relations, physical interdependencies and various strategic factors like module drivers. While these factors might be useful in totally new product development, their application alone within existing products did not lead to better architectures. For instance, the application of different module drivers on part level was used to cluster parts into modules. However, such strategic considerations alone did not generate an improved product architecture. In fact, the resulting architecture was not feasible. The same result came up when the team solely applied functional or physical analyses.
    - When different factors from the previous point were combined in the form of a matrix, graph or formula, the resulting product architecture was nothing more but a compromise from different perspectives. However this study found out that the view on the architecture in different phases of the module life cycle will remain differently. For instance, design engineers had a more functional view on the architecture while taking physical principles into account. Manufacturing engineers choose to cluster modules in the best way for the production flow while service engineers choose to cluster modules for best serviceability. Modules in different life cycle phases might be the same, but there are certainly cases where this is not beneficial. Thus, the study found no use in throwing different factors into a matrix, graph, algorithm or formula and calculating an optimised module function out of it. However, this is the way how it is suggested by contemporary modularisation methods.
    - Module clusters that have been generated with the help of algorithms did have to be reworked with engineering knowledge and experience. Moreover, the architecture that has been drafted prior to applying a method was very similar to

the architecture after applying a method. This means that when transitioning toward modularisation within existing products, the architecture is already in the heads of experienced engineers. The study revealed only two cases within approximately 50 modules with several hundreds of parts where *two* parts were assigned to a different module than previously.

  - The methods have been seen by engineers as overloaded with features of little benefit but with high resource consumption. Moreover, especially during application of algorithms or complex equations, the study did not provide any evidence that application of such optimisation indeed leads to better modularisation results.

- Phases after concept phase have only been weakly considered

  - Methods did not consider later phases than product architecting of the modular system life cycle. These phases have only been considered during architecting by, for instance, asking questions of what kind of architecture the service guys would need for improved product maintenance. However, previous section showed that exactly these are the phases (e.g. evolution and maintenance of the modular system) where the modular system is most vulnerable to failure and divergence.
  - The study showed that the methods only removed issues (see Section 5.3) from the front-end of the modular system life cycle. The methods did not attack all other issues that occurred after that.
  - The study showed that although a method had been applied in projects, problems started as soon as the modular concept was handed over to derivative product development. During later phases, the methods applied did not provide any mechanisms to pull derivative projects toward global modular system goals.

To summarise the study, it is suggested that there is limited use in applying modularisation methods for modularisation transition within existing products. The strengths of state-of-the-art modularisation methods may be in the development of totally new solutions or in radical changes of underlying assumptions during the life cycle of the architecture.

However, the projects studied contained a high level of given artefacts and reuse from previous designs. Even in such environments, the application of a method might be beneficial for front-end phases of the process if some elements of the methods are integrated and aligned to existing company processes. Moreover, methods might be beneficial for designers if the principles how they intend to improve product architectures are used as some kind of “checklist”.

Above all, it seems that methods require high resource consumption while no evidence was found that an applied matrix or algorithm indeed leads to better and more stable architectures. It is suggested that architecting itself is not a matter of method, tool or algorithm, but rather a matter of skilled and experienced engineers with the right resources, knowledge, motivation and mind-set for modularisation.

Additionally, the most significant finding was that state of the art methods do not offer any advice on how to prevent modular systems from diverging after concept phase. Though, these neglected phases were seen as significant by industrial practitioners when they were evaluating modularisation support. Therefore, the applied methods were not capable to solve the issues presented in Section 5.3. This finding is supported by the fact that this qualitative study did not identify a single case where a modularisation method has been transferred or applied in industry without involvement of a consultancy. Table 15 gives an overview about the investigation of modularisation methods within this study.

Table 15: Summary of use and limits of modularisation methods within existing products as output from the qualitative study

Use of methods	Limit of methods
Strong consideration of front-end issues like market analyses and variant decisions	High consumption of resources
Different viewpoints of multidisciplinary stakeholders (e.g. production, maintenance), expected downstream life cycle and value stream effects are considered	Limited use for establishing architectures within existing products
Structured way to make product architecture formal and explicit (e.g. through visualisation)	Low relevancy and weak use of product architecture matrices (e.g. functional-physical relations, functional relations, interface matrices, Module Indication Matrices) for establishing the architecture
	Weak use in applying algorithms and mathematical optimisation
	Only weak consideration of phases after product architecting, identified issues are not resolved during this major part of the modular system life cycle

5.4.3 Summary of use and limits of existing support

It was the aim of this section to test use and limits of existing main support for modularisation transition. Therefore, two main research streams for modularisation in engineering design have been studied.

The first part dealt with the analysis of current development life cycle models for the suitability for modularisation transition. Among the three categories that have been analysed, none of the models proved to be capable to remove the issues that have been identified in Section 5.3. First, the traditional product development model is largely incapable to handle the interplay between the modular system development project and derivative product development projects. Second, the identified platform development approach strongly supports the interplay between the platform and derivative products, also during later stages of the life cycle. However, this approach has not yet been adapted to flexible modular systems and transitioning within existing products. Third, the modularisation development life cycle model of consultancies has its strength in front-end phases of the development life cycle while performing poor during modular system realisation and usage.

The second part dealt with the analysis of modularisation methods that have been applied along the development life cycle. While there is some usefulness of these methods, especially during front-end phases, the methods failed to provide stability for the modularisation after concept phase. The tested support has been seen as time-consuming and partly irrelevant. Especially during modular system realisation and usage and its interplay with derivative product development, the scrutinised methods did not provide any support nor did they much contribute to better transparency, information, motivation, time and resources for the modular system development process. After all, the methods that provided a wide ranging set of optimisation, visualisation and evaluation support did not provide any mechanisms to pull derivative development projects with local goals toward global goals of the modular system.

Figure 55 has been generated from the analysis above to give an overview of the modular system life cycle and its interplay with derivative product development projects. The figure also shows corresponding issues and support to each life cycle phase. It should be noted that the figure is not true to scale.

Data from the primary case company are as follows: The requirements and concept phase for the modular system took in one project six months and in the other project nine months. Compared to that, the realisation and usage phase (i.e. evolution and change phase) is much longer with several product life cycles, each approximately four years. Thus, the right side of the figure should be extended with a true scale.

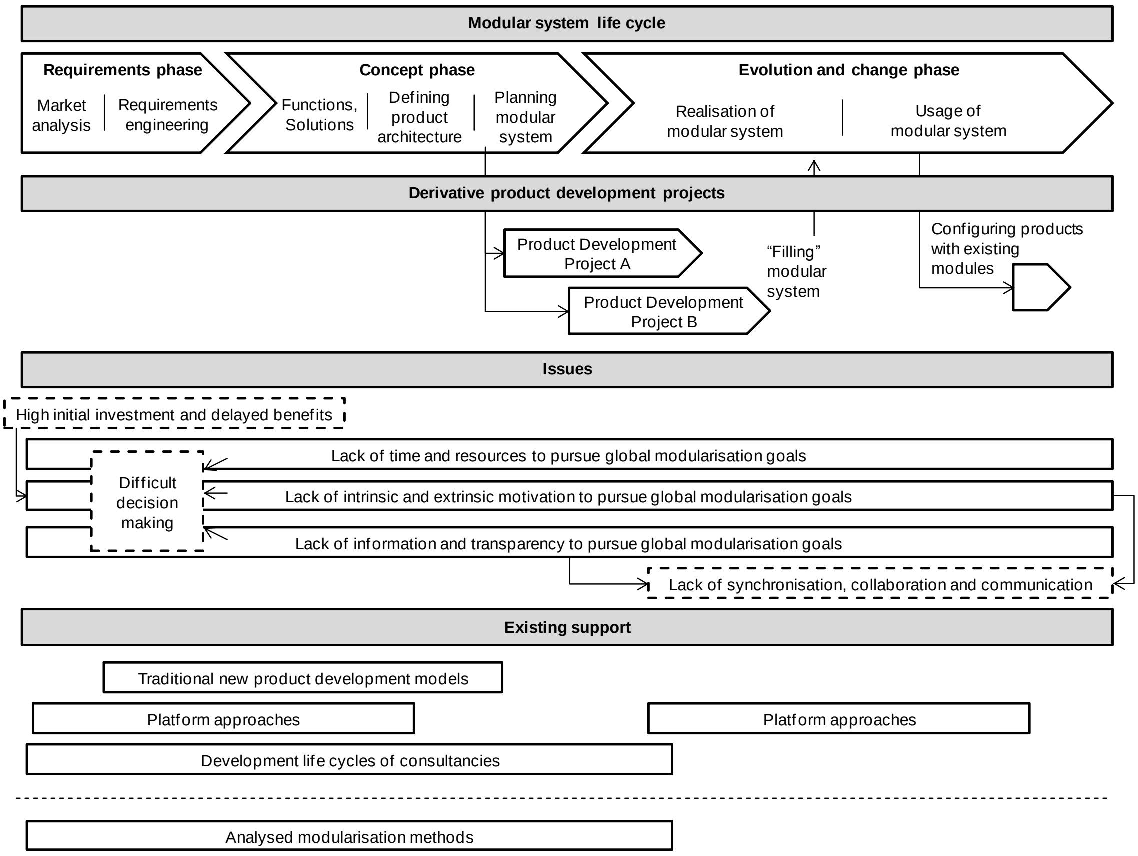


Figure 55: Overview of issues and existing support during the modular system development life cycle

5.5 Support framework for modular system development with focus on stability

Having shown what the real issues for modularisation transition are and where current means of support help and where they fail, this section will suggest a framework for modularisation transition.

The framework evolved during the qualitative study, mainly by consulting the collaborating case organisations for their solutions for dealing with issues arising during modularisation transition, i.e. support during later stages of the modular system life cycle and mechanisms that pull *diverging derivative projects* toward modular system development. The findings that are presented within this section have been discussed, refined and tested within the primary case company.

The framework for modular system development consists of the elements discussed in the next sections. The elements are also shown in Figure 56. The "focus on stability" discussed above means that the framework's focus has to be on the post-architecting phase where most of the issues occur and where existing support most lacking. In other words, the focus is on those phases where the modular system is most vulnerable to diverging and breaking apart. Nevertheless, those pre-architecting phases that are relevant for stability in later phases have also been considered.

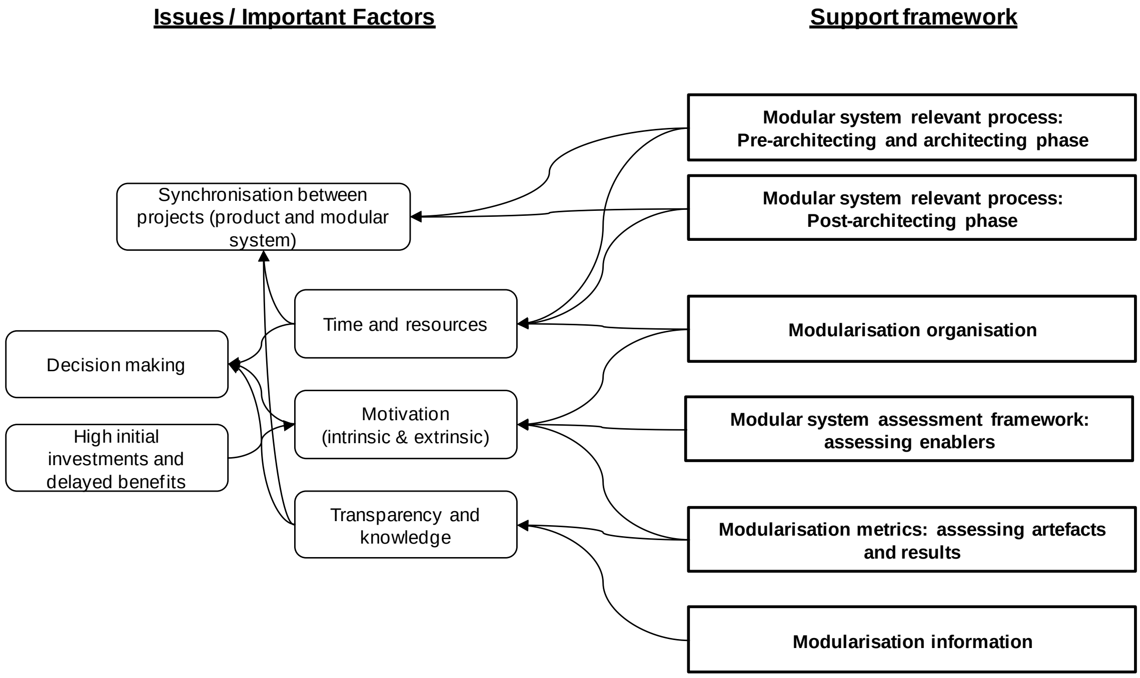


Figure 56: Main cause-effect relationship between issues, important factors and suggested modular system support framework

5.5.1 Modular system relevant process: pre-architecting and architecting phase

It is the aim of this element to make sure that all actors preparing and establishing the architecture have enough time and resources to consider the whole modular system instead of local solutions. Moreover, the prescriptive character of this phase must give guidance on the synchronisation between derivative product development interests and the interests of the modular system. The result of this element should be an agreed, feasible and fixed plan how to develop the elements of the modular system.

5.5.2 Modular system relevant process: post-architecting phase

This support element aims at prescribing activities that make the purpose of the engineering actor's activities to develop derivative products and modules that are compliant with the plans and specifications of the modular system. Where the specification of the modular reference architecture has to be adopted, this element provides guidance how to make such adoptions without losing stability of the modular system. By giving such synchronisation activities a dedicated process phase, involved employees are provided with adequate time and resources to work for the global goals of the modular system.

5.5.3 Modularisation organisation

Usually, engineers working on a product to be delivered to the customer are either assigned to functional departments or to derivative product development projects. By establishing modularisation roles or even organisational departments, there will be employees pulling functional departments and derivative product development projects toward overall modular system specifications. Due to their direct assignment to the modular system, these roles are equipped with adequate resources, time and motivation to work for the stability of the modular system.

5.5.4 Modular system assessment framework: assessing enablers

The modularisation assessment framework measures if identified enablers for stable modular system development are established. Therefore, it assesses whether all relevant means of support are in place so that engineering actors pursue the modular system goals of the company in addition to local goals. The application of this measurement framework ensures that all involved roles have appropriate extrinsic motivation to act according to best practices of modular system development throughout the whole development life cycle. The provision of these important enablers shall in turn lead to better technical results of the modular system.

5.5.5 Modularisation metrics: assessing artefacts and results

In general, product development projects are only measured on single project level. For instance, projects are measured on their speed with which they finish development of products or on direct product costs. This element of the framework shall ensure that development projects are not only measured based on local goals, but also on the achievement of global goals like contribution and adherence to the modular system. In fact, modularisation metrics which measure the artefacts of the modular system shall improve the motivation of involved roles to think beyond their current project and to provide transparency concerning the intended performance of the modular system. This element is of particular importance for post-architecting phases where it is important to early prevent derivative development projects from diverging from the modular system.

5.5.6 Modularisation information

This element of the framework helps to provide information about the modular system and to make product architecture information explicit throughout the company. Thus, it creates a modular system view in addition to a product- or project-centred view in standard IT-systems (e.g. PDM, PLM, ERP) of companies. By providing this kind of information, transparency and knowledge about the modular system is improved which helps engineering actors to develop in compliance with architecture specifications. Moreover, this framework element helps to synchronise relevant data between different projects that are supposed to develop according to the same plans and specifications. The provision of information also helps in decision making. The coherent documentation of fixed decisions about the modular system helps to suppress seemingly endless recurring discussions. Another purpose of modularisation information provision is the automated derivation of modularisation relevant metrics (see previous point). In sum, the provision of modularisation information within standard IT-systems is located in post-architecting phases of the modular system life cycle and helps to provide stability during realisation and usage of the modular system.

5.5.7 Support framework in the context of the development life cycle

Figure 57 shows how the framework for modular system development with focus on stability is related to the modular system life cycle.

When a company embarks upon modular system development, it does not necessarily have to change or implement all of the mentioned aspects. However, this framework can be used as a starting point for further analyses. If a company identifies some gaps within its modular system development, this framework can be used as a first guide for further improvement.

For the purpose of the research work reported in this thesis, the framework will be used to derive a more detailed evaluation framework for the assessment of enablers for modular system development. This evaluation framework will be presented in the next chapter.

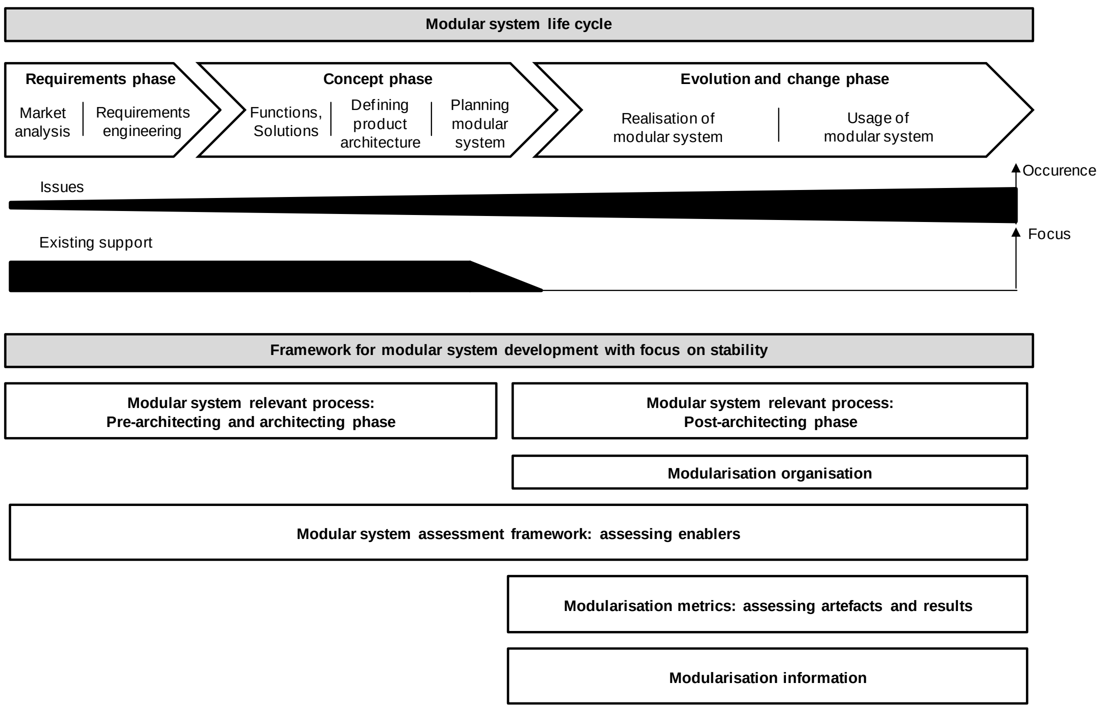


Figure 57: Support framework for modular system development with focus on stability in the context of the modular system life cycle with its issues and existing support

5.6 Discussion

The multi-faceted longitudinal case study (see Section 5.2) identified issues that companies encounter during modularisation transition. Moreover, existing modularisation support was evaluated in industry. The findings of the investigation were used to create a modularisation support framework with focus on stability which is built around the development life cycle. This allowed obtaining some new insights which are discussed in the following sections.

5.6.1 Modularisation transition and the development life cycle phases

The findings of this study that lack of knowledge, transparency, time, resources and motivation are major barriers in transitioning toward modular system development support other studies in the related field of introducing platforms in industry (see second part of Section 5.1). The fact that other studies in this field are rather theoretical derivations or narrow case studies, this longitudinal field study in industry with multiple cases helps to strengthen the common understanding in this field and to underline contemporary research findings of the second part of Section 5.1.

However, due to the consideration of the field from different angles and the coding analysis with the support of a relational database, categorising identified issues around different life cycle phases resulted in more detailed and new insights. These are shown below.

- Issues encountered during different life cycle phases

Requirements Phase:

The finding of this research that the requirements phase is a critical phase for modularisation confirms the findings of available literature in the field. A considerable amount of research stresses the importance of the requirements phase for modularisation and stresses the need to support this phase by additional elements like the application of various product management and market research methods (see literature about modularisation support in Chapter 3 and Appendix A). However, it has been neglected so far that it does not meet the needs of industry if researchers just add new elements to the requirements phase of a modularisation process without pointing out the need to install facilitating means of support for their application. Only such considerations may prevent overload of engineers during this phase with resulting paucity of time and resources.

Concept Phase:

After studying literature about modularisation support, it could have been concluded that this phase is the most critical one for modularisation transition (see literature about modularisation support in Chapter 3 and Appendix A). However, this study suggests that even though this is an important phase to consider, it is not the most critical one. Moreover, available literature suggests that optimising or improving the product architecture during this step is the most critical activity related to modularisation transition. The results of this study differ from that view in a number of respects. First, architecting is not seen as main issue during this phase. Second, no evidence could be identified during this study that available support remedies relevant issues during this phase. Rather, the question emerged if the application of very detailed and theoretical modularisation methods has indeed some negative effects (e.g. losing the big picture). Third, yet unrevealed by literature, this study suggests that the main issues during this phase concerning modularisation transition are rather knowledge and time to fix decisions that are necessary to create stable specifications for the future modular system.

Design Phase:

Literature regards the design phase more as side phase for modularisation than as major phase where many issues concerning modularisation transition occur. Actually, most of the studied approaches consider product architecting to be finished after concept phase (see literature about modularisation support in Chapter 3 and Appendix A). Whereas, the results of this study indicate that the design phase is of major importance for modularisation transition due to identified issues during this phase. The design phase has been identified to be crucial for transferring the concept of the modular system into reality. If the design phase is not considered during modularisation, the chances are high that the modular system already diverges during this phase. The raise of the importance of the design phase is quite innovative, compared to what can be found in literature.

Testing phase:

Even though testing is frequently linked to the benefits of modularisation, issues concerning testing have not been identified so far. It seems that literature has not made any attempt yet to answer when and how to test if the modular system is indeed working. Particularly for modularisation transition within existing products, this means that although not all modules of the modular system have been developed, at least their combinability should be tested before the concept is handed over to derivative product development. This is a new aspect to the current field of research.

Production ramp-up phase:

It is the prevailing view of literature that there is one modular structure that is equally suitable throughout the whole company. One of the main concepts of modularisation methods is the concept of module drivers (see literature about modularisation support in Chapter 3 and Appendix A). These module drivers attempt to incorporate different company views into the modular architecture. However, in literature the question remains whether it is more beneficial to regard the architecture as a compromise between different company views or whether to represent the architecture differently for each company view. This would mean that each company function would have its ideal view on the architecture (e.g. production view). This issue has been further elaborated in Section 8.6.

Maintenance phase of the modular system:

Literature has not ascribed great importance to this phase. Thus, the maintenance phase, i.e. the evolution and engineering change of the modular system, has been neglected so far by literature (see Section 3.2). In clear contrast to that, the results of this study accredit this phase with major importance. If this phase is not treated with the required rigour, the stability of the modular system is endangered. This could lead to a phenomenon in modularisation which has been scarcely and superficially described by some platform researchers as "diverging" platforms (see Section 1.6).

- Issues encountered during different life cycle phases

Frequent phase-overarching issues that have been connected to modularisation by literature contain the disadvantages of modularisation like oversized modules, compromises in performance and the like. Other researchers focus on theoretical issues of modularisation. The issues “synchronisation effort” and “decision making” have so far not directly been related to modular product development, though, to different fields of research like multi project management. There is one first parallel study from platform development in industry which confirms these findings (Ponn, 2015).

Concerning high initial investment and only indirect delayed benefits, this study challenges literature. Most researchers link modularity to cost savings, higher performance and part number count reduction without pointing out that there are huge investments to be made in order to achieve these goals. Some researchers point out that there are huge initial investments to be made during modularisation (Chao and Ishii, 2004; Nobelius and Sundgren, 2002). However, it is claimed that this has only to be done until the product architecture is established (Jose and Tollenaere, 2005, p. 375) or that it is more realistic to make the transition in a lightweight approach (Wijnstra, 2004, p. 137). Given the high number of issues of this study that also occurred in high numbers after the product architecture was established, this study disagrees with current literature. For company-wide modularisation transition (e.g. across development sites, brands, engineering design traditions or markets), a lightweight approach is a dangerous temptation and companies have to be aware that there are high investments to be made, also particularly in the development life cycle phases after the product architecture has been established.

Given the reports from automotive industry, it is shown that modularisation is a promising approach that can indeed be successfully implemented. Moreover, they also indicate that this can only be achieved with painful heavy initial investments which also affect later phases of the modular system life cycle (see Section 3.3). This is supported by this work.

5.6.2 The limitations of support for modularisation

Product development models that adequately address multi-product development are only weakly covered in literature, if at all on a quite high level (Lehtonen, 2007; Ulrich and Eppinger, 2012). In addition, the whole aspect of the interplay between different development projects that are supposed to share common elements is not considered by literature (Nielsen, 2010).

Given the high complexity of today’s product development environment, constant cycles of iterations with work products that never seem to be finished and the majority of elements to be reused, it seems that contemporary product development models in industry are outdated as well. This study showed that the identified process models in industry may have advantages, but need adoptions in order to embrace commonality, reuse and modular system development with stable architectures. Consequently, this study supports the study of Nielsen (2010) who identifies a lack in the maintenance of elements to be shared across projects.

As indicated by the literature, establishing the “right” product architecture is very important. Recent researchers support such activities with modularisation methods. While modularisation methods might work well for new product development on a blank sheet of paper, the qualitative study comes up with strong evidence that modularisation methods are less useful than claimed by contemporary researchers. This is particularly true for transitioning toward modular system development within existing products in a brown field approach.

There are two sorts of modularisation methods: holistic ones and architecting ones. The studied methods in practice contained holistic and architecting elements. While some elements of the holistic part were found as quite useful (e.g. broad coverage of the market phase, evaluating different architecture alternatives from different perspectives, visualising the architecture), the usefulness of architecting methods itself (e.g. clustering algorithms) is questioned and seen as not properly validated in industry over a prolonged period. Current, overly sophisticated, architecting methods are always validated in the same manner. They are either validated in a sample project or through expert interviews without looking at the real issues and the sustainable benefits of their application in industry. Consequently, these findings of this study strongly contradict to the contemporary stream of literature and existing support that has been attempted to be transferred from academia to industry.

The second main finding of this study within this category is the weak support of the researched modularisation methods to remove issues during later phases of the design process. The applied methods did not address post-architecting phases, actually where earlier findings of this study identified the modular system to be most vulnerable to lose its support within the organisation. This significant limitation of existing support is strongly claimed by this study. Thus, this study supports claims from other researchers that there is paucity of research in this area (Bahns, Gregor Beckmann, et al., 2015, p. 4–5; Nielsen, 2010, p. 120).

5.6.3 A proposed support framework for modularisation transition

The support framework presented with its elements process, organisation, evaluation and information provision is confirmed by other studies with different purpose from different industries. The second part of Section 5.1 shows that other researchers came up with frameworks which cover similar aspects. However, none of the frameworks has a focus on stability of the modular system during its life cycle.

What is not considered in the framework of this thesis are the aspects leadership and change management (Arnoscht, 2011; Kraus, 2005) because they have been filtered out for the purpose of this work due to their presence in various transition projects. Hence, they are not particularly specific to modularisation, but are still valid and important. Skold and Karlsson (2007) also mention managing the trade-off between communality and brand identity in their framework. That this is an important topic to handle is strongly supported by this research work. However, this topic has been integrated into other elements of the research framework of this work (e.g. during product architecture definition or planning of the modular system). Section 5.6.5 will briefly provide the strategy how this topic is handled within the purpose of this work.

While the researchers presented in Section 5.1 do not provide much detail how to actually handle the mentioned aspects, further research on these topics can be found in literature. These publications support the findings of this study and are as follows.

The whole bulk of research about modularisation methods can be seen as some kind of process-related. Researchers like Arnoscht (2011) and Nielsen (2010) focus more on the overarching process side of this topic. Munk (2011, p. 153–155) depicts a "product platform system model" which describes platform-based product development with involvement of multiple stakeholders. Compared to these works, the work of this thesis emphasises the need to look deeply at the later phases of the modular system life cycle which includes the evolution and change of the modular system.

Concerning modularisation organisation, earlier works have mentioned the need for a match between organisation and the architecture (Goepfert and Steinbrecher, 2000; Oosterman, 2001). Recent studies have shown the need to adapt the organisation to modularisation (Arnoscht, 2011, p. 154–168; Persson and Ahlstrom, 2013; Schuh, Sommer and Rudolf, 2015; Bahns, G. Beckmann, et al., 2015) while reports from industry underpin these findings (Daimler, 2014). However, the researchers do not provide much detail how to actually adapt the roles of the develop{Citation}ment life cycle to the purpose of modular system transition.

The element of the evaluation of modular architectures is itself a major research stream in the field (Gershenson, Prasad and Zhang, 2004; Simpson et al., 2014), see Section 7.2. Where the findings of this Chapter differ from existing works is that the focus of evaluation should also be on later modular system life cycle phases.

Finally, information provision of architecture information is mentioned by different researchers as vital element (Bruun et al., 2015; Gebhardt, Bahns and Krause, 2014; Gebhardt and Krause, 2015; Harlou, 2006). In addition, examples for these statements can also be found in industry (Kreimeyer, 2012). However, other researchers have not made detailed studies how standard information systems can support the stability of modular systems.

While this thesis supports above mentioned aspects concerning activities to be worked on, its focus on stability of the modular system, i.e. consideration of the post-architecting phases has so far not been considered by other researchers. Thus, this innovative aspect is the basic foundation for the support in the following chapters of this work.

5.6.4 Evaluation of qualitative study

Compared to similar studies in the field, this qualitative study enhances the quality of such a study through sample size and timeline. However, compared to quantitatively significant sample sizes, it is a weakness that the sample size of this study is still quite small. This means that this study depends on the setting of the associated primary and secondary cases that were investigated. Although the findings are transferable to other industries as well, this transfer is facilitated by the background of the samples in different industries where they have been studied and due to the comparison of this study to literature of other industries. Constant comparison between different stakeholders from different industries and comparison to literature also helped to mitigate the issues that arise when the findings of a qualitative study are interpreted.

5.6.5 Different strategies to solve the modular system dilemma

When transitioning toward modular system development, the trade-off between different goals is omnipresent. One issue is the constant struggle between commonality and diversity and between overall company goals and local derivative product development goals. As a result, platforms are diverging and do not achieve the expected benefits. The struggle is referred to as modular system dilemma within this section. There are several strategies how research can resolve this dilemma:

Firstly, attempts have been made to solve this issue by guiding the modular system life cycle with support for behaviour, negotiations and discussions around the trade-offs and conflicts of interest (Arnoscht, 2011). Moreover, it is suggested to intensively communicate the platform strategy within the company (Munk, 2011, p. 151–152).

Secondly, other researchers try to directly make the impact of different architecture design alternatives transparent, "graspable" and "harvestable" (Fixson, 2002; Munk, 2011, p. 126–139). In our study, it was found that to date, there is not really a good solution for that and that (insufficient) support has so far only be developed for front-end decisions. For instance, a project that decides to go for a local solution will always directly benefit from such a local solution, despite advantageous implications on overhead cost or the supply chain that a global solution might have somewhere else in future. These global advantages cannot be harvested directly by the development project in the present. Thus, under such circumstances the project team will always decide to go for its own solution which is advantageous for its project calculation.

Thirdly, there is also the argument that the trade-off is just a matter of the product architecture. It is claimed that if the product architecture is designed in the right way, there will be no trade-offs afterwards. This is only partly true as front-loading and establishing a good architecture is definitely important, but it has been shown that this is not sufficient as the best architecture cannot resolve all emerging trade-offs and issues. This is also referred to as "making the platform solution the only one" (Munk, 2011, p. 151–152). However, this statement of Munk (2011, p. 151–152) could also be interpreted in a way that leads to the next point.

Finally, it is the approach of this work to make a thorough and agreed decision for a modular system architecture alternative and to push its realisation and usage throughout its life cycle. The "pushing through" is supported by the support framework presented in this chapter. This approach does not mean that there are not any changes to the architecture during its life cycle. However, these changes have to be carefully controlled. Munk (2011, p. 151–152) uses the phrases "goal setting and tracking" and "adjustment of the platform solution" for this approach, without providing any concrete details on realisation.

5.6.6 Significance and impact of findings

While industry makes high investments into the development of platforms and modular systems, there are still a considerable amount of failed cases that "burn" these investments (Gudmundsson, Boer and Corso, 2004; Harlou, 2006). Recent studies show that these issues are still present in industry (Munk, 2011; Ponn, 2015). With this qualitative study companies now have improved knowledge about what issues to expect during transition and which support to focus on. It is suggested that this will help industry to significantly reduce the number or impact of failures after they decided to introduce modular systems. The approach of this study to being built around the modular system life cycle, with a particular focus on later phases and the stability of the modular system is the first of its kind.

Another important point of the findings of this chapter is that a considerable amount of findings from literature, that were originally theoretically derived, that have their origin in other industries, that applied other research methods or means of validation could be confirmed (e.g. predominantly that of Section 5.1) or challenged (e.g. predominantly that of Sections 3.1.1, 3.1.2, 3.1.3 and 3.1.4).

The findings of this chapter are not only the input to the support approaches presented in the following chapter, but they also serve as guidance for further research. It is the clear message behind these findings to bundle resources in order to solve the real issues that engineers and engineering managers encounter in industry in their daily life, instead of developing overly complicated algorithms or overly sophisticated methods with the purpose to just satisfy a research thesis itself. Even researchers from this field complain that there has only been poor transfer of methods from academia into industry (Beckmann, Gebhardt and Krause, 2014, p. 121). Following a research framework, like the one presented within this chapter, will help other researchers to overcome this shortcoming.

5.7 Summary

This chapter dealt with RQ 1. Firstly, it identified issues during modularisation transition in industry. It has shown that the main issue is that actors during the engineering design process are following their local goals instead of "global" modular system goals. This is because the involved engineers neither have enough time, resources, knowledge or motivation to pursue overall modular system goals. In consequence, the stability of the modular system is undermined.

Secondly, the use and limits of support for modularisation transition was investigated. Current product development life cycle models cannot fully cope with the identified issues, nor are they capable to handle the interplay between modular system development and derivative product development projects during later stages of the design process. Modularisation methods have some useful elements (e.g. front-loading, market knowledge), but they have their limitation in making the transition toward modularisation especially within existing products due to lack of two main points: a) phases after the product architecture is established are not considered and b) weakness to resolve above mentioned issues.

Finally, a transition support framework with a focus on post-architecting phases and stability has been introduced. The framework concerns the consideration of process, organisational, evaluation and information aspects along the modular system life cycle. This support framework can be used as starting-point for more detailed research about support for modularisation transition.

In order that companies can identify their areas of improvement for successful modularisation transition, the next chapter will present an evaluation framework that is built around the findings of this chapter. This evaluation framework assesses in detail if enablers for modularisation transition are in place.

6 Modularisation assessment framework

This chapter gives detailed support how companies can improve their capability for modular system development (see Figure 58).

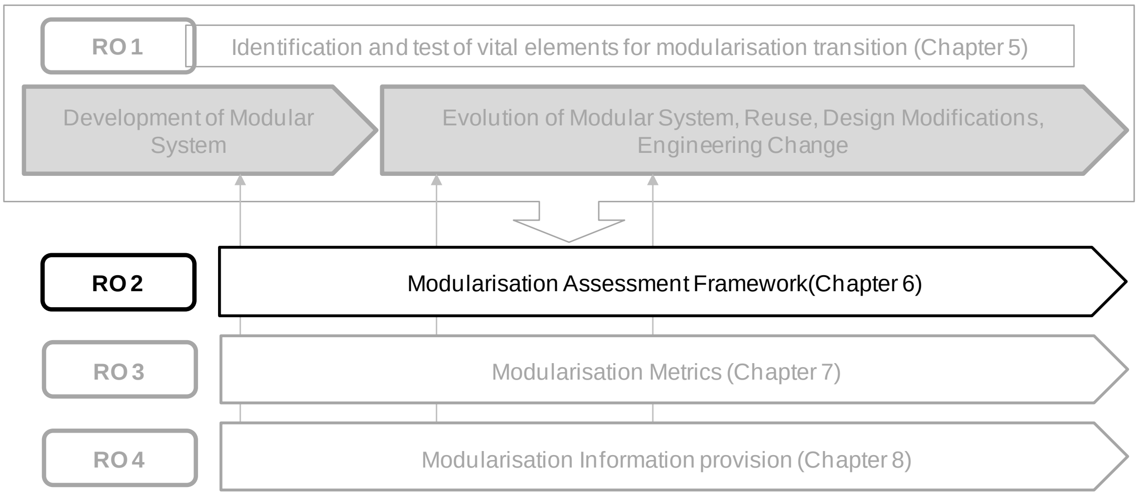


Figure 58: Context of modularisation evaluation framework within other chapters of this research work

As outlined in Section 5.6.5, it is the "philosophy" of this work to pull the organisation toward modular system development throughout the entire development life cycle. This means that there has to be a constant push in the organisation to ensure that enablers for stable modular system development are in place. In order to achieve this "push", this chapter will provide a framework with which organisations can flexibly monitor their performance and identify fields of improvement through constantly comparing their practice with best practices in the field. It is suggested that this approach will support companies in achieving their targets that they set for modularisation transition.

Figure 59 shows on the left part some important enablers for modularisation transition. These enablers consist of the modularisation assessment framework itself that is presented in detail within this chapter. The framework assesses the other enablers processes, organisation and modularisation evaluation through metrics. Metrics are included into the assessment framework as it is seen as important to measure the progress of modularisation transition with metrics which are further detailed in Chapter 7. Moreover, the assessment framework also takes into account modularisation information (see Chapter 8). Modularisation information has an "enabling" character because it provides transparency and knowledge to engineering actors. On the other hand, modularisation information that is explicit and easily available within companies can be used to derive modularisation metrics. This is the reason why it is also partially displayed on the results section of Figure 59 (on the right hand side).

The lower part of Figure 59 shows the basic reasoning of this work. It is suggested that the results of modularisation transition improve after some time (e.g. one to several development life cycles) after taking use of this assessment framework. In the end, the results of modularisation should correlate with the improvement of the maturity of enablers for modularisation transition with some delay.

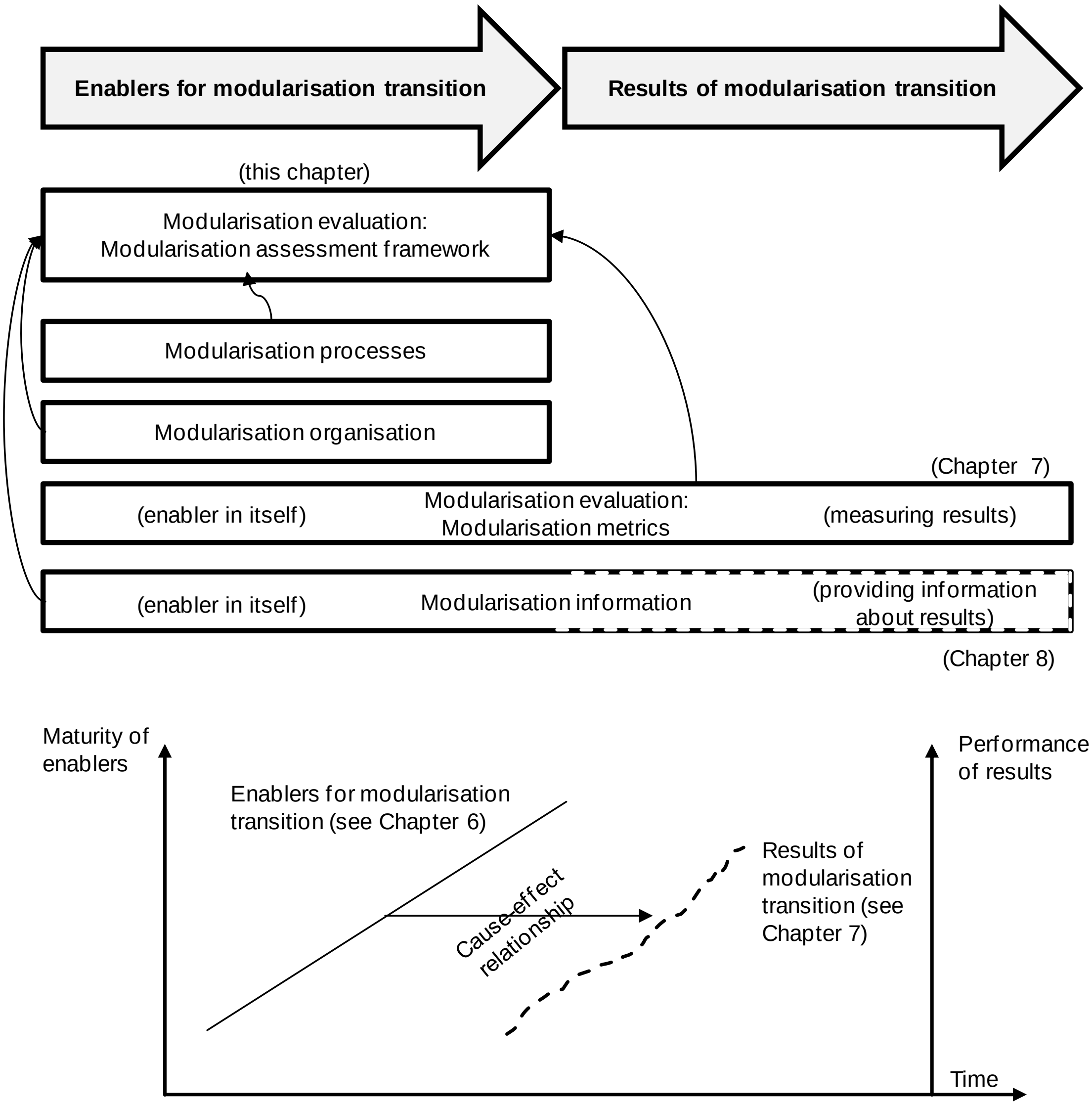


Figure 59: Relation between enablers and results for modularisation transition (in the style of the European Foundation for Quality Management EFQM-Model (Verband der Automobilindustrie (VDA), 2003, p. 22))

The next section will give a brief overview about the background of the assessment framework.

6.1 Assessment frameworks in literature and industry

In order to increase the performance of various processes, companies frequently take use of recommendations and best practices that have been collected by mature and well-performing organisations. One of the most famous collection of requirements and recommendations is the ISO 9000 series of standards. The ISO 9001 (DIN Deutsches Institut für Normung e.V., 2008) standard provides requirements for good organisational practice and important factors recommended to be established. In addition, the Annex A of the ISO 9004 (DIN Deutsches Institut für Normung e.V., 2009) standard also contains an assessment framework for measuring an organisation's capability and identifying potential areas for improvement.

Different means to systematically improve processes like process benchmarks, quality prices, key performance indicators, maturity models and audits have been applied in industry for years (Rauchenberger, 2010). For instance, the VDA 6.3 standard (Verband der Automobilindustrie (VDA), 2010) has been adapted to auditing needs in automotive industry and the CMMI-DEV Capability Maturity Model (Software Engineering Institute (SEI), 2010) is a detailed framework tailored to the needs of development, particularly in systems engineering and software development. Various different assessment frameworks can be found where particular areas shall be improved, where new concepts are introduced or where transfer from academia to industry is sought. For example, the areas of application comprise risk management capability for complex systems (Yeo and Ren, 2009), data management (Hüner, Ofner and Otto, 2009), lean sigma implementation (Barton and Thomas, 2009) or the leanness of manufacturing systems (Bayou and de Korvin, 2008).

In the particular field of complexity management, researchers have recently found ways how to identify areas for improvement in software product line engineering and component based software engineering (CBSE) by applying capability maturity models. Rehesaar (2011) concentrates on social factors for reuse of software components, Jasmine and Vasantha (2010) focus on the introduction programme of a reuse based software development process and Tripathi and Ratneshwer (2009) divide their measurement on a component process and an overarching component-integrating software process. Van den Linden (2005) divides his product family evaluation framework around four different dimensions. Firstly, the business dimension deals with matters like costs, profits or planning of variability. Secondly, the architecture dimension deals with mechanisms for variability and the relation of the common platform architecture to the application architecture in derivative projects. In the process dimension, Van den Linden (2005) differentiates between platform, application, collaboration and coordination processes which he all evaluates against a common maturity model like CMMI. Finally, he evaluates organisational structures and responsibilities according to their contribution to the platform and corresponding applications.

Even though, modularisation has gained increasing importance in recent years, no research in the field has been found that has a particular focus on an assessment framework for supporting modularisation transition. Moreover, the approaches investigated have a strong focus on different dimension like introduction, social factors, process or organisation. However, their investigations are not built around different development life cycle phases or "stages" like engineering design processes which can be assessed and controlled (Clarkson and Eckert, 2005, p. 54; Cooper, 2014). For this reason, the remaining part of this chapter will focus on the assessment of the transition toward modular system development and its position in the development life cycle.

6.2 Chapter overview

It is the aim of this chapter to answer RQ 2. This means that it is the objective to develop an assessment framework which companies can use to assess modularisation transition. After identifying an appropriate scheme for the assessment framework, it is the goal to develop and test the assessment framework for its application in industry.

In addition to the aspect that the assessment framework constitutes support that companies can use to assess their capability for modularisation transition, it also helps involved roles to get a deeper understanding about support for modularisation transition itself. This is done by providing consciousness how engineering actors can be pulled toward the goals of the overall modular system in addition to the goals of derivative product development projects. Thus, it is a further aim to change the behaviour of design engineers during modularisation transition.

6.2.1 Research setting and methodology

This chapter forms the first part of the Prescriptive Study of this research work. It gives an overall perspective on aspects that companies have to improve for successful modularisation transition. Each of these aspects has to be further deepened and adopted to the specific context of the transitioning company. Thus, it is rather the aim of this chapter to prescribe important factors that have to be in place than prescribing detailed support methods or tools for each aspect (see Section 6.3). During this chapter, detailed means of support is rather mentioned as example than as prescription while the two coming chapters 7 and 8 will focus on two specific and detailed means of support that are only a small part of the overall perspective to consider by a transitioning company.

The input for the modularisation assessment framework strongly evolved from the qualitative study in industry described in the previous chapter. The framework was developed in three steps. Firstly, potential support was identified in literature and in the case study in industry. Secondly, the most promising support from the benchmark study was applied and tested to a large extend in the primary case company. Thirdly, the refined understanding was framed into an assessment framework which was validated in industry for its applicability and its relevancy as appropriate means of support for modularisation transition.

Consequently, the success criteria for this chapter are as follows:

- The modularisation assessment framework assesses factors that are indeed relevant for industry to be considered during modularisation transition.
- The modularisation assessment is suitable for its application in industry and offers new understanding for industrial practitioners during its application.

Due to reasons of validation, it is no success criteria of this chapter to directly measure whether companies indeed perform better during and after modularisation transition by applying the assessment framework than without its application.

6.2.2 Chapter elements

As shown in Figure 60, Section 6.1 presents existing assessment frameworks in literature and industry. This Section 6.2 gives a brief overview of the modularisation assessment framework chapter. The next Section 6.3 will move on to clarify some formal aspects for the assessment framework. Section 6.4 will establish an assessment scheme by bringing factors to be assessed into the context of a proposed modular system development life cycle. Section 6.5 is the core of this chapter and presents the actual content of the modularisation assessment framework. The following section 6.6 will give some hints on application before Section 6.7 will present the validation of the framework. This chapter will conclude with a discussion and brief summary in Sections 6.8 and 6.9 respectively.

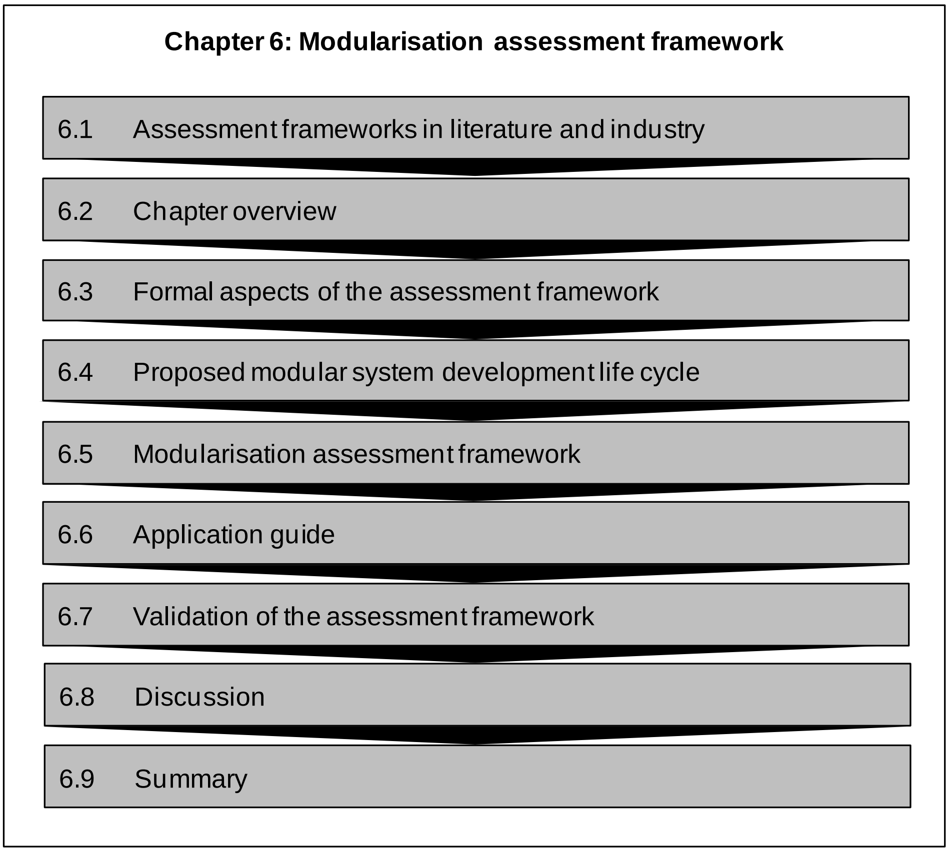


Figure 60: Elements of chapter 6

6.3 Formal aspects of the modularisation assessment framework

This proposed framework is to be applicable to organisations in different industries. Acknowledging varying needs, possibilities and circumstances of different companies, the modularisation assessment framework has been designed in order to provide flexibility to its users. The questions do not directly aim at detailed support, but rather at important factors that a company has to consider during modularisation transition. Therefore, every company applying the assessment framework has to decide how it addresses a certain assessment question, i.e. an important factor, in detail. In each case, this modularisation assessment framework is applied; a detailed company-specific improvement plan will be the result.

The following sections will not go into detail how the assessment framework can be applied, how its rating might be done or how a tool has been applied to collect questions. The focus of the next sections will be on the content that is targeted at for modularisation transition. Nevertheless, some hints for application of the assessment framework will be provided in Section 6.6.

Each development life cycle phase contains important factors that are assessed by questions of the assessment framework (e.g. see Table 18). It is proposed that these are the factors to be fulfilled by companies for successful modularisation transition. Thus, the assessment framework contains following elements which are given for each factor throughout the life cycle:

- a question which directly assesses the factor
- further supportive information:
  - a description of the factor
  - requirements to be fulfilled in order to help companies identifying fields of improvement
  - example (where applicable)
  - supporting comment (where applicable)
  - cause-effect relationship between issues (see Section 5.3) to be tackled by the question and support elements of the support framework (see Section 5.5)

The next section will introduce the development life cycle in detail that has been used to shape the assessment framework of this research.

6.4 Proposed modular system development life cycle

The basic principle of the presented modular system development life cycle has its origin in the benchmark study in industry (see Section 5.2). However, it has been refined for the purpose of this work in order to overcome the limitations of existing development life cycle models (see Section 5.4.1 of the Qualitative Study). Thus, similar process representations have been used already in Figure 52, Figure 53, Figure 54 and Figure 57 during the Qualitative Study, but it has been provided with more details in this chapter. For example, gates (see (Cooper, 2014)) have been added to the life cycle that allow practitioners to incorporate assessments prior to proceeding to the next phase. Moreover, some processes have been added in order to being able to better illustrate factors that are assessed by the framework (see Figure 61).

Figure 61 shows the phases of the modular system development life cycle, its interplay with derivative product development projects and the development of modules within existing products or as parallel but separate module development projects. In addition, the figure shows gates or milestones that are used to place "checkpoints" for the assessment of each phase of modularisation transition within this work.

The modular system life cycle starts with a pre-study phase which deals with the business potential of the modular system and whether the organisation can be convinced to invest into the modular system. If this phase is conducted successfully, it ends with a clear commitment for the modular system and results in the start of the overall development project. The next phase is the market phase and deals with the market study and requirements engineering for the entire modular system. The results of this phase are confirmed and well-accepted modular system requirements that can be passed on to concept development. Concept development phase translates requirements into the architecture and the plans for the modular system. This means that specifications for modules, interfaces and products which are later passed on to derivative development projects are generated. The result of this phase is a confirmed modular system concept. Until the end of this phase, the whole modular system is only on "paper". Therefore, the next phase was established to test and refine the modular system concept based on real simulations and artefacts. The result is a modular system *concept* that has been proved to be feasible.

Once the feasibility of the modular system has at least been preliminary tested, the modular system specification can be handed over to derivative product and module development projects. Thus, the succeeding phase mainly deals with the evolution and the change of the modular system. The main purpose of the modular system life cycle during this phase is fourfold:

- to make sure that a working modular system is indeed realised by derivative development
- to update specifications for the modular system
- to ensure that derivative development indeed adheres to the specification of the modular system
- to provide reusable modules for the usage by derivative product development projects

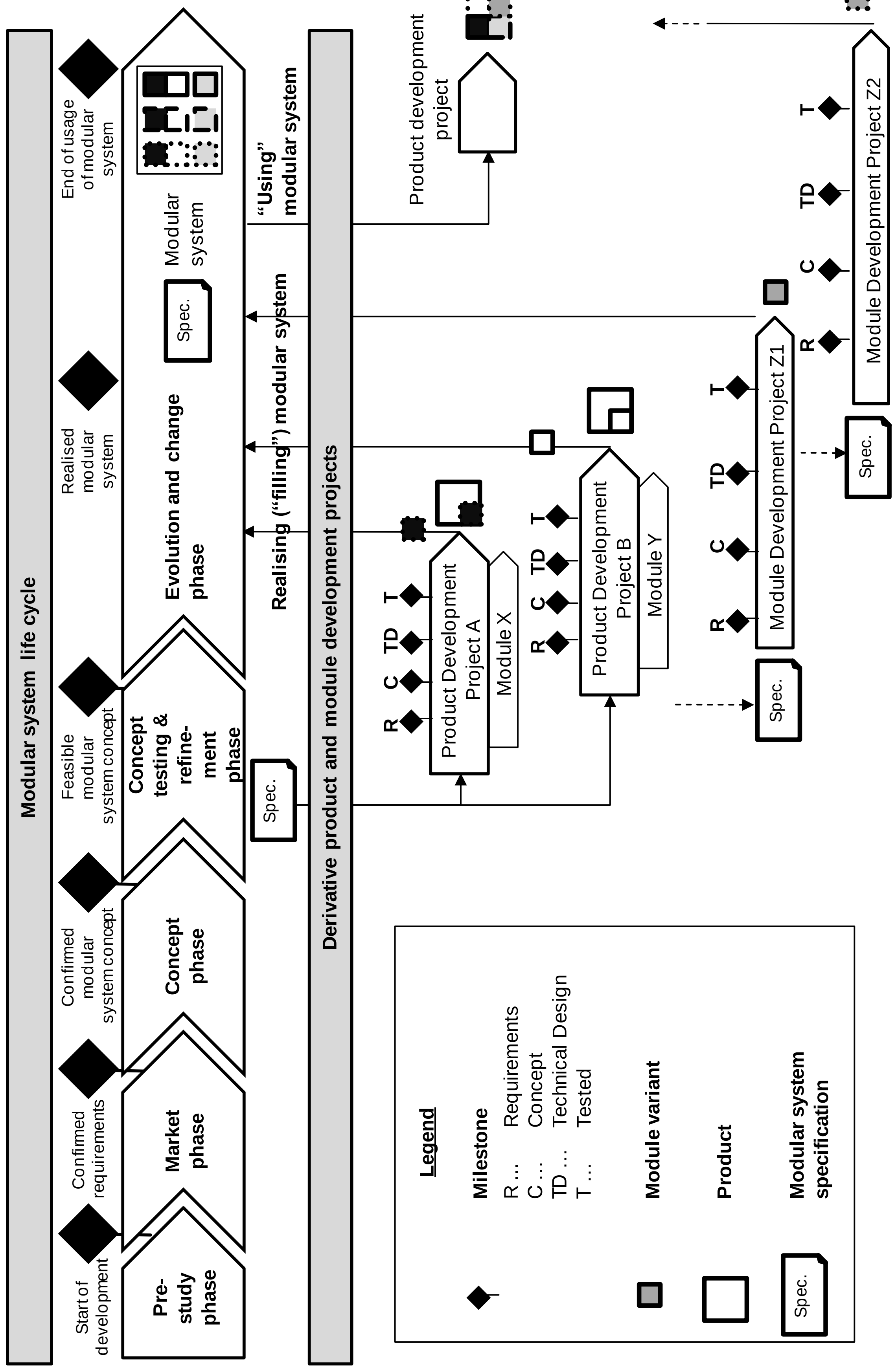


Figure 61: The proposed modular system development life cycle, derivative product development and life cycle milestones

It gets evident from the right hand side of Figure 61 that while transitioning within existing products in the phase evolution and change of the modular system life cycle, derivative development projects get the main actors for realising ("filling") and "using" the modular system. The development of module variants for the modular system takes place either during product development projects with dedicated module development (i.e. mixed projects) or during pure module development projects. Thus, there is a constant interplay between the modular system life cycle (see upper part of Figure 61) and derivative product and module development projects (see lower part of Figure 61).

Modular system specifications from the modular system life cycle are the input for derivative product and module development projects. However, the qualitative study (see Chapter 5) identified that it has to be ensured that development phases of these projects are indeed in line with these specifications in order to keep the stability of the modular system. This can be provided by establishing certain points of assessment like milestones after requirements engineering (R), concept definition (C), technical design (TD) or testing (T), (see Figure 61). Each deviation from the modular system specification or each request to change this specification has to be closely coordinated with advocacy of the modular system and those advocacies in local projects that might be impacted by such a change.

In consequence, by emphasising the phase evolution and change of the modular system life cycle, the assessment framework will contribute to pull local projects toward the specification of the modular system and to make later phases less prone to failure like it has been shown to have happened in various preceding undertakings (see Sections 5.1 and 5.3 of the Qualitative Analysis).

Even though, the modular system life cycle presented within this section has been described in a neat and sequential manner, it has to be stated that this is just a model of the development life cycle. Thus, awareness has to be risen that in practice there are much more overlapping and iterative phases. Moreover, companies are encouraged to adapt the model to their specific needs. For instance, it could make sense to run the definition and test of the modular system concept in parallel.

6.5 Modularisation assessment framework

The previous section described different elements of the modular system life cycle. This section now moves on to explain how important factors of each life cycle phase for the transition from single product development toward modular system development can be assessed. Therefore, this section has been divided into the different phases of the modular system life cycle (see Figure 61).

6.5.1 Pre-study phase – modular system life cycle

The following questions can be used to assess the pre-study phase of the modular system life cycle.

Has a common understanding about the "vocabulary" and the vision of what to achieve with the modular system been established?

Description:

It is the purpose of this point to establish a common mindset about transitioning toward modular system development.

Firstly, modularisation has been used in a large variety of different contexts and, thus, the same terms have been used to define different things in industry. Even worse, even across different departments of the same organisation, the terms in the field of modularisation are likely to be mixed up and used to express an individual's opinion. In order that it is clear to everyone in the organisation what is meant by transitioning toward modular systems, the company has to establish and communicate a clear vocabulary of what it means with a modular system.

Secondly, modular systems have been used and can be used to achieve different objectives like better serviceability or upgradeability. It is the purpose of this work to introduce modular systems with the main purpose of reducing complexity through reusing module variants that should ideally be common. Thus, a clear vision has to be established about the aims of the future modular system. Furthermore, it is important that the company clearly shows to its employees the difference between previously established "modular systems" and future modular systems to be developed.

Requirements:

Following requirements can be seen as relevant for this question. It is important that there is agreement across departments and development sites on these points.

- Different sites and departments define their "modular artefacts" according to the common company definition.
- Modules are built for a designated strategic or organisational purpose and can be distinguished from ordinary assemblies, parts and components.
- A module is built for a defined modular system and not for a single product or development project.
- A module is categorised into "common", "variant" or "optional", depending on the intended scope of products for its application.
- An *abstract* module contains a defined set of designated *concrete* module variants.
- Module variants have to comply with module specifications and other organisational rules.
- A module or rather its module variants have standardised and managed interfaces with a module boundary that is aligned to other modules.

Example:

In order to implement this factor, a series of coordination meetings were held in the primary case company (see Section 4.3). Moreover, agreed definitions and vision were made centrally available and communicated.

Cause-effect relationship:

The issues (see Section 5.3 or Figure 55) that are sought to be removed and the related aspect of support (see Section 5.5 or Figure 57) which are addressed by the factor in this question are as follows:

Addressed issues	Related support aspects:
Lack of information and transparency	Process: pre-architecting and architecting phase

Establishing a common definition and vision about the modular system at the beginning of its life cycle shall help to increase information and transparency among involved employees.

Is there an agreed understanding within the organisation about the investments to be made and the benefits to be expected?

Description:

Once it is clear to everyone what it exactly means to transition toward modular system development, the potential of such an undertaking has to be analysed. Specifically, this means that the costs and benefits have to be estimated and agreed by involved roles. It has to be clear that *not* transitioning toward modular system development could also be an option that could be perfectly justified. The main purpose of this question is to make sure that the huge investment that has to be made at the beginning is well-justified. This is especially important because the benefits of modularisation cannot be realised before one or several product development life cycles have been passed through. During such a "drought" between high initial investment and delayed benefits, a detailed and agreed potential analysis is seen as important enabler to keep the motivation and focus of involved engineers on the overall modular system, instead of falling back onto local development focus. In any case, it has to be expected that there are phases where single product solutions seem to be more appealing. For such situations, a strong advocate for the modular system in the form of a potential analysis can be crucial. In the end, it must be officially agreed by all involved roles that a modular system is the best option to be pursued, despite major drawbacks to be expected during its development. Thus, the agreed potential of the system can be seen as main justification and motivation that keeps people going during the "drought" period.

Requirements:

Following actions suggested to be in place can be seen as relevant in order to give a satisfying answer to this question. In any case, it is important that there is agreement across departments and development sites on these points. Moreover, commitment of top management is crucial.

- Agreed potential analysis that demonstrates investments and benefits to be expected from the modular system.
- Benchmark partners or pilot project as some kind of flagship initiative.

Example:

This example shows how the potential of a modular system can be calculated. The example has its origin in a real case from industry and has been applied in the primary case company (see Section 4.3), but has been disguised for the purpose of this work.

- *Potential complexity reduction per business unit*

The analysis started by counting distinct part numbers in the bill of materials of all existing product families under scope of the future modular system. This can be seen as the baseline with which a future modular system can be compared with.

Subsequently, a feature list with all external market characteristics that the new modular system should fulfil was set up. Then, the draft of a modular system which is capable to fulfil the external features was set up. This was assured by relating the modules and their module variants to the features in a typical QFD matrix. The basic procedure for the determination of the sketch of the new modular system is depicted in Figure 62.

After it was assured that the modular system is capable to fulfil the features demanded by the market, the part number count for each module was estimated by experienced engineers. For instance, the engineers had to determine how many pressure sensors they had in their business units as part numbers in IT-Systems. During discussions, they estimated how many they will really need in future if they use an improved product architecture. The estimated part numbers for each module could afterwards be totalised for the whole modular system.

The difference between the existing part number count per platform and estimated future part number count gave the potential in terms of complexity reduction for the new modular system. For correct estimations, it was necessary to only take into account those features that can be fulfilled by the old platform and by the new modular system. For instance, if the new platform was much more complex in its features than the old platform, modules contributing to the distinct features might not have been considered for the comparison. Another possibility to "normalise" features would have been to include all features of the old platform and the new modular system into the comparison and relating complexity directly to respective features.

		Type			Fuel		Power			...	Distinct part numbers
		Roadster	Cabrio	Coupé	Petrol	Diesel	55 kW	90 kW	135 kW		
Door-Module	Door 1	X									157
	Door 2		X								+ 38
	Door 3			X							+ 54
	Total										249
Engine - Module	Engine 1				X		X				2000
	Engine 2					X		X			+ 150
	Engine 3				X				X		+ 380
	Total										2.530
...											
Total platform											15.690

Figure 62: Relation of features to modules and estimation of part numbers per module and per modular system (platform)

The final result of this procedure was the potential part complexity reduction in terms of part number count for each business unit.

From the technical side, the study showed significant and impressive potential by improving and unifying the product architecture across the company. The estimated part number count reduction potential ranged between 10 % and 65 %, depending on the product platform and the business unit.

- *Calculating complexity costs*

In a parallel project, complexity costs per part number were determined. In this project, the cost for an activity incurred by a new part number was calculated. This was done in a classical activity-based costing approach with the part number as cost driver in company functions like development, production, purchasing, logistics, production, sales, and service. A disguised example for some main complexity cost drivers along the life cycle in the case company is given in Figure 63. *The underlying concept of the complexity cost approach is described in Section 7.2.4 "Assessment of monetary and performance implications". Moreover, Sections 7.2.4 and 1.2 reference detailed complexity cost studies. The monetary potential of the new modular system compared to the old platform could be calculated by multiplying the potential part number count reduction by the complexity cost per part number.*

The concrete analyses from the sample indicated that complexity costs for simple and small parts are at several hundred Euros and costs for huge and complex parts are at several ten thousand Euros. The reason for that difference is, for example, that a new cylinder block *variant causes much more inner company activities (e.g. drawings, tooling, loading equipment, storage space, supplier negotiations) than an additional resistor from the catalogue.*

The impressive potential impact of a new modular system becomes clear when exemplified figures are multiplied. For instance, if a product architecture alternative for a new modular system enables to save just 1000 part numbers compared to a conventional platform, the saving potential across the company, especially in indirect areas, could be between several hundred thousand of Euros and several ten million Euros.

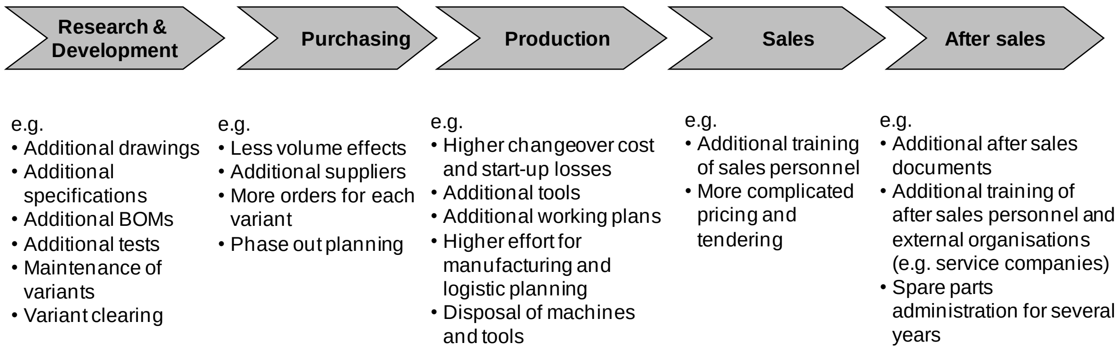


Figure 63: Disguised example of some main complexity cost drivers along the life cycle in the case company

However, the potential savings were not for free. The costs for making the shift toward modular reference architectures had to be considered as well. In any case, significant implementation costs like training costs, setting up additional teams or costs for change process accumulated and had to be considered in the calculation as well.

Cost-benefit calculations with realistic scenarios within the case company showed that the estimated benefits were more than two times higher than the costs the company had to expend by implementing such a programme.

Even though, no concrete figures but only estimations have been provided by the analysis, the procedure for this potential analysis was found to be an appropriate approach for the purpose of this study by involved engineers and engineering managers. The analysis was commonly agreed during management presentations that it delivered the expected results for such an early pre-study phase. This acceptance represented a strong motivation to pursue the modular system over a prolonged period.

Cause-effect relationship:

The issues (see Section 5.3 or Figure 55) that are sought to be removed and the related aspect of support (see Section 5.5 or Figure 57) which are addressed by the factor in this question are as follows:

Addressed issues	Related support aspects:
Lack of intrinsic and extrinsic motivation	Process: pre-architecting and architecting phase Modularisation evaluation

Ensuring that a potential analysis for the modular system is carried out at the beginning of the modular system life cycle helps to boost intrinsic and extrinsic motivation of involved employees during later phases.

Are initial investments and capacity for the required frontloading indeed provided?

Description:

The qualitative study (part of the work in Chapter 5) revealed that there were cases with a strong declared intention to pursue a modular system strategy. However, companies failed to provide additional budget for modularisation activities that do not amortise in the short term. In consequence, projects tried to modularise, but only had the resources to pursue short-term goals of the underlying product development project. Modularisation has to be done with the right commitment and with stringency from the beginning. If this is not done, it is prone to failure during later phases. Therefore, companies have to provide budget for modularisation transition activities which are uncoupled from traditional project resources.

Requirements:

Following points should be provided by the company in order to give a satisfying answer to this question.

- Dedicated budget for modularisation transition activities, organisational changes, additional roles and changed machinery, particularly during transitioning
- Appropriate budget and commitment to withstand negative side-effects of modularisation like higher direct material cost due to overdimensioning
- Engineers that have at least been partially freed up from project development tasks in order to appropriately devote their time to modular system development

Example:

The primary case company set up a dedicated team to plan and implement overarching activities that have to be conducted in the course of modularisation transition.

Cause-effect relationship:

The issues (see Section 5.3 or Figure 55) that are sought to be removed and the related aspect of support (see Section 5.5 or Figure 57) which are addressed by the factor in this question are as follows:

Addressed issues	Related support aspects:
Lack of time and resources	Process: pre-architecting and architecting phase

Assigning dedicated resources to modularisation activities helps to remedy lack of time and resources for involved employees.

How is the overall organisation adapted to the transition toward modular system development?

Description:

Modularisation transition is an undertaking that has consequences in almost all functional departments of a company. On the other hand, in order to make a smooth transition, companies have to change a wide range of aspects and the way the company is working. Therefore, each area of the company must be scrutinised on its capability to work for the modular system instead of solely for single projects.

Requirements:

All of the following aspects should be considered by the company for modularisation transition:

- Separating the modular system development life cycle and derivative product development life cycle with an appropriate emphasis on the modular system
- Rearranging the organisation so that it reflects the architecture of the modular system
- The IT-systems have to be adapted so that they represent the architecture of the modular systems
- Change of how development projects are evaluated and monitored

Cause-effect relationship:

The issues (see Section 5.3 or Figure 55) that are sought to be removed and the related aspect of support (see Section 5.5 or Figure 57) which are addressed by the factor in this question are as follows:

Addressed issues	Related support aspects:
Lack of time and resources Lack of intrinsic and extrinsic motivation Lack of information and transparency	Process: pre-architecting and architecting phase Process: post-architecting phase Modularisation organisation Modularisation evaluation Modularisation information

It is suggested that only if a wide-ranging set of aspects is amended to modular system development, this will improve resources, motivation and information of involved employees.

Is the transition toward modular systems constantly monitored with the involvement of top management?

<u>Description:</u>

The constant commitment of top management for the modular system concept is important to be maintained over several years during the "draught" transitioning period. Because the modular system will not yield "profits" during this period, but will be attacked from various players in favour for local solution, it is important that higher management sticks to its made decision and intervenes in favour of the modular system programme whenever appropriate.

<u>Requirements:</u>

The organisation ought to implement following measures in order to satisfactorily answer this question:

- Regular reporting of the programme to attentive and involved managers
- Roadmaps and visual dashboards to detect deviations from plan during transitioning
- Interventions from top management in case of deviations

<u>*Example:*</u>

The primary case company set up modularisation roadmap which was regularly reported to management. The roadmap included the modularisation decision process, training activities, building up experts, process integration, organisational integration, implementation of regular meetings for local experts, pilot projects, operational ramp-up activities for development projects, implementation of key performance indicators and achievement of modularisation targets.

Cause-effect relationship:

The issues (see Section 5.3 or Figure 55) that are sought to be removed and the related aspect of support (see Section 5.5 or Figure 57) which are addressed by the factor in this question are as follows:

Addressed issues	Related support aspects:
Lack of intrinsic and extrinsic motivation	Modularisation evaluation

It is suggested that only if implementation activities of modularisation are evaluated, the motivation of involved employees will be sustainably improved.

Is there a clear distinction between a central modular system development process and derivative product development processes?

Description:

Modular systems which scope multiple development projects cannot be only developed by single product development projects themselves due to their limited resources and local goals. However, development process descriptions of companies most often only give advice on "one of a kind" solutions. Therefore, it is important that the process landscape of the company reflects a process for the modular system and a process for derivative products. Development processes have to be built in a way so that artefacts for a modular system are developed for a broad solution portfolio in scope and not only for single projects.

Requirements:

Within the processes involved in modular system development, there should be following streams separately identifiable (see Figure 61):

- Process stream for overarching modular system development activities
- Development of module variants
- Development of derivative products that are derived from the modular system

Example:

The primary case company installed a project called "process integration of modularisation". The project had two parts: a) adapting and introducing modularisation-relevant activities and work products in derivative development projects and b) introducing an overarching modular system development process. Other activities like integration into supply chain processes had to be considered as well, but they are beyond the scope of this work.

Cause-effect relationship:

The issues (see Section 5.3 or Figure 55) that are sought to be removed and the related aspect of support (see Section 5.5 or Figure 57) which are addressed by the factor in this question are as follows:

Addressed issues	Related support aspects:
Lack of time and resources Lack of intrinsic and extrinsic motivation	Process: pre-architecting and architecting phase Process: post-architecting phase

Time, resources and motivation to work for the modular system are substantially increased if the company installs processes which are dedicated to the overarching modular system.

6.5.2 Requirements phase – modular system life cycle

After a clear decision for the modular system has been made and after planning and committing required changes within the organisation, the team can start to define requirements for the modular system.

How is it ensured that that the required deep market knowledge for the overall modular system is available in time?

Description:

The difference to single product development is that modular system requirements affect a much wider variety of future products which gives a greater impact when changing them. The requirements phase has to contribute to a stable product architecture. Therefore, it has to be ensured that the requirements which are linked to the product architecture and which are later passed on to products and modules are as stable as possible. Although it is obvious that requirements will always be changing in a volatile market environment, this change has to be limited to those requirements that are affected by changing customer or regulative demands. Change of requirements because of lack of knowledge during front-end phases has to be eliminated during modular system development. Thus, it is necessary to get as much knowledge as possible during this phase as it has to be clear that later changes can only be realised by changing a module variant and only hardly by changing the modular system or architecture.

Requirements:

In order to satisfy above mentioned demands of this phase, following requirements have to be fulfilled:

- Agreed strategic positioning of all products under scope
- Definition of success factors, unique selling points and differentiation to competitors
- Well-grounded market segmentation
- Market research, market analyses, use cases and customer involvement

Example:

The methods and tools that have been used in companies identified during the qualitative analysis have been for example primary & secondary market research, SWOT analysis, Porter's Five Forces Analysis, User Experience, Quality Function Deployment and other methods from this field. Chapter 3, particularly Section 3.1.5, and Appendix A give further advice on modularisation literature dealing with the input of market information for product architecting.

Cause-effect relationship:

The issues (see Section 5.3 or Figure 55) that are sought to be removed and the related aspect of support (see Section 5.5 or Figure 57) which are addressed by the factor in this question are as follows:

Addressed issues	Related support aspects:
Lack of information and transparency Lack of time and resources	Process: pre-architecting and architecting phase

Involved employees gain significant information and transparency to fix the requirements of the modular system if the company installs a dedicated phase and resources for extended requirements engineering.

Is the selection of requirements restrictive, well-justified, documented and is the focus only on those requirements that promise profitability?

Description:

Usually products are adapted to the customer requirements that have been derived in the course of the related development project. However, in the case of building a modular system, it is necessary to define architecture-relevant requirements prior to product development. For this reason, product management could be inclined to raise as much requirements as possible during the pre-architecting phase in order not to be restricted in choice during product development later on. Even though this might be beneficial for product management and possible during single product development, this is not possible during modularisation with the goal of a stable modular reference architecture. Too many accommodated requirements would lead to an architecture that is vulnerable to change and to a modular system that is overly costly and like an "all-in-one device suitable for every purpose" which can do everything, but nothing good. Thus, the number of requirements has to be limited based on their potential for profitability. Other researcher call this activity "reducing variety in product solution spaces" (Haug, Hvam and Mortensen, 2013).

Requirements:

- Quantified decision for most relevant requirements or features (e.g. through their sales potential and contribution to the product's value)

- Documentation of the final modular system requirements specification and the way it evolved (e.g. documented reasons when requirements were removed)
- Common agreement on final modular system requirements specification by different company functions, product development projects and top management

Example:

The primary case company used several different methods from variant management (e.g. interdisciplinary analysis of variant drivers, transparency through feature trees, impact analyses and target agreements for variants) to determine the variance to be derived from the modular system (for further examples see Alders (2006, p. 229–231), Wildemann (2005), Schuh (2005) or the methods of Section 3.1 and Appendix A with input factors from the market phase). Moreover, an extended requirements specification was introduced for the purposes of modular system development within the primary case company.

Cause-effect relationship:

The issues (see Section 5.3 or Figure 55) that are sought to be removed and the related aspect of support (see Section 5.5 or Figure 57) which are addressed by the factor in this question are as follows:

Addressed issues	Related support aspects:
Lack of time and resources (to build a "one fits all" system)	Process: pre-architecting and architecting phase

In order to avoid overwhelming engineers with unmanageable requirements, there has to be a dedicated process phase for restriction of requirements and solution spaces.

How is traceability between market input, requirements and the common modular reference architecture established?

Description:

The impact of changing conditions (e.g. market environment), requirements or items (e.g. a part, interface or a module variant) has a much higher impact than within single product development due to the much broader scope of the modular system. Hence, making the impact of changes to the modular system or to underlying requirements transparent is a precondition to effectively control the modular product architecture during its life cycle.

Requirements:

- Available traceability information between fundamental market data and requirements of the modular system
- Available traceability information between requirements and the items of the modular system (e.g. module variants, parts, interfaces, products)

<u>Examples:</u>

The qualitative analysis of Chapter 5 showed that different companies had totally different means of how they provide information about traceability of data and artefacts. Examples could be found in the form of spreadsheets, individual database solutions or commercial solutions like IBM Rational DOORS. Thus, the solution should be chosen based on the specific situation.

<u>Cause-effect relationship:</u>

The issues (see Section 5.3 or Figure 55) that are sought to be removed and the related aspect of support (see Section 5.5 or Figure 57) which are addressed by the factor in this question are as follows:

Addressed issues	Related support aspects:
Lack of information and transparency	Process (complete)

In order to gain information and transparency companies have to provide and maintain traceability throughout the modular system development life cycle.

6.5.3 Concept phase – modular system life cycle

How is the common modular reference architecture established?

<u>Description:</u>

It is important that the modular system covers all requirements and products under scope from earlier life cycle phases. The common modular reference architecture is the linchpin of the modular system to create commonality and variability. For this reason, those elements that should ideally be common and those elements that can be used to generate variety should ideally be grouped into modules. In order to make profound decisions about this, the technical concept for each module should be decided. Moreover, the architecture should comprise standardised interfaces and independent modules in order to be resistant to change from the environment but to be flexible to create variety by combining modules. Besides these main purposes of the architecture, other factors influencing or being influenced by the architecture have to be considered as well. For instance, the architecture might influence the performance, recyclability or serviceability of a product. As a consequence, the architecture has to be systematically designed according to well-founded principles.

<u>Requirements:</u>

- Team of interdisciplinary experts with appropriate resources contributing to architecting
- Technical knowledge and agreement on technical concept for the realisation of functionality for each module

- Systematic module clustering of the entire modular system based on business goals like strategic factors, interfaces or product functions
- Definition of elements of product architecture, i.e. interfaces, modules, module boundaries and classification of modules
- Output of this step is made explicit (e.g. documentation in graphs or spreadsheets)

Supporting comment:

The literature review in Chapter 3, Appendix A and the Qualitative Study in Chapter 5.4 gives some insights on systematic methods for product architecting and their usefulness in practice.

Cause-effect relationship:

The issues (see Section 5.3 or Figure 55) that are sought to be removed and the related aspect of support (see Section 5.5 or Figure 57) which are addressed by the factor in this question are as follows:

Addressed issues	Related support aspects:
Lack of information and transparency	Process: architecting phase
Lack of time and resources	

A dedicated process phase for this activity helps to provide time and resources for extended architecting and making the architecture explicit. Moreover, it supports engineers in gaining transparency and information about the technical concept, the overall architecture and interfaces to be expected.

How are the artefacts of the modular system established and planned?

Description:

After the common modular reference architecture and its elements are defined, it is important to agree upon their level of variety and to plan their actual implementation. This means that size range, number of variants and degree of standardisation of module variants and interfaces has to be determined. Moreover, it has to be analysed how the development of these variants matches the roadmap of products intended to be developed.

Requirements:

- Appropriate time for technical and market roles to discuss and agree realisation of the modular system
- Number of module variants to be developed for each module (e.g. documented in a module variant matrix and module roadmap)
- Alignment between product and module development (e.g. documented in a product-module matrix)

- Technical and market specification for each module variant agreed to be developed
- Combination restrictions between modules (e.g. based on variant trees)
- Plans for the modular architecture are made explicit
- Plans of the modular system are in line with complexity goals of the company

Example:

The primary case company used several different methods from variant management (e.g. variant trees, configuration matrices or interdisciplinary impact analyses of variant scenarios) to plan the variants of the modular system (for further examples see Wildemann (2007), Alders (2006, p. 229–231), Wildemann (2005), Schuh (2005), Design for Variety within the PKT-Approach (Krause et al., 2014) or the holistic methods of Section 3.1.5 and Appendix A).

Cause-effect relationship:

The issues (see Section 5.3 or Figure 55) that are sought to be removed and the related aspect of support (see Section 5.5 or Figure 57) which are addressed by the factor in this question are as follows:

Addressed issues	Related support aspects:
Lack of information and transparency	Process: architecting phase
Lack of time and resources	

It is suggested that a dedicated process phase for this activity helps to free up appropriate time and resources and that explicit plans for the modular architecture help to improve information and transparency.

6.5.4 Concept testing and refinement phase – modular system life cycle

Prior to starting with the development of the modular system and after specifying the elements of the modular system, it is important to test whether the modular concept is indeed working.

How is the feasibility of the common modular reference architecture which covers a broad scope of parallel and future products proven?

Description:

When designing a new system, there are always many uncertainties. In the case of single product development, these uncertainties can be removed by fully testing uncertain aspects. However, this is not possible during modular system development where most of the products are not directly built after concept phase, but maybe several years ahead. Moreover, there is the broad scope of the modular system which makes a full design and test of all products almost impossible. However, the modular system in this life cycle phase is still a theoretical concept on paper. Therefore, at least the underlying concept of the modular system (e.g. interface specifications, combinability of modules) has to be tested and whenever necessary refined.

Requirements:

- Dedicated process phase for architectural feasibility studies.
- Reduction of uncertainties which could not be removed during preceding phases.
- Development of sample concept products to demonstrate “proof of concept” of the modular system, but also to test the reaction of the customer.
- A few well-selected sample module variants should be built and tested from a technical perspective in order to demonstrate combinability of module variants.
- Fixed architecture specifications for the modular system to be handed over to derivative development projects.

Example:

An example from the primary case company for requirements of samples for the modular system is as follows:

- *Product architecture related requirements and specifications (e.g. interface specifications) are the input for building conceptual samples of the modular system. Such samples might be built in the sample department of the company and may incorporate dummies if this has no negative effect on the demonstration of feasibility.*
- *Concept testing products are no products that have necessarily full functionality implemented. Rather, they are used to demonstrate and confirm arrangement of module variants and components within the given boundaries of a product and its internal interfaces.*
- *Concept testing modules which have not been developed so far, have at least to demonstrate the feasibility of their space envelope and their interface specification. Moreover, arrangement of components within modules has to be demonstrated.*
- *The agreement on technical feasibility and on whether sample products have been accepted as appealing to customers, will be the trigger to fix architecture specification for the handover to derivative development.*

Cause-effect relationship:

The issues (see Section 5.3 or Figure 55) that are sought to be removed and the related aspect of support (see Section 5.5 or Figure 57) which are addressed by the factor in this question are as follows:

Addressed issues	Related support aspects:
Lack of information and transparency	Process: post-architecting phase
Lack of time and resources	

It is claimed that a dedicated process phase to test and refine the modular system helps to increase time for gaining transparency about the feasibility of the modular system concept.

What is the content of the refined architecture specification that will be passed on to derivative development projects?

Description:

Local development projects usually optimise their products based on the given requirements of the underlying project. However, during modular system development, projects also have to focus on products or modules which are to be developed in separate projects. The requirements to synchronise the interplay between different development projects have their origin in fixed architecture specifications. Thus, it is the purpose of this phase to ensure that these architecture specifications are mature enough to be handed over and that their binding content makes derivative projects developing in accordance with the common modular reference architecture.

Requirements:

In order that derivative product development has a stable base for the development of the modular system, following specifications have to be provided by central modular system development:

- Detailed roadmap of products to be derived from the modular system
- Overview of module variants to be developed for each module in order to demonstrate the agreed level of variety or standardisation
- Detailed roadmap of module variants and add-ons to be developed, synchronised with the product roadmap (e.g. through a product-module dependence matrix)
- Requirements specification of the modular system which has been broken down to product and module variant level
- Interface and module specification with description of module boundaries

Cause-effect relationship:

The issues (see Section 5.3 or Figure 55) that are sought to be removed and the related aspect of support (see Section 5.5 or Figure 57) which are addressed by the factor in this question are as follows:

Addressed issues	Related support aspects:
Lack of information and transparency Lack of time and resources	Process: post-architecting phase

The modular system development process shall make sure during this phase that enough time and resources are provided to create architecture specifications which improve information and transparency in derivative development projects.

6.5.5 Evolution and change phase – modular system life cycle

From this phase on, it is assumed that module and product development projects can proceed to develop artefacts of the modular system, based on above mentioned specification. However, it is assumed that there will be a considerable amount of internal and external change requests emerging so that architecture-relevant specifications have to be updated and the work between separate development projects be synchronised.

How are external changes like market characteristics, technology trends and requirements constantly controlled and implemented?

<u>Description:</u>

This point deals with externally induced changes on the product architecture and the control of its impact. It was introduced to this assessment framework because local development projects are usually closer to the change within their direct market. However, changes shall only be passed through from modular system development to derivative development in close collaboration, but without permitting any local or work-around solutions. Thus, the central architecture specification has to be appealing to local development projects in order to increase their motivation for usage. Therefore, the specification of the architecture has to be regularly maintained as architecture-relevant requirements are constantly passed through to derivative development projects. If changes are managed in a proactive manner, their impact on the product architecture can be assessed and upon provision of the stability of the architecture, the changes can be implemented. If the impact of the requested change is not in compliance with manageable architecture changes, there has to be an agreed decision whether to reject the change, to make major revisions to the architecture or to handle the change separately.

<u>Requirements:</u>

- Dedicated resources to analyse external developments and to analyse their impact on the architecture (e.g. monitoring of changes in requirements, target costs or new trends in technology).
- Central requirements change process for architecture-relevant requirements. There is a close collaboration with local development projects in order to “feel the beat of the market”, but the decision authority stays centrally in order to keep stability of the modular system.
- Changed requirements and specifications are passed through from central development toward impacted local development projects.

Cause-effect relationship:

The issues (see Section 5.3 or Figure 55) that are sought to be removed and the related aspect of support (see Section 5.5 or Figure 57) which are addressed by the factor in this question are as follows:

Addressed issues	Related support aspects:
Lack of information and transparency	Process: post-architecting phase
Lack of internal and external motivation	

A dedicated process phase for this activity shall increase motivation and time of employees to provide transparency about requirements on a modular system level instead of on a single product level.

How are internal changes on requirements and architecture related items managed?

Description:

Usually, changes to the modular system are broken down from modular system level to derivative development project level. However, there are situations where local development projects initiate changes, for example based on cost reduction programmes or DFA activities. Due to previously mentioned traceability information, the impact of such changes on the product architecture and the modular system can be analysed. If there is indeed such an impact, the change has to be controlled from a modular system perspective, i.e. from an authority in charge of the modular system (see question with modular system organisation of this assessment framework). Thus it is the purpose of this question to ensure that changes to the modular system like the common reference architecture, module variants or interfaces have to be strictly controlled from a central modular system perspective.

Requirements:

- Availability of a designated engineering change process for modular system relevant changes which controls change impacts with regards to:
  - Global complexity goals of the company
  - Sustaining the modular system as a functioning whole
  - Protecting the modular reference architecture against local goals in derivative projects
  - Compliance to modular system specifications
- Availability of a designated release process for new items of the modular system, particularly for module variants in order to control the creation of new variance. This process can also be a part of the above mentioned change process. The release process is supported by comparing the change request to specifications and plans. Changing

plans and specifications or releasing additional modules has to be well-justified by considering global goals of the company (e.g. through impact analyses on modular system goals and calculation of complexity costs). After further analyses, there are three possibilities how the request for a new variant can be processed: a) the request is rejected, b) the request is integrated into the modular system or c) there will be an additional variant or an unavoidable local solution for the derivative project.

- Maintenance of plans and specification of the modular system
- Communication of the change

Supporting comment:

Controlling that the specifications of the modular system are indeed met has to be verified directly within product and module development projects (see Section 6.5.5 and Section 6.5.6).

The change and release process for modular items can be supported with standard information technology (see Chapter 8). Moreover, it is recommended that the modularisation organisation is supportive to the engineering change process.

Examples:

It is recommended to adapt the existing engineering change process of the company to above mentioned requirements. This means that architecture-relevant changes have to be first identified before they are analysed and approved by a central modular system authority or department (Figure 64).

The example that was used in industry took use of classifying items of the modular system (see Section 8.3.2 of IT-Integration) and of a filter to assess the impact of the change. Such a filter can be assembled by taking use of methods and metrics that are used to establish and evaluate product architectures (Bahns, Gregor Beckmann, et al., 2015) like the enhanced change mode and effects analysis (CMEA) tool used by Keese et al. (2006). The literature sections of this research thesis can be used as further reference to the mentioned methods and metrics (see Section 3.1 for methods and Section 7.2 for metrics). Further examples from academia and industry for change and release processes are presented by Alblas and Wortman (2008), Bahns et al. (2015) and Schuh et al. (2013). However, the qualitative study (see Chapter 5) showed that approaches that try to exactly quantify effects of fuzzy impacts have not gained better acceptance in industry than expert opinion or an interdisciplinary change control board without formalised method. Moreover, unlike the examples from literature the study showed that the impact of changes also heavily depends on the impacts on single part level than solely on a superior module level. For instance, sometimes the impact of changing a single component within a module has a low impact, even though the module is classified as module which should not be changed over its life cycle.

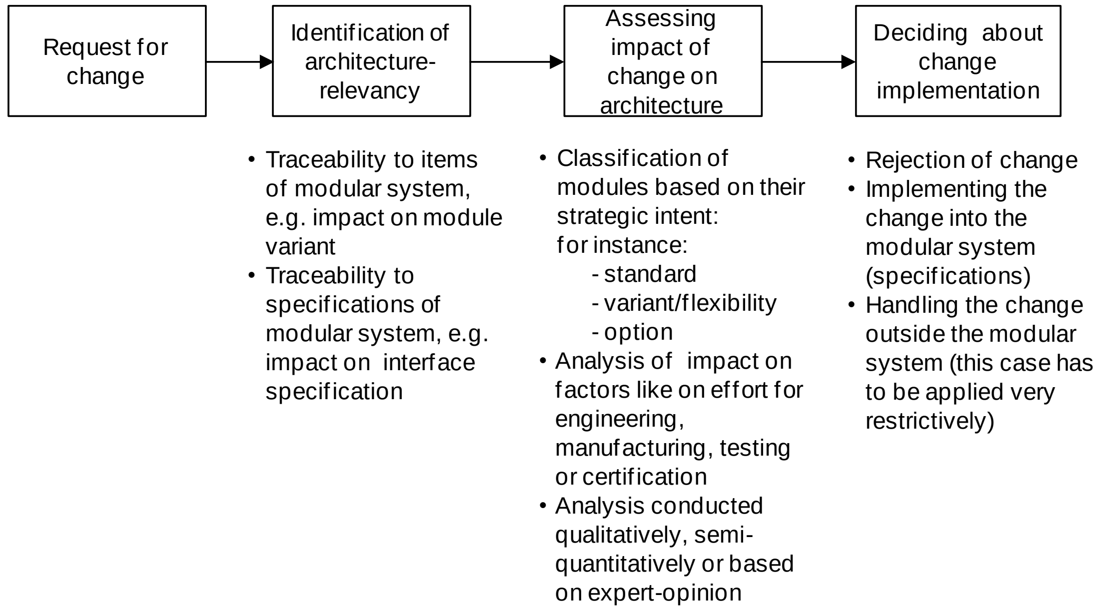


Figure 64: Example for a modular system change process

Cause-effect relationship:

The issues (see Section 5.3 or Figure 55) that are sought to be removed and the related aspect of support (see Section 5.5 or Figure 57) which are addressed by the factor in this question are as follows:

Addressed issues	Related support aspects:
Lack of external motivation	Process: post-architecting phase

A dedicated process phase to control changes to the modular system will impose external motivation to comply with the common modular reference architecture on engineering actors.

Is there a designated organisational structure that is in charge of modular system development?

Description:

At the latest after establishing the architecture and setting up plans for the realisation of the modular system, roles that are in charge of the modular system have to be established. It is important that these roles are assigned centrally and represent the interests of the modular system concerning its goals toward complexity and variety. Consequently, modular system roles represent the counterpart to roles that strive toward local goals in derivative development projects. How these roles are implemented into the organisation in detail depends on the character and situation of the company.

<u>Requirements:</u>

- Product management: Organisational responsibility to ensure that the modular system remains appealing to the market without an overload of unprofitable variety. The tasks comprise following activities:
    - Observation of the market environment for the whole modular system and initiating potential changes to the modular system
    - Maintenance of market data for the whole modular system
    - Maintenance of the requirements specification and product line roadmap for the modular system in coordination with engineering roles
    - Analysing and negotiating variance of modules and products to be developed with module and modular system roles
    - Monitoring and reporting market performance of the modular system (see Chapter 7 with modularisation metrics for further advice)
- Module engineering: Organisational responsibility to ensure that module variants are developed and used according to plans and specifications.
    - Definition and maintenance of module and interface specifications and module envelopes in coordination with modular system engineering
    - Maintenance of module roadmaps in coordination with modular system engineering and roles responsible for product roadmaps
    - Definition and maintenance of module variance in negotiation with product management
    - Assessing that modules are developed according to module and interface specifications
    - Authority for change process of affected modules
    - Monitoring and reporting target fulfilment of respective modules (see Chapter 7 with modularisation metrics for further advice)
- Modular system engineering: Organisational responsibility that balances the need between high variety and high reuse while ensuring that the modular system remains stable.
    - Coordinating definition and maintenance of product architecture plans and specifications
    - Breaking modular system requirements down to module level
    - Assessing that derivative products and modules are developed according to plans and specifications
    - Authority for change process for changes which affect the modular system

- Monitoring and reporting target fulfilment of modular system (see Chapter 7 with modularisation metrics for further advice)

Examples:

As mentioned above, the concrete implementation of the roles depends on the characteristics of the respective company. Therefore, different streams were found during the qualitative study:

- *Central modular system department which fulfils above mentioned tasks centrally and in close collaboration with the market and derivative development projects*
- *Modularisation roles which are assigned to functional departments or act as independent roles: e.g. the product manager from the product management department is assigned to the modular system, senior engineers are freed up to overtake the tasks of module engineering and a senior engineering manager is overtaking tasks of modular system engineering*

Other studies that have been validated in industry show that there is not one size fits all solution to this topic. While Kreimeyer (2014) favours a central product architecture department in truck industry, Bahns (2015) presents an independent modular system officer, Schuh et al. (2015) propose an overarching product architecture role, module roles and product roles and Arnoscht (2011) advocates a concept with independent "decentral" roles for modular system development.

Cause-effect relationship:

The issues (see Section 5.3 or Figure 55) that are sought to be removed and the related aspect of support (see Section 5.5 or Figure 57) which are addressed by the factor in this question are as follows:

Addressed issues:	Related support aspects:
Lack of time and resources	Modularisation organisation
Lack of internal and external motivation	

Establishing a modularisation organisation makes sure that there are roles with appropriate time, resources and motivation to pursue the goals of the common modular system.

How are organisational roles implemented and executed?

Description:

It is not sufficient to establish the right roles. It is necessary that these roles have the right targets, that they are equipped with appropriate power and resources and that they are involved in the right processes.

Requirements:

- Modularisation roles are senior experts in the corresponding field

- Roles act independently from derivative product development projects
- Roles possess adequate resources and "power" to fulfil their tasks. The balance of power between modular system development and derivative product development should be in favour of modular system roles.
- The target agreement, possibly incentives, of modularisation roles are linked to the targets of modular system development
- Reporting of modularisation roles has a link to top management of the company

Cause-effect relationship:

The issues (see Section 5.3 or Figure 55) that are sought to be removed and the related aspect of support (see Section 5.5 or Figure 57) which are addressed by the factor in this question are as follows:

Addressed issues:	Related support aspects:
Lack of time and resources	Modularisation organisation
Lack of internal and external motivation	

Time, resources and motivation of the modularisation organisation will substantially increase if above mentioned requirements are met.

How is the performance of the modular system assessed during transitioning?

Description:

When a company transitions from single product development toward the development of modular systems, its performance measurement is usually aligned to local goals of product development. However, it is important that single projects are pulled toward the global goals of the company. Moreover, with their development work in line with the specifications of the modular system, development projects have to contribute to the stability of the modular system. Therefore, the development of the modular system has to be constantly measured on different levels. Within this work, it is suggested to measure modularisation transition based on three levels: a) adherence of derivative products to the common reference architecture, b) direct results of the transition like higher commonality within development projects and c) effect of modularisation on global business objectives of the company. Moreover, it is important that these measurements have a direct impact on motivation of employees by implementing them into the company's measurement system.

Requirements:

- Constant monitoring of architecture-related metrics in order to measure the stability of the product architecture
- Constant monitoring of direct results of modularisation with metrics because the product architecture is not an end in itself

- Constant monitoring of impacted business goals by modularisation and analysis of the cause-effect relationship between modularisation activities and the development of business goals
- Measuring plans and specifications of the modular system against actual developments within derivative projects
- Modularisation metrics have to directly impact the motivation of involved employees (e.g. through payment-relevant target agreements)

Supporting comment:

Chapter 7 will give a detailed description how the requirements mentioned above can be fulfilled by taking use of modularisation metrics.

Cause-effect relationship:

The issues (see Section 5.3 or Figure 55) that are sought to be removed and the related aspect of support (see Section 5.5 or Figure 57) which are addressed by the factor in this question are as follows:

Addressed issues:	Related support aspects:
Lack of internal and external motivation	Modularisation evaluation
Lack of information and transparency	

Evaluating different aspects of modular system development helps to gain transparency and information about transitioning. Moreover, it considerably motivates involved roles to contribute to modularisation transition.

How is modularisation information represented within the company and how is modularisation transition supported by standard IT-systems?

Description:

The qualitative study from Chapter 5 revealed that information about the modular system (e.g. architecture specification) is usually stored within distributed folders and files like spreadsheets, graphs and textual descriptions. This kind of information is neither easily retrievable by consulting standard information systems (e.g. PLM, ERP) of the company, nor it is stored where this kind of information has actually to be used (i.e. in partially parallel derivative development projects and in the central modular system organisation). Moreover, single product development projects usually make their information accessible within the context of their project. For this reason, information about the modular system has not been stored centrally within companies so far. In the end, it is suggested that modularisation transition can be supported by making information about the modular system explicit within the company.

Requirements:

- Information about the modular system (e.g. roadmaps, module & interface specifications) is stored and maintained centrally and made available to derivative projects
- The structure of derivative products has to be comparable to the reference structure of the modular system
- The items of the modular system are linked (e.g. for impact analyses or derivation of metrics)
- Classification of product-architecture related items like modules for prompt identification
- Standardised naming of items according to modular system specification
- Neutral configuration of items so that they can be reused across projects
- Engineering change process for the modular system is supported by the standard IT-system

Supporting comment:

Chapter 8 will deal in detail how modular system information can be represented within standard IT-systems of companies in order to fulfil the requirements mentioned above.

Cause-effect relationship:

The issues (see Section 5.3 or Figure 55) that are sought to be removed and the related aspect of support (see Section 5.5 or Figure 57) which are addressed by the factor in this question are as follows:

Addressed issues:	Related support aspects:
Lack of information and transparency	Modularisation information

Providing information about the modular system centrally in standard IT-systems helps to remedy the lack of information and transparency during modular system development.

6.5.6 Evolution and change phase – product and module development projects

Even though, product development projects, module development projects and mixed development projects (i.e. a module for the modular system is developed during a product development project, see Figure 61) are own processes for themselves, they are subordinate processes of the modular system development life cycle. For the purpose of this section, it is suggested that each development project developing under the roof of the modular system has to adhere to overall architecture specifications, no matter if it is a product, a module variant or a mixed system. This is similar to the V-Model of Systems engineering where systems, products or modules are simply classified as upper and lower level systems with a similar but adaptable life cycle (Haskins et al., 2011, p. 27). Therefore, the milestones to be passed after each development phase have been suggested to be the same for product and module development. For instance, before proceeding to more detailed development works, requirements have to be defined, no matter if they are passed over from a higher level system or if they have to be directly derived from the market.

Are derivative development projects continually assessed on their adherence to product architecture specifications?

Description:

Derivative projects might have a number of reasons to create their local architecture instead of sticking to the specifications of the modular system. Even though, a part of these reasons might be well justified from a local perspective, it has to be ensured that all projects belonging to the modular system life cycle adhere to superordinate architecture specifications. Therefore, derivative development projects have to be frequently assessed if their design is in line with the overall modular system.

Requirements:

- Assessment of each development life cycle phase of the derivative project on its adherence to modular system specifications (see Figure 61): requirements phase, concept phase, technical design and testing phase.
- In order to synchronise different projects, it is important that the projects sticks to the overall module and product roadmaps or to other plans like the module-product matrix during each phase.
- During the requirements phase, it is important that the derivative project sticks to its assigned architecture relevant-requirements from the modular system requirements specification.
- During concept phase, it is important that the derivative project meets conceptual architectural specifications like module and interface descriptions or the module variety matrix, both from a functional and technical perspective.
- During technical design phase, it is important that a) module variants and interfaces stick to detailed drawings, interface descriptions and module envelopes and b) that the bill of material (either of the product or the module) is in line with the given structure of the modular system.
- During testing phase, it is important that technical parameters of the product or of module variants are indeed of physical coherence with the given test specifications. For example, during this phase it is important that interfaces of the module variant are proven to be suitable for application in other products to be derived of the modular system.

Supporting comment:

Chapter 8 shows how above mentioned assessments can be supported by representing relevant modular system information in standard IT-systems.

Cause-effect relationship:

The issues (see Section 5.3 or Figure 55) that are sought to be removed and the related aspect of support (see Section 5.5 or Figure 57) which are addressed by the factor in this question are as follows:

Addressed issues	Related support aspects:
Lack of extrinsic motivation	Process: post-architecting phase Modularisation information Modularisation evaluation

It is suggested that if products are regularly assessed on their compliance with the modular system during post-architecting phases, the extrinsic motivation of employees to pursue modularisation will be improved substantially. This evaluation is substantially supported by providing information about the modular system architecture in the core IT-system of the company.

On what level are cost reduction programmes established?

Description:

Cost reduction programmes, product benchmarks or DFA activities usually favour improvement of single products instead of a wide range of products because (oversized) modular products are compared to optimised one-of-a-kind-solutions. This means, that it is likely that single integral architectures are selected after some time instead of sticking to the common modular reference architecture. Therefore, it is important that cost reduction programmes do not favour single products but keep the modular reference architecture stable.

Requirement:

- Cost reduction programmes are conducted on module level instead of on single product level.

Cause-effect relationship:

The issues (see Section 5.3 or Figure 55) that are sought to be removed and the related aspect of support (see Section 5.5 or Figure 57) which are addressed by the factor in this question are as follows:

Addressed issues	Related support aspects:
Lack of intrinsic and extrinsic motivation	Modularisation evaluation

It is claimed that intrinsic and extrinsic motivation to pursue the goals of the overall modular system will be substantially increased if programmes such as cost reduction programmes will be conducted on module level instead of on single product level.

How are direct and indirect effects of complexity within derivative projects measured?

Description:

It might be beneficial for derivative development projects to violate against modular system specifications in order to achieve their local goals. However, any additional and unnecessary complexity that is generated by such violations has to be absorbed by other areas of the company and not by the single development project. Thus, complexity might increase steadily without penalising the root cause of it. Therefore, the effects of complexity have to be assessed directly at the point of their creation.

Requirements:

- Analysing or at least estimating the effects of complexity within derivative development projects
- Including complexity costs into project calculation
- Establishing costing systems that are capable to measure effects of complexity

Supporting comment:

Applying a classical activity-based costing approach might be helpful. A disguised example of some main complexity cost drivers is given in Figure 63. The underlying concept of the complexity cost approach is described in Section 7.2.4 “Assessment of monetary and performance implications”. Moreover, Sections 7.2.4 and 1.2 reference detailed complexity cost studies.

Cause-effect relationship:

The issues (see Section 5.3 or Figure 55) that are sought to be removed and the related aspect of support (see Section 5.5 or Figure 57) which are addressed by the factor in this question are as follows:

Addressed issues	Related support aspects:
Lack of intrinsic and extrinsic motivation	Modularisation evaluation

Intrinsic and extrinsic motivation of involved employees to contribute to the overall modular system goals can be improved if the effect of complexity is penalised directly where it is created.

6.6 Application guide for the assessment framework

The assessment framework can be used by industry to assess its capability to transition toward modular system development with stable architectures. In order to derive full support of the assessment framework, some kind of capability audit can be created by assembling questions and requirements into a detailed audit catalogue.

Such an audit catalogue has been used in the next section to validate the assessment framework. Following rating scheme and components have been used to identify gaps and potential for improvement:

- Rating scheme: According to the fulfilment of requirements, most questions of the audit can be rated with following rating scheme:

Score	Assessment Level	Range
4	Success criteria are met fully proven. There is no or only very little potential for improvement.	75% ≤ x ≤ 100%
3	Success criteria are met predominantly. Minor corrections are suggested for this question.	50% ≤ x < 75%
2	Success criteria are met partially. Major improvements are suggested for this question	25% ≤ x < 50%
1	Success criteria are met occasionally. An action plan and major improvments are suggested for this question.	0% < x < 25%
0	Success criteria are not met. A dedicated project or an action plan how a higher score can be achieved has to be implemented.	0%
n.a.	Not applicable	n.a.

Figure 65: Rating scheme for modularisation capability audit according to Heilemann and Culley (2015, p. 401)

- Further components of the audit catalogue: The questionnaire that was used in practice contains further data sets like consolidation of ratings, evidence & shown documents, reasoning for rating & comments, areas for improvement and measures to be worked on (Heilemann and Culley, 2015, p. 401).

The adjusted capability audit can either be used as an isolated stand-alone approach, it can be integrated into internal development-specific audit questionnaires of companies, or it can be integrated into quality gate questions of a specific stage-gate process.

6.7 Validation of the modularisation assessment framework

The modularisation assessment framework from Section 6.5 evaluates key factors which have been derived from the qualitative study of this work (see Section 5.2). The factors have been discussed and implemented either at the primary case company or at another secondary case company which was consulted as benchmark. Thus, each factor can be seen as appropriate for its use in industry with regards to content.

Hence, validation of the assessment framework targeted its usability and usefulness to identify potential for improvements in industry. In detail, the framework has been validated in different ways. Firstly, the framework has been validated by reasoning and refined after expert interviews. Secondly, the framework has been applied in two different industrial case companies.

6.7.1 Validation by reasoning and expert opinion

The assessment framework evolved during the qualitative study, described in Chapter 5. It can be seen as a collection of applied best practices that aim at removing the presented issues from Section 5.3 and Appendix D. The cause-effect relationship between issues and suggested means of support is listed below each question of the assessment framework (see Section 6.5) Each suggested factor that is listed in the assessment framework can at least be backed with a successfully tested example either in a primary case and/or in a secondary case.

After almost three years of the study, first concepts of the assessment framework have been validated with experts from industry. Therefore, the framework has been sent out to the experts and an interview was conducted afterwards. The first expert was an engineering manager in charge of a central engineering department that dealt with modularisation transition of a major global manufacturer. The assessment framework has been considered as applicable and useful while suggestions for improvement have been worked in for a new release of the framework.

Afterwards, a second expert interview was conducted with a senior engineering manager in charge of a business unit-wide modular system development project within the same manufacturer. The expert also had a strong background as head of development. The assessment framework was seen as applicable and highly relevant from a content point of view. The results of the discussion were used to further improve the assessment framework.

The results of validation with the two engineering managers have been directly recorded in a spreadsheet version of the assessment framework. In addition, handwritten notes were made in order to capture reasons behind the comments for further improvement of the framework. The spreadsheet versions have not been published because they contain some company-specific details.

The next step was to validate the revised assessment framework on a real project in industry.

6.7.2 Validation in Company A

The application of the assessment framework was done by an engineering project manager who was in charge of a project that transitioned from single product development toward modular system development within existing products in the primary case company.

Due to its earlier application, the disguised example was in its structure slightly different than the framework presented within this work (see Figure 66). The framework version has not been published because it contains confidential company-specific information.

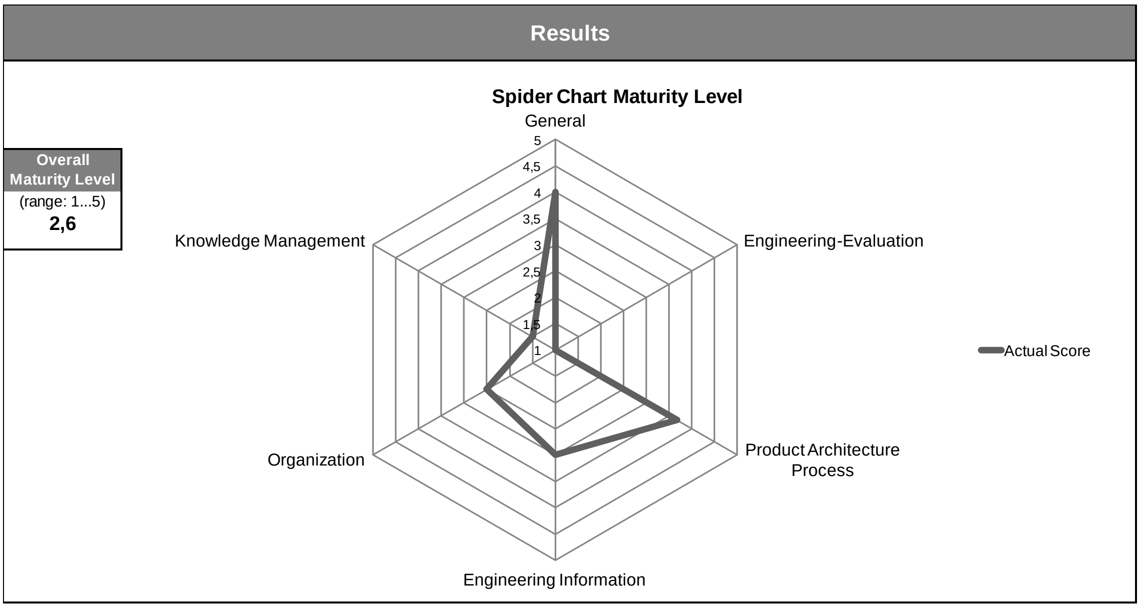


Figure 66: Example for the results (disguised) of applying the assessment framework in company A

Without going into detail, Figure 66 shows that the project performed well in general aspects of modularisation and in performing product architecture relevant processes (i.e. enablers for modularisation). There was considerable necessity to improve organisational and information/IT aspects of modular system development. Moreover, the assessment framework revealed that knowledge management and the evaluation of the modular system had to be set up from scratch. Concrete actions for improvement could be derived from the corresponding release of the assessment framework.

To sum up, the assessment framework was seen as applicable and relevant for industry. The concern was raised that the project will never achieve full score on the assessment scale. However, this is what the framework is built for. Wherever a company encounters issues and seeks out improvement, the assessment framework serves as flexible guide for further action that can be adopted to company needs. In order to being able to integrate the assessment framework into quality gates of the development process, the demand arose to add different life cycle phases as an additional dimension.

This amendment was handled with a different release of the framework and validated in a different company, described below.

6.7.3 Validation in Company B

The validation project of this section was designed in order to evaluate the modularisation assessment framework additionally from a life cycle perspective. The project took place in a secondary industrial case company where the overall organisation and a “flagship” project could be assessed on its ability for stable modular system development during all development life cycle phases. The main validator was a process engineer working on the development life cycle who used a mature version of the assessment framework to identify potential for improvement. The results of validation have been disguised in the course of this section.

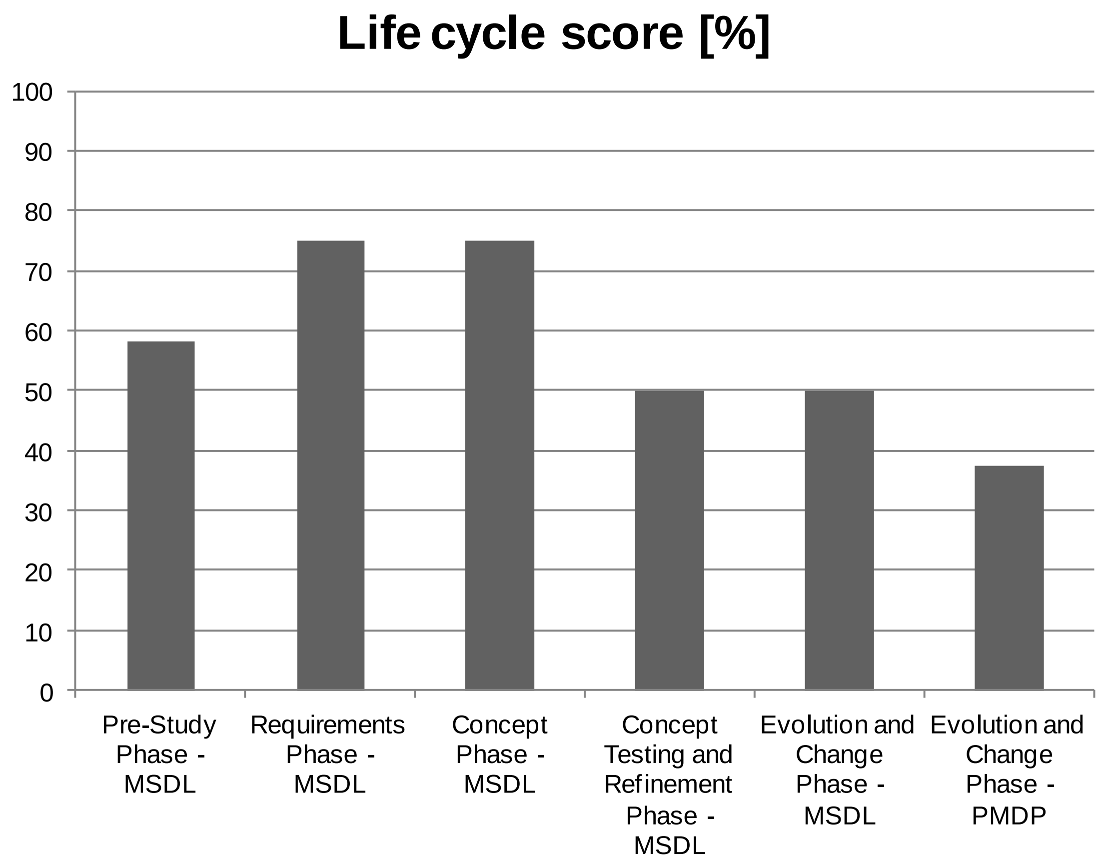


Figure 67: Results of modular system development life cycle assessment (disguised) in validation project B

The results of the assessment are summarised by Figure 67. The assessment and improvement potential for each phase of the figure is explained in the following paragraphs. It has to be noted that the real value behind the assessment is not its quantification but the qualitative findings for each phase.

- Pre-Study Phase – MSDL (Modular System Development Life Cycle):

Assessment summary:

It has been shown that the requirements of this process phase are mainly met. However, improvement actions for this process phase have been identified in almost all areas.

Assessment result:

There is a main initiative and a "flagship" project where a clear vision and vocabulary can be found in presentations. Moreover, there is a glossary defined which is mainly derived from norms and standards. A clear vision is also established on the programme (i.e. product line approach) together within communities and conventions. Guidelines on technical product plans and reuse strategy are available. Expectations about investments to be made and benefits to be expected have only been analysed in detail within the flagship project, but not within other projects. The budget has originally been provided for the concrete flagship project and the project has been set up.

However, the project was postponed for a year due to delivery and capacity problems. If the project will commence with some delay is still not confirmed yet. Moreover, there is no guarantee if such delays will be present more often.

Nevertheless, the overall programme on product line engineering is still ongoing. There has been a separate project established for the overall project and there was an organisational shift not only toward the flagship project, but also toward an organisation with common overarching roles which will guide the modular system. The transition did not affect IT and evaluation. The main focus of evaluation stays on quick delivery and immediate project cost. In the main phase of the flagship project there was constant monitoring and reporting to top management. There is no reporting about the degree of implementation and fulfilment of the overall product line engineering programme to top management. A distinction between a central modular system development process and derivative product development processes is not apparent from the process landscape of the company. However, the distinction works quite well if it is separately considered for the flagship project.

Potential for improvement:

The suggested improvement actions for this process phase comprise that the concept of the overall project should be better communicated and implemented on lower levels and in all diverse business units. The gap between documentation and practice should be removed. A concrete potential analysis considering complexity cost should be conducted. No delays in such projects should be allowed in future, thus, long-term goals should be more prioritised and not only short-term goals. The real implementation of transition should be monitored company-wide and the programme should cover IT and evaluation activities. In order to improve processes, the process landscape should reflect that there is a central and continuous modular system development process in which derivative development projects are embedded.

- Requirements Phase – MSDL:

Assessment summary:

Important factors of this phase are predominantly fulfilled. Only minor suggestions for improvements have been made.

Assessment result:

It has been shown that the systems business with one-of-a-kind solutions for totally different markets is very hard to predict. It is not predictable which bids can be won and which cannot be closed with a successful order. However, core functionalities of the solution have been derived and are applicable to different market segments and countries. Still, more effort needs to be put behind understanding different, yet unknown, markets. Nevertheless, knowledge, methods and tools are available how this understanding can be derived. It is done for projects, and for the core of the flagship programme, but not in general development projects. It is assumed that after each bid, the company implements those requirements documented in the customer specification. In any case, the customer is willing to pay for the specified requirements. Thus, there is not the same need to restrict or negotiate variety like it is within other companies producing for a market which demands high variety. For the purpose of single projects, traceability information is well-established between requirements and other artefacts. Thus, project-overarching artefacts could be better linked.

Potential for Improvement:

In order to further improve this process phase, the requirements procedure should be integrated into an overarching continuous modular system development process with dedicated resources for sustained market analyses. Moreover, traceability information could be more granular, resulting that more items and information such as success factors could be linked and represented. Another important point is that traceability information should be provided for artefacts across projects.

- Concept Phase – MSDL:

Assessment summary:

It has been shown that the requirements of this process phase are predominantly met. However, improvement actions for this process phase have been identified in almost all areas.

Assessment result:

The logical and physical architecture is well established, reviewed and documented. The architecture follows some guiding principles like safety-principles. Senior experts are involved in architecting. The architecture is set up for several projects, but not explicitly documented for several projects, except in the areas where model-based system engineering has been established. Architecting was well done in flagship project and appropriate documentation to establish architectures have been provided. Plans for artefacts of the modular system comprise modules, module variants and interfaces which are established within the project. No explicit plans for project-overarching artefacts are available. However, planning and synchronisation of artefacts is appropriate in the flagship project.

Potential for improvement:

In order to get improvements for this process phase, the architecture should be made explicit and binding for various projects under scope (not valid for flagship project). Moreover, planning should be made explicitly across projects, yet it is only done for single solutions (not valid for flagship project).

- Concept Testing and Refinement Phase - MSDL:

Assessment summary:

This process phase has been met. However, there are considerable improvements suggested for this process phase.

Assessment result:

Artefacts which are seen as common are tested and refined in the project where they are developed. No testing for other purposes and no dedicated process phase is provided. The flagship project is an exception to this. Generally, there are no central common architecture specifications passed on to derivative development projects. Documentation is only made project-wise.

Potential for improvement:

In order to improve projects, a dedicated development process phase for testing project-overarching artefacts should be implemented. Moreover, a dedicated development process phase for documentation of project-overarching artefacts should be implemented. This documentation should be systematically passed on to derivative development projects.

- Evolution and Change Phase - MSDL:

Assessment summary:

The important factors for this process phase have been met. Nevertheless, there should be major improvements implemented during this phase.

Assessment result:

External changes like market characteristics, technology trends and requirements are only for products with actively managed product policy. This is not valid for products which evolve from bidding processes. However, in any case there are various role descriptions which cover these aspects. There is a dedicated and well-defined change management process for all artefacts of the modular system. However, there is no special treatment of generic project-overarching artefacts.

On the one hand, there are roles dedicated to the product line from market and technical perspective. On the other hand there are organisational departments or projects for generic product development. Concerning implementation of roles, it depends on the specific project to be analysed. If all projects use the roles as described by the product line programme (i.e. role for overarching product line management, technical role for artefacts of overarching modular system) the roles are implemented appropriately. However, there are dozens of projects not having these roles or where these roles are doing other tasks than focusing on project-overarching activities and goals. Especially in smaller projects focus, also with roles, is on fast delivery of the single derivative product.

There is no measurement of modularisation, standardisation or architecture goals and metrics. Project metrics like the delivery time which could be effected by the architecture are measured. However, there is no linkage between the architecture and such common project metrics. There are several metrics like the metric Design Complexity (for single projects), the metric for Code Reuse and a checklist for reusability of software. There is an extensive metric catalogue, but it is not compulsory and only weakly applied.

Concerning IT-integration, the documentation of product line information is quite well described: generic configuration items, product line structure in the engineering IT-system. However, reality could not be shown to be that neat. There are different IT-systems for different departments and no neutral configuration, consequently, there is not one single source of truth. Configuration is done project-wise. No generic specifications are set up and the structure of derivative products is not comparable to the reference architecture. There are no links between items of the modular system. There are no unified naming and attributes for items in IT-systems which makes them always getting mixed up.

Potential for improvement:

Several potential improvements have been identified for this phase. Resources to work on continuous analysis of market trends and their impact on the modular system could be strengthened. Moreover, the focus could be more on overarching generic products and systems than on a narrow scope. For the engineering change process, there should be a differentiation of the change process into project/solution changes and generic project-overarching changes. Moreover, changes to generic modular system specifications should be constantly managed. Protection of the modular system architecture should be in the focus of the dedicated change process. Roles could be established for particular tasks of the modular system development life cycle and modularisation activities of these roles in daily practice could be strengthened. The mentioned overarching roles should focus more on project-overarching activities than on technical project management lead in the concrete project on hand. The metric system should be extended from metrics for reuse to architectural compliance. Moreover, there should be an explicit focus on project-overarching/multi-project metrics. In the case, where such metrics are applied, there should be quantified metric calculation instead of qualitative estimations. Modularisation metrics should be better implementation into real practice and lifted up to management reporting level. From a perspective of time, it should be measured during milestones if architecture metrics and modular system specifications are met.

- Evolution and Change Phase - Product and Module Development Projects (PMDP):

Assessment summary:

The requirements for this process phase have only partially met. Thus, major improvements have to be planned for this phase.

Assessment result:

Overarching product architecture issues are not covered by milestone reviews. No metrics are applied during derivative development projects to measure architectural adherence. Quality gates do not cover whether the architecture of the derivative product is in line with the overarching modular system architecture. There are no consequences if common specifications are not met. No mutual change process between overarching and derivate specifications has been established. Ratio projects are not conducted on module level, but on a unit level that is handed over to the customer. Quantified complexity calculation does not build a part of the project reviews, reporting or analysis. Reporting on project status is rather on main events, milestones, finance, risks or quality assurance than on complexity matters. An installed complexity tool measures the complexity of the derivative project in order to being able to derive measures for systems engineering or risk mitigation. However, this is not modularisation-specific or for an project-overarching perspective. There is no consideration of effects of complexity for other functions (e.g. service, manufacturing) or on other development projects which cannot reuse a certain module.

Potential for improvement:

Suggested significant improvements are manifold. Sticking to generic architectural rules and specifications should be regularly measured during derivative projects. Not meeting the goals of common specifications should have negative consequences on derivative projects, outbalancing other goals of the project. The project-overarching aspect of modular system development should be integrated for instance into templates of engineering work products, milestones, quality gates and the company development process description. Ratio projects should be broken down onto module level instead on the solution that is delivered to the customer. It must be negotiated that the customer cannot have everything that he wants to have. The designated project complexity tool should be extended to the dimension of effects on other areas (i.e. consequences of neglected reuse). Adding calculation of complexity cost to project calculation is seen as potential means for improvement. Awareness about effects of increased complexity and root cause of increasing complexity have to be significantly increased.

- Scores for modularisation support aspects:

In order to identify improvement potential from another perspective, the results of the assessment framework have also been arranged around modularisation support aspects. Figure 68 shows the scores for each modularisation support aspect. It shows that the company performs comparatively well in implementing modularisation roles/organisation and in performing processes which are relevant for modularisation. However, these aspects should be guided toward a more stringent focus on project-overarching modular system activities. Modularisation evaluation should be significantly improved toward the measurement of cross-project artefacts in order to reach global company goals. The aspect with the greatest potential for improvement is the handling of modularisation information. In this area it is suggested that the company should establish a unified configuration management for central storage of all modularisation items. This would support design engineers in obtaining knowledge about modules and other modularisation items to be reused.

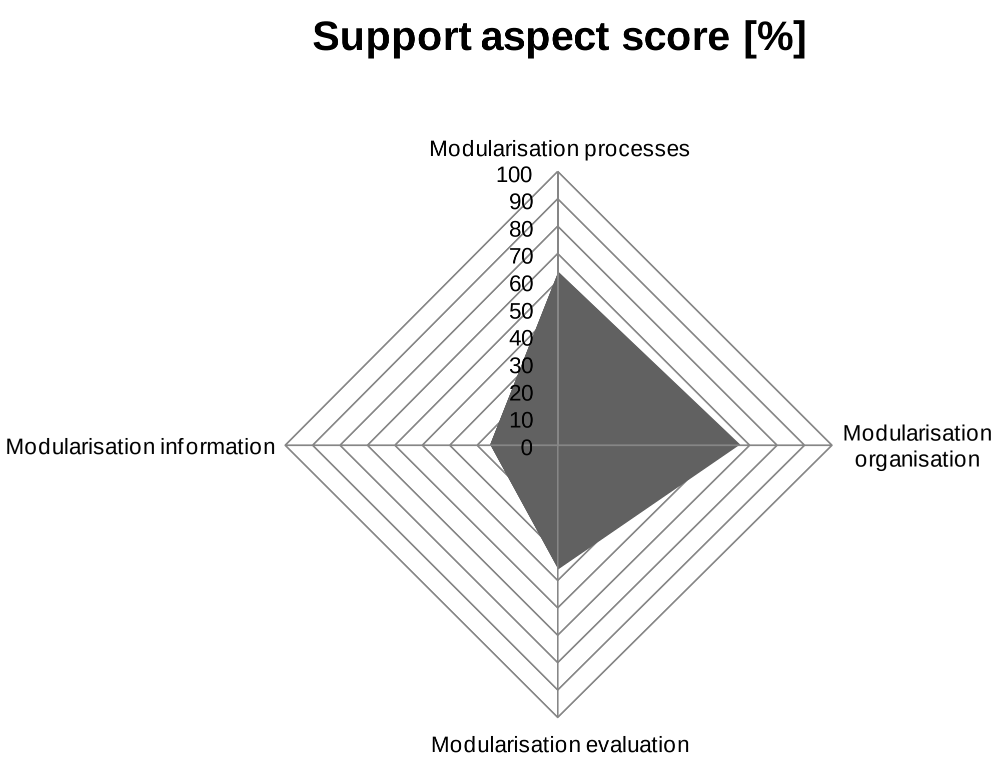


Figure 68: Results of modular system development life cycle assessment in validation project B – broken up into modularisation aspects

- Summary of validation activities:

The assessment framework has been validated from two different perspectives.

Firstly, it is important to know whether the framework is relevant for the case company and whether it indeed leads to new insights for the company. It has been shown that the assessment framework leads indeed to significant findings. On the one hand, it might be common sense that there could always be more done in every modular system life cycle phase and for each support aspect while the pros and cons of each action have to be weighted out carefully. It has also been shown that there is a huge gap between theoretical working groups that are experts to "methods" and the daily project life of engineers. The application of the framework has also risen awareness that the overall modular system goals should be more important than short-term goals of single derivative projects in order to fight complexity sustainably. On the other hand, there have also been significant and surprising new findings. It has been shown that there are some areas which are only weakly capable to tackle the previously strongly “advertised” push toward reuse, modularisation and complexity reduction. The development life cycle should be redefined in order to clearly show how the company wants its modular system to be continuously evolved over time. This also requires to introduce a modular system engineering change process which is relevant for the whole company. Moreover, the system how development projects are measured should be extensively rearranged in order to penalise creation of complexity and to support reuse and compliance with the common modular reference architecture. Another striking finding was that the way product information is managed is totally incapable to support engineers in designing module variants which are in line with the common reference architecture. Thus, the level of reuse of such variants could be significantly increased. Furthermore, the stability of the modular system during post-architecting phases could be improved. This has been an area that has so far been neglected by the company.

Secondly, it can be stated that the assessment framework is indeed usable by another industry and another kind of business. During application of the framework, no major defect within its applicability or its basic rationale could be identified. Nevertheless, it was found that each time the framework is applied within another setting, it is helpful if it is adapted to the needs and processes of the specific object of application beforehand. This will maximise the value that the company can derive from the assessment. It has been shown that the assessment framework is still generic enough, but that it can be adopted flexibly to the detailed settings where it is applied.

6.8 Discussion

This has been a very detailed and intensive part of the research and the results have a number of implications for academics and practitioners. They are detailed below.

6.8.1 Implications for researchers

Previous studies have been frequently built around different aspects and dimensions. Assessment frameworks which are built around certain dimensions, like the one presented by Van den Linden (2005), have the advantage that they can be applied independently from a concrete modular system project. Similarly, the preliminary assessment framework that evolved in the course of this research thesis (Heilemann and Culley, 2015) and that has been validated in Section 6.7.2, also showed that this type of assessment framework can be applied conveniently as standalone approach. However, if a concrete modular system life cycle and its corresponding derivative product development projects shall be guided toward the overall goals of the company, it is necessary to make those assessments within concrete projects. Moreover, process and time dependencies of all phases of such a project can only be covered if the assessment framework is applied sequentially in a concrete project. For instance, it does only make sense to assess roles and responsibilities of a modular system after the product architecture is established and planned how it is going to be implemented. This is one reason amongst others why this framework has been built around different development life cycle phases.

The modular system development life cycle which has been presented in Section 6.4 is in itself a significant research finding. This development life cycle was not only used to the assessment framework of this chapter but also to foster understanding about evolution and change of the modular system during later phases of the life cycle. While benchmark partners in industry mainly applied single product development models and platform development models, the proposed life cycle can be seen as innovative and supportive for transitioning toward modular system development. Even though such models might possibly exist in industry (e.g. in automotive companies that heavily depend on modular systems), their transfer and the adaption of research literature to contemporary streams in industry remains a weak spot that has been remedied with the help of this research work.

Another reason why the assessment framework was organised along the life cycle (see Table 18) was the weakness of other research, discussed in Sections 3.2, 5.4 and 5.5, to provide support for the stability of the modular system which is vulnerable during all life cycle phases. However, the most problematic phases have been identified as those phases which are passed after establishing the product architecture.

Firstly, the assessment framework handles the implementation of the modular system in the pre-study phase. By emphasising this phase, this research supports the studies of Jasmine & Vasantha (2010) and Karandikar & Nidamarthi (2007) who stress the importance of properly implementing reuse strategies and standardisation.

Secondly, the assessment framework deals with the requirements and concept phase of the modular system life cycle. These are the phases where most of recent research has been focused on (see Section 3.2 and Chapter 5). Therefore, this research supports the importance of these phases while it not fully agrees with the way how these phases have been processed by other researchers (see Section 5.4). A good example for handling the requirements phase and the concept phase is given by Kreimeyer (2014) who reports about modularisation transition from the view of an internal product architecture department at a global truck manufacturer. However, his task description of the department mainly focuses on the market phase, the concept phase and on providing transparency in early phases, but not in later phases which are important for stability.

While there has been no similar research for testing the concept of the modular system phase, very few other researchers have conducted research in the evolution and change phase of the modular system. During this phase, other researchers focus on modularisation processes (Alblas and Wortmann, 2008; Arnoscht, 2011; Bahns, Gregor Beckmann, et al., 2015; Schuh, Aleksic and Rudolf, 2015; Schuh, Aleksic and Arnoscht, 2013) and modularisation organisation (Arnoscht, 2011; Bahns, Gregor Beckmann, et al., 2015; Kreimeyer, 2014), but not on modularisation assessment. While this phase is also emphasised by this work, but remedied with different means of support, shows that the findings of this chapter are novel and highly relevant.

6.8.2 Implications for industry

The modularisation assessment framework supports organisations to maintain stable modular product architectures throughout the whole modular system life cycle. This is done by offering advice on important aspects to consider during pre-architecting, architecting and post-architecting phases and by removing issues that other companies encountered during modularisation transition.

Companies can integrate the assessment framework into their stage-gate processes, quality gate reviews, milestone reviews or they can use the assessment framework as stand-alone assessment in a CMMI-like manner. Section 6.1 has shown that such integrations into assessment “tools” have already been done in other areas like risk management or component-base software engineering. However, the field of modular system transition had so far not been integrated into such an assessment “tool”. This shortcoming is remedied with the presented framework in this chapter. Given other areas where these integrations have been served as helpful support, it is suggested that this will be the case for this modularisation assessment framework as well.

Besides the benefits of the assessment framework discussed above, there are two other aspects which could be beneficial for industry:

- It makes the overall concept of modularisation transition more transparent to people that are not deep into that topic. This is done by braking down a massive overall thing into several small steps.
- It is a good tool to manage continuous process improvement for modularisation and prevents the modularisation initiative “falling asleep” after some time, like it is with other improvement initiatives after generating a lot of “noise” at the beginning.

In order to make the assessment framework applicable to different companies and industries it has been kept on a level that does not go into detailed tools, methods or templates. As the whole assessment framework has its origin in real industrial application, it would be possible to present a detailed method, template or tool for each answer of a question of the framework. However, the qualitative study has shown that on a very detailed level, different companies have different conditions and needs. Therefore, it was decided to design the framework in following way: “The framework tells you what is important to consider and gives hints for implementation. However, you have to adjust a detailed solution to your specific needs.” It is suggested that this approach provides more value to industry than prescribing which field to fill in a specific template or matrix.

6.8.3 Limitation of the assessment framework

Although the assessment framework has been validated in industry, there has not yet been a complete validation of this support. This kind of support has been validated mainly based on the aspects "logic", "relevancy" and "applicability" so far. While these aspects have been validated successfully, the proof whether the assessment framework leads indeed to better results of modularisation transition has not been validated. However, this was not possible within the nature of such a study. Moreover, the size of the assessment framework has been found as quite huge and time-consuming to be applied in industry. Therefore, industry may wish to downsize the framework during its adaption in order to obtain a lean assessment.

6.9 Summary

It was the aim of this chapter to answer RQ 2. The chapter presents an assessment framework which can be used to assess the transition toward modular system development. By applying the framework, industrial practitioners can identify weak spots and areas for improvement. The framework can be applied flexibly, either in an isolated assessment or it can be used by industry to integrate it into milestone reviews of their stage-gate process. The framework has been iteratively validated and improved in industry with the result that it is relevant from a content point of view and applicable by engineers and engineering mangers.

The assessment framework is set out to tackle the issues presented in Chapter 5.3 and Appendix D. The link to each issue is established for each question of the assessment framework. This is done by drawing upon the support aspects modularisation processes (pre-architecting, architecting and post-architecting phases), modularisation organisation, modularisation evaluation and modularisation information. It gets obvious that there is a complex link between issues, support aspects and different modularisation life cycle phases. The following tables show the frequency of how issues (see Table 16) and support aspects (see Table 17) are covered by the assessment framework.

Table 16: Coverage of identified issues by the assessment framework

Lack of time and resources	**Lack of information and transparency**	**Lack of intrinsic and extrinsic motivation**
11 Questions	11 Questions	12 Questions

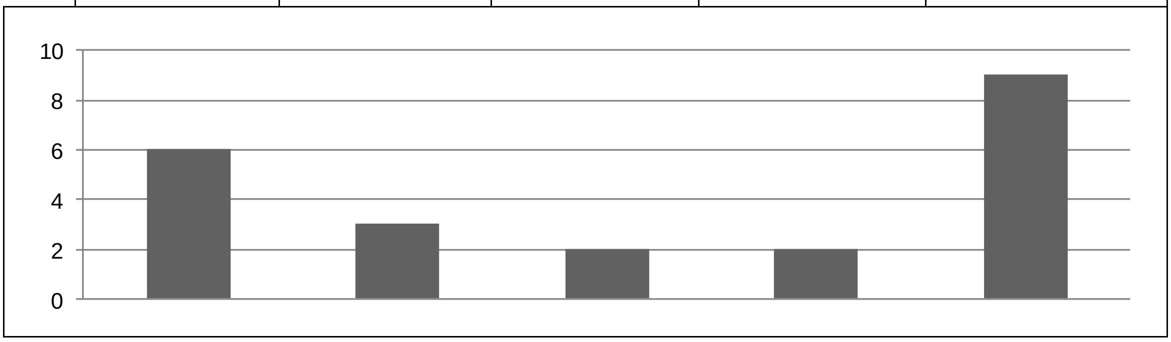
Table 17: Coverage of support aspects by the assessment framework

Modularisation evaluation	**Modularisation process**	**Modularisation information**	**Modularisation organisation**
7 Questions	15 Questions	3 Questions	3 Questions

The qualitative study in Chapter 5 revealed that later phases of the modular system life cycle, i.e. after concept phase, are most vulnerable to failure with respect to stable modular system development. Therefore, the questions of the assessment framework are assigned to different life cycle phases of the modular system life cycle. Thus, it is possible to attack issues at the point of their root cause. Nevertheless, this means that earlier phases have to be considered as well if a flaw occurring there endangers the stability of the modular system later on. Table 18 shows how the 22 questions of the assessment framework are organised. It gets evident that the emphasis for support is on the evolution and change phase which is the phase where most of the issues of Section 5.3 have been shown to occur. Moreover, this is the phase which has attracted only little attention by contemporary researchers so far (see Section 3.2).

Table 18: Coverage of different modular system development life cycle phases by the modularisation assessment framework

Pre-study phase	Requirements phase	Concept phase	Concept testing phase	Evolution and change phase
6 Questions	3 Questions	2 Questions	2 Questions	9 Questions



The suggested support for each question in the assessment framework is of medium detail level and does not prescribe any concrete methods or tools. Few researchers have started to work on the evaluation and change phase by researching the support aspects modularisation processes (Alblas and Wortmann, 2008; Arnoscht, 2011; Bahns, Gregor Beckmann, et al., 2015; Schuh, Aleksic and Rudolf, 2015; Schuh, Aleksic and Arnoscht, 2013) and modularisation organisation (Arnoscht, 2011; Bahns, Gregor Beckmann, et al., 2015; Kreimeyer, 2014) quite recently. In order to close the remaining gaps, the next two chapters of this research thesis will focus on detailed support for modularisation evaluation and modularisation information during the evaluation and change phase of modular system development.

7 Modularisation metrics

The previous chapter has introduced an assessment framework that assesses enablers for modularisation transition (see Figure 59 of Chapter 6) with a specific focus on the modular system development life cycle. This chapter will now move on to examine metrics which assesses the results of modularisation transition (see Figure 69). In this context, the term "results" refer to rather technical outcomes of modularisation transition and its effect on the organisation (see Figure 59 of Chapter 6).

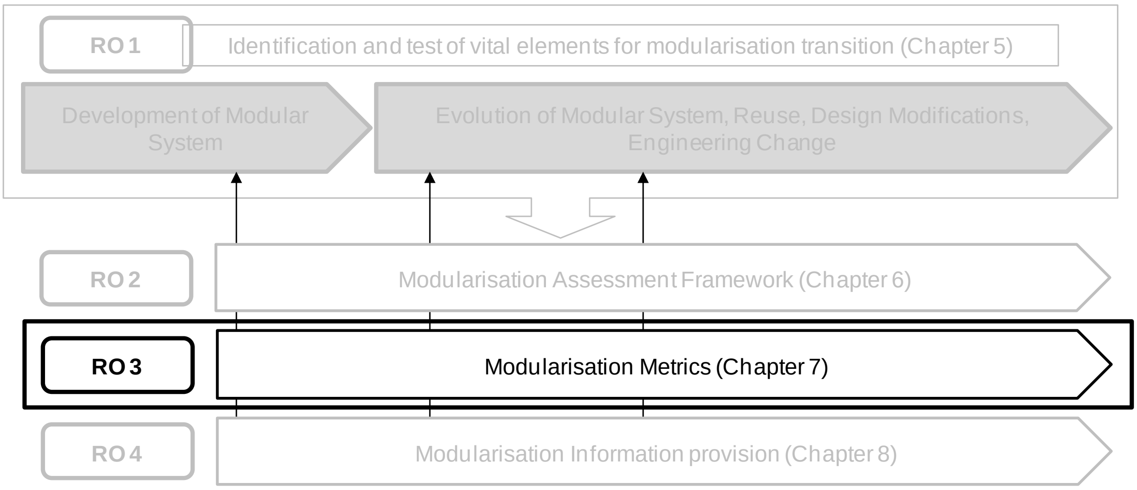


Figure 69: Modularisation metrics in the context of this research thesis

7.1 Chapter overview

It is the aim of this chapter to present a coherent set of modularisation metrics which can be applied to support companies in transitioning toward modular system development. The metrics are applied along the development life cycle of the modular system. They constantly pull the focus of affected roles towards specifications of the modular reference architecture instead of thinking in terms of isolated single development projects.

7.1.1 Research setting and methodology

Development of the presented modularisation metrics constitutes the second contribution of the Prescriptive Study of this research work. This is done by applying following steps:

1. Analysing the literature about state of the art of modularisation metrics
2. Collecting and analysing requirements for measuring modularisation transition in the primary case company
3. Iteratively developing metrics by moving from a concept and detailed design towards realisation, application and intervention of the metrics in industry. This was done by

following a goal-driven measurement methodology that breaks down business goals and other requirements into measurable data elements and implementation plans (Park, Goethert and Florac, 1996).

4. Validating the metrics based on tests, application and expert interviews

All these steps were conducted in projects of the primary case company.

7.1.2 Chapter elements

This chapter overview Section 7.1 is followed by a state of the art review in Section 7.2. Then, a survey about requirements collection for modularisation metrics in industry is briefly presented (see Section 7.3). Afterwards, a case study in industry with test of existing metrics is shortly described (see Section 7.3). Section 7.4 is the core of this chapter and presents the modularisation metrics developed within the course of this research work. This is followed by a section that reports about validation of the metrics in Section 7.5. Finally, the chapter concludes with a discussion (see Section 7.6) and a summary (see Section 7.7). An overview of the elements of Chapter 7 is given in Figure 70.

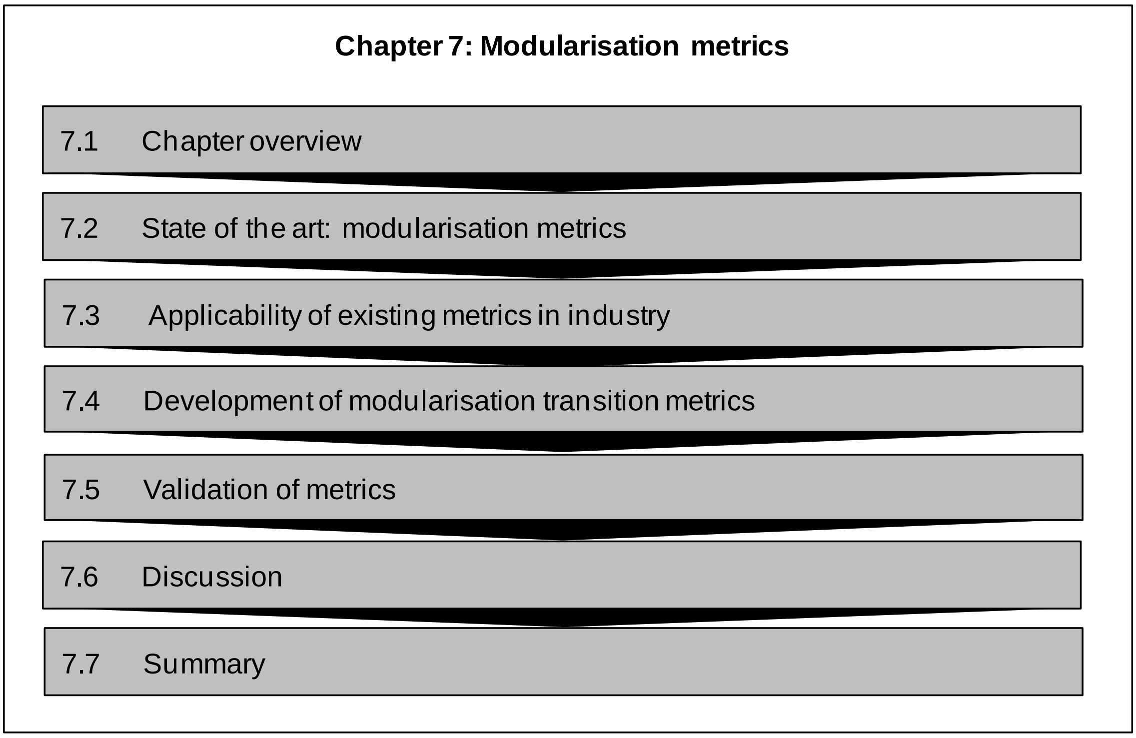


Figure 70: Elements of Chapter 7

7.2 State of the art: modularisation metrics

Metrics for evaluation of modularisation can really be classified according to abstract modularisation principles or the objective that shall be achieved with modularisation. Such objectives could be higher commonality, reuse, variety or cost savings. The review also covers the evaluation of strategic and financial effects of modular systems. Finally, several holistic evaluation frameworks which cover a set of different modularisation-related metrics are presented. An overview of the formulas of selected metrics is given in Appendix F.

7.2.1 Assessment of modularity principles

On an abstract level there are modularisation metrics which capture the type of product architecture of a product or product range.

- Functional structures

Holtta and Otto (2003, 2005) use functional structures to derive a design complexity metric with the purpose to evaluate different product architecture alternatives. The underlying principle behind this metric is to structure product functions so that complex interfaces are located within modules and less complex interfaces are located between modules. Stone et al. (1999) apply a customer-weighted sub-function similarity metric to identify architectural similarity across a set of products and an aggregate customer need rating for modules in order to justify modular product architectures.

- Functional-physical relations

Relations between functional and physical elements are a major factor in engineering design (Suh 2001). For instance, Steva et al. (2006) present a function-component frequency value and Mattson and Magleby (2001) apply a modularity metric which relates the number of modules to the number of functions.

- Physical interactions between elements

Gershenson et al. (1999) and Guo and Gershenson (2004) present modularity metrics which captures the degree of modularity based on the occurrence of connections inside and outside of modules. In a similar manner, Holtta-Otto and De Weck (2007) use component-component DSMs to derive two modularity metrics. The first metric, the non-zero-fraction, generally describes the sparsity of connections in a DSM. The second "singular value modularity metric" measures the location of connectivity between components. It reflects if connectivity is concentrated between a few components which may form a module, in a connecting modular bus or if connectivity is distributed across the product.

Mikkola & Gassmann (2003) and Mikkola (2006) develop a metric which measures the degree of modularity embedded in product architectures of product families. The suggested modularisation function considers components, interfaces between components, coupling between components and substitutability of new-to-the-firm components.

In order to evaluate product architectures based on the strength of coupling, Martin and Ishii (2002) introduce a Coupling Index (CI) which measures the strength of coupling between components or modules of a product architecture.

Sosa et al. (2005) define three metrics to measure the interactions between components. The three modularity metrics are based on how components share interfaces with adjacent components, how changes are propagated to all other components in the product and how components lie in the "dependency path" of two interacting components.

Further coupling metrics that also belong to this metric category are reviewed by Holtta-Otto et al. (2012).

7.2.2 Assessment of internal complexity and external variety

- Assessment of commonality

The Degree of Commonality Index presented by Collier (1981, 1982) reflects the degree of component commonality in a considered product spectrum. This metric is extended to a Total Constant Commonality Index, TCCI (Wacker and Treleven, 1986) with fixed boundaries for better comparability. Martin and Ishii (1996) propose a commonality index that measures the proportion of unique parts to the proportion of total parts in the product family. Beisheim and Stotz (2013) introduce three standardisation metrics (standardisation degree regarding part usage, standardisation degree regarding part consumption, standardisation degree overall) in order to divide parts into preferred parts, service parts and run-out parts. Sinigalias and Dentsoras (2015) calculate a composite standardisation index by taking the percentage of common parts and the compliance of parts with designated standards into account.

The Product Communality Index (PCI) of Kota et al. (2000) is an indicator for the share of common components and the overall number of common components which do not add specific features to the product within a product family. Thus, distinct components that should ideally be common are penalised.

The viewpoint on commonality is extended by Siddique et al. (1998). The question of how good a product platform is answered by addressing component, interface and assembly commonality. The presented commonality indices by Siddique et al. (1998) can be applied to assess the commonality of various dimensions which can be weighted individually. However, the metrics do not show how beneficial the commonality of a certain domain item really is as all domain items are considered equally. For instance, it is more beneficial to standardise stamped parts with expensive workstations and tools than standardising the length of a screw. To overcome some related shortcomings, Jiao and Tseng (2000) consider price and cost for a component, the quantity of a component for each operation, and the end product volume and integrate it into the metric "Component Part Commonality Index" ($CI^{(C)}$).

Johnson and Kirchain (2010) present six different commonality metrics which are equipped with different weighting-factors. According to the weighting-factors, the metrics can be classified as follows: piece-based, mass-weighted, cost-weighted, investment-weighted, production volume-weighted and production volume/investment weighted.

The "Total Commonality Index" aims at the redesign of existing product families taking their generic bill-of-materials (GBOM) into account. It is the aim of the metric to evaluate the overall commonality of a product family by evaluating common components, must-generic items and options with the GBOM (Blecker and Abdelkafi, 2007). Romanowski and Nagi (2005) take use of classical BOMs in order to assess the symmetric difference between unordered BOMs. It is the purpose of their metric to analyse commonality in BOM-trees.

With the purpose to redesign existing products, the "Comprehensive Metric for Commonality" (CMC) evaluates if components deviate from ideal commonality or variety. Therefore, the factors components size, geometry, material, manufacturing, assembly & fastening, cost and allowed variety within the product family are taken into consideration (Thevenot and Simpson, 2007).

The "Commonality versus Diversity Index" (CDI) takes the component categories "common", "variant" and "unique" into account. The index evaluates if commonality and diversity are in line with given specifications (Alizon, Shooter and Timothy W. Simpson, 2009).

- Assessment of variety and flexibility

Rapp (1999) suggests simple key performance indicators which can be used to measure the variance generation performance of a product architecture alternative (e.g. number of products which are or can be sold to the customer). The metric of Kohlhase (1996) relates the number of product derivates from the modular system which are actually sold or prognosticated to be sold to the number of product derivates which could ideally be sold in theory (Kohlhase, 1996, p. 55).

- Generational variety and flexibility

Martin and Ishii (2002) develop a generational variety index (GVI) which measures the flexibility of a product architecture for future design changes. The GVI indicates the redesign effort which is necessary to accommodate future design changes. A similar evaluation method is the "Change Mode and Effect Analysis" (CMEA) which is different from the Generational Variety Index in its ability to handle added functionality. Keese et al. (2006) develop an enhanced CMEA to assess the flexibility of products to planned and unknown design changes. The work draws upon the research of Rajan et al. (2003, 2004, 2005). The CMEA methodology shows the effect of changes to the product and calculates a flexibility measure that shows how flexible the product is for future changes.

7.2.3 Assessment of strategic and PLC reasons for modularity

Blees et al. (2009) evaluate different product architecture alternatives in the concept phase prior to selecting the best one for further detailed design. The architecture alternatives are qualitatively and subjectively assessed against their estimated fulfilment of different module drivers.

Compatibility of components in the modules of a product architecture alternative concerning the post life intend is measured by Newcomb et al. (1996). The post life intent comprises the viewpoint of recycling, reuse, incineration and land filling. Gershenson et al. (1999) evaluate product architectures by measuring similarities and dependencies between components and all relevant life cycle processes. Their modularity measure expresses how well similar and dependent components are grouped into modules and to which extend they use the same PLC processes. The presented measure also evaluates component independence, PLC process independence, component similarity and PLC process similarity.

Fixson (2005) develops a framework which can be used to support product architecture design. The framework aims at assessing different product architecture dimensions (function-component allocation style, interface intensity, interface reversibility, interface standardisation) in conjunction with decisions from the organisational product, process and supply chain domain (e.g. size/location of production capacity).

Marti (2007) measures functionality and physical complexity of products and derives a complexity matrix from the calculation of those metrics. Depending on the position of the product in the complexity matrix, guidelines from different fields (basic strategy, overall strategy, product life cycle considerations) can be used to improve the position of the products in the complexity matrix.

7.2.4 Assessment of monetary and performance implications

- Evaluation of cost

To measure the cost effectiveness of product architecture design activities, Martin and Ishii (1996, 1997) develop a methodology to estimate costs that are caused by variety. The researchers state that the effect of variety on indirect cost is difficult to consider and generally not well understood. Therefore, they use three metrics to indirectly measure the effect of product variety on indirect cost of a whole product line: the Communality-Index (CI), the Differentiation-Point-Index (DI) and the Setup-Cost-Index (SI).

To overcome the shortcomings of estimating the effects on cost with the help of indices (Martin and Ishii, 1996, 1997), (Fixson and Blanchard, 2001; Fixson and Clark, 2002; Fixson, 2002, 2004) set out to directly compare the cost effects of different architecture alternatives.

Siddique and Repphun (2001) use another approach by linking product architecture decisions to associated activities. Each activity is related to certain cost distributions which helps to identify financial gains or drawbacks of a platform approach. Park and Simpson (2005, 2008) apply an activity based costing system to evaluate product platforms. Their analysis considers direct material, direct labour and activity costs. For this approach, it is necessary to identify those design decisions which have an influence on cost drivers which again have an influence on related activity cost. Thyssen et al. (2006) use activity based costing as a tool to assess the effects of modularisation within organisations. To resolve the trade-off between the cost associated with implementing a platform strategy, manufacturing cost and product performance, Farrell and Simpson (2010) also apply an activity based costing method.

In recent approaches, it is suggested to assess different product architecture concepts based on *complexity cost*. Therefore, the financial effects of different levels of complexity on the value chain are estimated (Hansen et al., 2012; Myrodia and Hvam, 2015; Ripperda and Krause, 2015, 2014).

- Evaluation of value

Kohlhase's (1996, p. 119) value of a modular system (W_{BKS}) evaluates the technical and economical value of a modular system. Meyer and Lehnerd (1997) suggest to use three metrics on a quite high level to compare the performance of product platform development to the development of derivative products. The researchers observed that most development products are monitored based on the slip rate. The slip rate determines the deviation of the project from time and budget plans. However, this monitoring leads to a favouring of derivative product development which is easier to handle and control concerning time and budget. For this reason, the researchers introduce the three comparative efficiency metrics. The platform efficiency metric asks whether a platform has enough potential to create volume effects when developing derivative products. The cycle time efficiency metric helps to understand the ratio between the development time of derivative products and the development time of the platform. Platform effectiveness answers the question about commercial effectiveness of a platform by relating revenue that is earned by derivates compared to the development cost of the platform and the derivates (Meyer & Lehnerd 1997).

Gonzalez-Zugasti et al. (2001) develop a quantitative measure to assess the value in terms of benefits and investments of different product architecture alternatives. Thus, companies are supported to choose between multi and individual product development during early design stages. Zacharias and Yassine (2008) propose a model for the optimised value of a modular product family based on initial platform investment, commonality level and the number of variants to be produced. The model helps design engineers in the conceptual design of the product family.

7.2.5 Integrated approaches to assess product architectures

Several researchers have developed integrated approaches by applying various metrics. These approaches can be divided into two categories with different purposes. Firstly, such an approach can be applied to choose between different architecture concepts and, secondly, such an approach can be used to control and monitor performance of product architectures. Examples from both categories are given below.

- Multi-criteria frameworks for screening different architecture concepts

Based on a framework for the effects of complexity (Orfi, Terpenny and Sahin-Sariisik, 2011), Orfi et al. (2012) develop a performance measurement system which comprises eight different metrics: a product variety index, component variety ratio, process variety ratio, part-level index, interconnectivity level, customer sensitivity, specificity level and a coupling level. The metrics can be used to assist engineering managers in product family design.

Otto and Holtta (2004) identify 19 platform assessment metrics based on expert interviews, personal experience and literature search. Based on the 19 metrics, a multi-criteria framework is developed for screening preliminary platform concepts. The framework is applied in the early platform architecture phase. Prior to the application of the evaluation methodology, a functional flow model of the platform needs to be established and platform modules on functional level need to be identified. The metrics reflect several viewpoints on platform performance and are grouped in the categories complexity, customer, flexibility, organisation, variety and after sales. It is beyond the scope of this work to go deep into the formulas of the 19 metrics.

In a similar approach, Ericsson and Erixon (1999) develop ten metrics which evaluate product characteristics of modular products and relate them to possibly resulting effects. The ten characteristic-effect relationships are as follows (Ericsson and Erixon, 1999), see Table 19.

Table 19: Modularisation characteristics and effects defined by Ericsson and Erixon (1999)

Characteristic	Effect
Interface complexity	Lead time in development
Share of carryover	Development costs
Share of purchased modules	Development capacity
Assortment complexity	Product costs
Share of purchased modules	System costs
Number of modules in product	Lead time
Share of separately tested modules	Quality
Multiple use	Variant flexibility
Functional purity in modules	Service/upgrading
Material purity in modules	Recyclability

- Multi-criteria frameworks to control and monitor architectures over time

Döpke et al. (2009) develop a performance measurement system to assess the effects of variant management. Möller et al. (2011) integrate various metrics from different fields to control and monitor standardisation initiatives Both research streams integrate various simple metrics into assessment dashboards. Schuh et al. (2014) establish a performance measurement system that is integrated into a balanced scorecard model that reflects company-specific objectives. The balanced scorecard contains product program, product architecture, supply chain, production and finance related metrics. A similar but more extensive balanced scorecard approach is given below by Junge (2005).

Rennekamp (2013) applies nine metrics within three dimensions to determine the level of complexity of a company. Therefore, he derives possible cause-effect relationships between the outcome of a metric and its effect on a certain dimension. The first three metrics are used to identify potential of improvement within the product programme. The second three metrics are used to "optimise" the product architecture. Finally, the last three metrics are used to find actions for improvement in the value stream.

Junge (2005) develops a holistic evaluation methodology to control modular product families during the development phase. Therefore, the researcher introduces an integrated performance-measurement-approach which closely follows the principles of a balanced scorecard (BSC) for modular product families. The BSC includes following relevant company perspectives: development, production, marketing, sales, and finance.

Firstly, the overall target of the modular product architecture is determined along with the targets of the perspectives finance, development, production, and marketing and sales. Secondly, cause-effect relationships between the overall goal and the goals of the different perspectives are established (see Figure 71). Then, 22 KPIs are collected from literature and introduced to measure the characteristics of the product architecture. The KPIs are also related to the different company perspectives. Finally, the measures are related to the goals of each perspective.

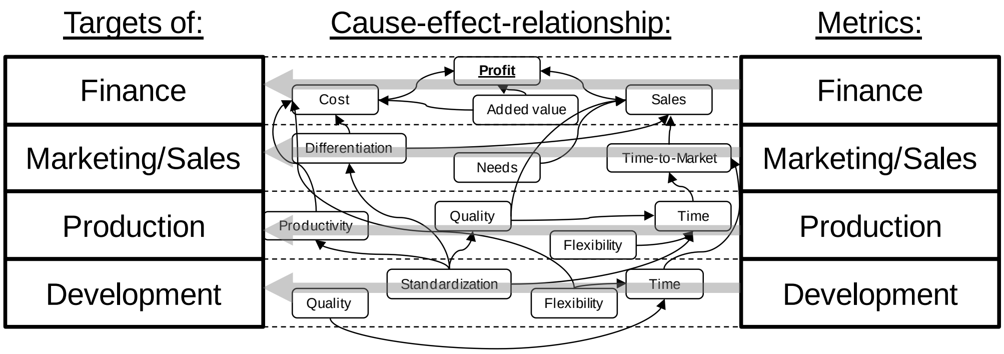


Figure 71: Principle of Junge's (2005) holistic evaluation methodology

7.2.6 Summary: existing metrics

The metric approaches presented are an excerpt of available literature on modularisation metrics. Appendix F gives a detailed overview about the formulas of selected existing metrics available in literature This literature review was used to characterise how and why fellow researchers apply modularisation metrics instead of presenting the metrics in detail. To complete the picture, many more similar metrics on modularisation can be found in different reviews with specific focus like on coupling modularity or on component commonality (Gershenson, Prasad and Zhang, 2004; Holtta-Otto et al., 2012; Jiao, Simpson and Siddique, 2007; Simpson et al., 2014; Van Eikema Hommes, 2008).

Metrics from literature that focus on product architecture principles are quite abstract. Such metrics have been applied quite early in the development life cycle.

Less abstract evaluation of product architectures can be found in the field of internal complexity and external variety. Evaluation of commonality is well developed in literature and metrics for various variety applications can also be found.

The evaluation of cost for complexity and cost for providing variety is of high interest in practice. However, complex cause-effect relationships between characteristics of the product architecture or internal and external complexity with actual cost often result in metrics that have to be carefully applied. Data that is used to calculate cost metrics has either to be roughly estimated in an early phase or tediously derived in expensive analyses. Although there are possibilities to relate complexity to cost, they are also fraught with disadvantages practical limitations like too many influencing factors and effects lagging behind causes by several years.

The evaluation of value of certain product architecture alternatives is the most difficult and vague way of evaluation. The evaluation approaches discussed here are either on a high and managerial level or on an abstract and theoretical level.

Integrated methods for product architecture evaluation either combine various factors or relate certain architecture characteristics to overall company goals. This combination is backed by estimated linkage between causes and effects of product architecture characteristics. All integrated methods have in common that they are information intensive or vague in cause-effect relations.

The next section will move on to scrutinise applicability of the presented metrics for modularisation transition in industry.

7.3 Applicability of existing metrics

In order to determine the suitability of existing metrics as a basis for the overall research goal, Section 7.3.1 briefly presents a study which derives requirements for modularisation metrics from industry. It is then possible to evaluate in Section 7.3.2 exiting metrics for application in industry.

7.3.1 Survey - Requirements from industry for modularisation metrics

It is the purpose of this section to define requirements of industrial practitioners for the application of metrics assessing the transition toward modular system development. A detailed description of the survey is given by Heilemann et al. (2013).

The study took place in the primary case company (see Chapter 1.8, Chapter 4 and Appendix B) of this research work. Therefore, a semi-structured questionnaire approach (Blessing and Chakrabarti, 2009) was adapted. After studying literature and conducting some preliminary interviews, a questionnaire with pre-selected requirements was generated. The participants were asked to rate the provided requirements and to give and rate further requirements they have. In addition, participants were encouraged to give further comments or to contact the researcher directly to discuss their requirements. Prioritisation was done based on a high-medium-low rating scale where nine points were given for a "high" rating, three points were given for a "medium" rating and zero points for a "low" rating. The unequal score distribution had the aim to avoid "middle" ratings and to go for extreme values in the rating scheme. The questionnaire (see Appendix E) was sent out to 49 top managers, engineering mangers, engineers and consultants who were selected based on their involvement and expertise in modularisation and engineering design. The reply rate was at about 82 %. In several cases, further interviews were conducted with participants with the aim to obtain mutual understanding.

In order to obtain unambiguous results, the collected requirements were clustered. Above mentioned rating scores were assigned to each requirement cluster. For further requirements prioritisation, those 50 % of requirements that achieved a score of 75 % have been selected for implementation – provided that contradictions between requirements have been resolved and that the final set of requirements is coherent. Figure 72 gives an overview of rated requirements to be implemented and requirements that have been excluded during development of modularisation metrics.

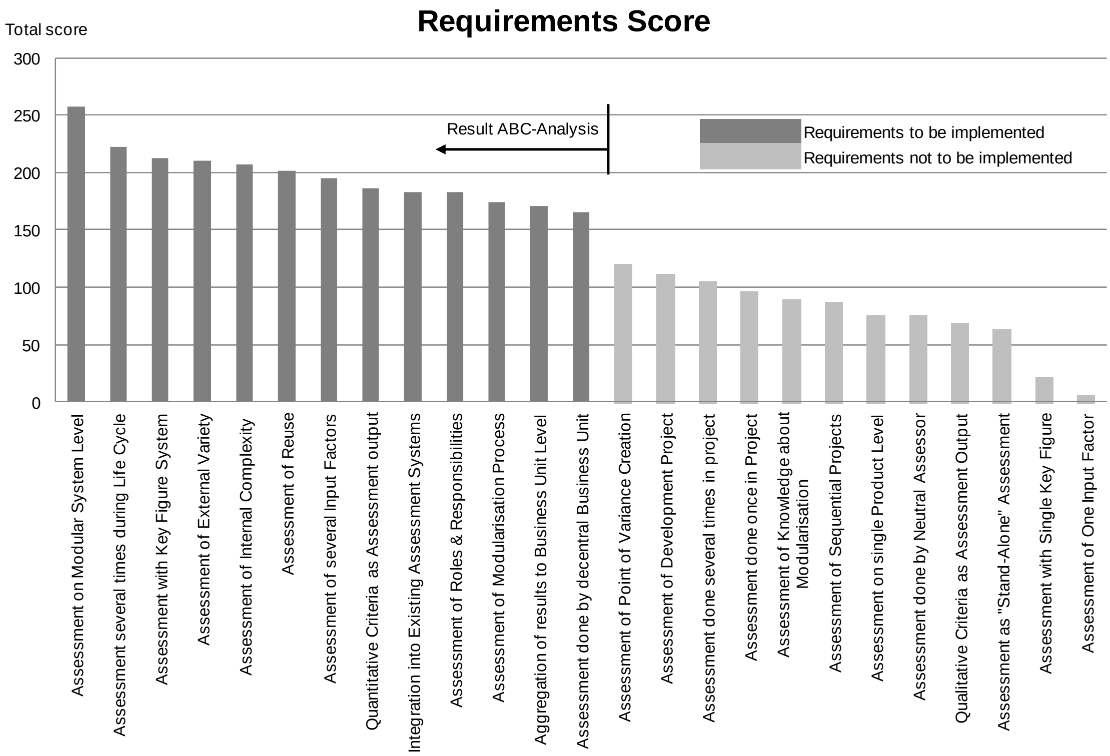


Figure 72: Results of modularisation metric requirements prioritisation

Table 20 shows the final set of requirements of industry for assessing modularisation transition with metrics that have been derived from Figure 72. The requirements have been divided into two parts: a) "what" to measure with the metrics and b) "how" to measure modularisation transition. Moreover, the table also shows a premise for modularisation metrics that have been mentioned by most of the participants. This premise was seen as mandatory to be implemented and concerns efficient calculation of the metrics.

Table 20: Requirements of "what" and "how" to measure with modularisation metrics

"What" to measure	**"How" to measure**
Premise: Automated calculation of metrics with data from standard IT-system	
Assessment on modular system level: External Variety, Internal Complexity, Reuse	Several times during life cycle
	With key figure system that considers several in- and output factors
Assessment of modularisation roles	Quantitative criteria instead of qualitative criteria and estimations
Assessment of modularisation process	Integration into existing assessment systems like milestone reviews
	Aggregation of results to business unit level
	Assessment done by de-central business unit itself

7.3.2 Evaluation of existing metrics in industry

Before proceeding to the implementation of the mentioned metrics, it is important to analyse existing metrics based on above mentioned premise (see Table 20) and their ability to be automated and efficiently calculated. A detailed report of the analysis is provided by Heilemann et al. (2013). The main points of the analysis are summarised below.

Most of the metrics and assessment systems presented in the literature review of this chapter (see Section 7.2) require large amounts of information input. The information required for the calculation of the metrics is either retrieved by analysing bills of material (BOMs), making use of Design Structure Matrices (DSMs), analysing functional graphs of products, disassembling products, estimations of impacts or expert opinion. However, such information retrieval is resource-intensive or dependent on personal judgement.

The study reported in this section took place in the primary case company of this research work where the information retrieval approaches mentioned above were not acceptable or practical, given that several thousands of products and hundred thousands of parts had to be analysed during a modularisation transition activity on a company-wide basis. Thus, it was considered necessary to calculate the metrics based on information which is readily available in standard IT-systems, i.e. ERP and PLM. Also as most of the metrics require information that is related to the product structure, it was required to retrieve BOM information that is available within the case company.

In addition to the systems of the case company, the International Demonstration and Education System (IDES), which represents the SAP ERP-system of a sample enterprise, was used to triangulate the findings of the case company.

In order to make the applicability analysis, following questions were asked:

- Can the metrics be calculated by using BOM information of the case company or of IDES?
- Can the metrics be calculated by drawing upon information that is available by using standard ERP or PLM (i.e. SAP R/3 and Teamcenter PLM) systems of the case company or of IDES?

In order to provide comparability, the answers have been divided into the same categories as the literature review (see Section 7.2) of this chapter. These categories are listed and evaluated below:

- Metrics assessing modularity principles

None of the presented metrics of this category could be automatically derived by drawing upon BOM, PLM or ERP information. While input information like the number of parts or the number of modules could be calculated, it was not possible to retrieve informal information like functions, similarity, dependencies, interactions, interaction strength and information about newly/uniquely purchased components in a structured manner for further analyses.

- Metrics assessing internal complexity and external variety

This category is the category where most of the metrics could be derived automatically compared to the other metric categories, though, the majority of the metrics were not easily calculable due to following reasons:

  - Need to do estimations
  - Need to distinguish between *ideally* common components, unique components and variety components
  - Need to determine interface and assembly characteristics that are related to a specific component
  - Classification data or information not available in IT-systems
  - Parts lists are not in a GBOM format

The following table shows a detailed overview of calculability of the metrics of this category:

Table 21: Overview of computability of commonality and variety metrics, updated version of Heilemann et al. (2013)

	Collier (1982)	Wacker and Treleven (1986)	Martin and Ishii (1996)	Beisheim and Stotz (2013)	Sinigalias and Dentsoras (2015)	Kota et al. (2000)	Siddique et al. (1998)	Jiao and Tseng (2000)	Johnson and Kirchain (2010)	Blecker and Abdelkafi (2007)	Romanowski and Nagi (2005)	Thevenot and Simpson (2007)	Rapp (1999)	Kohlhase (1996)	Martin and Ishii (2002)	Keese et al. (2006), Rajan et al. (2005, 2004, 2003)
	Commonality Metrics												Variety/Flexibility			
Computable with BOM information	✓	✓	✓	✓	X	X	X	✓	✓	only w/ GBOM	only w/ GBOM	X	✓	X	X	X
Computable with ERP/PDM information	✓	✓	✓	✓	X	X	X	✓	✓	X	X	X	✓	X	X	X

- Metrics assessing strategic and PLC reasons for modularity

This metric category requires input data like life cycle data, service data or supply data. None of the metrics of this category could be calculated by using structured or formal data from BOM, PLM or ERP-systems.

- Metrics assessing monetary and performance implications

None of the metrics of this category could be calculated by drawing upon data available in BOM, PLM or ERP information. Information required for this category includes activity based costing information or cause-effect relationships between different factors and cost. However, such kind of information has not been easily available within the case company or IDES.

- Integrated approaches for metrics

There is a differentiated view on simple integrated approaches because they integrate various different input factors for metrics from other categories. For instance, the metrics of Döpke (2009) and Möller (2011) could be calculated easily with available BOM, PDM or ERP information within the case company.

On the other hand, it was not possible to easily calculate, for instance, the metrics of Junge (2005) with reasonable resources. One limitation of the approach is the calculation of 22 metrics which could be cumbersome for any organisation without additional support. How difficult it is to gather information and calculate the metrics from his standalone approach is demonstrated by the author. The application in industry which was conducted as case study does not involve real data for each metric. Therefore, many assumptions were made with notional data to demonstrate workability of the BSC.

Due to the high number of metrics which have been collected and developed by researchers in this area, not all of them have been shown here. In sum, they either fall into the "easy" category of Döpke (2009) and Möller (2011), or into the "impossible" category of Junge (2005).

7.3.3 Overview: applicability of existing metrics

Section 7.3 provides an extended overview about the usefulness of existing metrics for the purpose of this work. Table 22 clusters different metrics and different requirements into overall metric- and evaluation-categories. Acknowledging that each distinct metric has its own detailed strengths and weaknesses, it is the aim of the table to give a sense about strength and weaknesses of each metric-category for the purpose of this work. This is done through judgement and not through detailed calculations. The evaluation-categories consider calculation, application, purpose and life cycle requirements (see below). They can be seen as some kind of combination between the requirements of industry and the requirements of this research work (i.e. establishing stable modular systems). A detailed description of each evaluation-category can be found in Table 22. Therefore, more detailed requirements have been assigned above each evaluation category. The evaluation-categories and the corresponding evaluation of metrics are as follows.

Calculation

Concerning calculation almost all metric categories contain weaknesses, except commonality metrics and some integrated approaches.

Application

The categories commonality, variety, financial, value and integrated approaches perform quite well in the way they can be applied along the development life cycle, mainly due to the fact that they are independent from module clustering and, thus, can handle any given modular system architecture from earlier concept phases.

Purpose

It is also important that the metric categories support the purpose of guiding derivative development projects toward central modular system development with stable architectures. While there is no metric category that fully supports this criterion, there are the commonality, variety, financial and integrated categories which at least pull projects toward a more global optimum without making sure that it is done systematically with a stable modular system.

Life Cycle

Finally, while most of the presented metrics aim at the concept phase in order to support engineers in selecting between different product architecture alternatives, there are some metric categories which can be used throughout various life cycle phases: commonality, variety, financial and integrated approaches with focus on continuously measuring the performance of modularity.

Table 22: Overview of reviewed metric categories and their usefulness for the purpose of this work

Legend

Requirements of evaluation-category judged as …

- ○ not met
- ◔ met occasionally
- ◑ met partially
- ◕ met predominantly
- ● met fully

	Calculation (Quantitative criteria: not based on estimations; Automated derivation from standard IT-Systems; Data availability in companies)	Application (Independent from reasons for module clustering (e.g. module drivers); Metrics can be applied throughout modular system lifecycle)	Purpose (Guiding projects toward modular systems; Pull different roles toward "global" modularisation goals instead of single project goals; Metrics foster stability of modular system over time)	Life cycele phase
Metrics based on functional structures: e.g. Holtta and Otto (2003, 2005), Stone et al. (1999)	○	○	◔	Concept phase - designing product architecture
Metrics based on functional-physical relations: e.g. Steva (2006), Mattson and Magleby (2001)	◔	○	◔	Concept phase - designing product architecture
Physical interactions between elements: e.g. Guo and Gershenson (2004), Holtta-Otto and De Weck (2007), Mikkola (2006), Martin and Ishii (2002), Sosa et al. (2005), Holtta-Otto et al. (2012)	◔	○	◔	Concept phase - designing product architecture
Assessment of commonality: e.g. Collier (1981, 1982), Wacker and Treleven (1986), Martin and Ishii (1996), Beisheim and Stotz (2013), Sinigalias and Dentsoras (2015), Kota et al. (2000), Siddique et al. (1998), Jiao and Tseng (2000), Johnson and Kirchain (2010), Blecker and Abdelfaki (2007), Romanowski and Nagi (2005), Thevenot and Simpson (2007), Alizon et al. (2009)	◕	●	◑	Various phases - For redesign of existing products (BOM has to be available) or for estimations in novel designs, to compare different manufacturing firms, to compare the effect of commonality on different factors (e.g. cost)
Assessment of variety & flexibility: e.g. Rapp (1999), Kohlhase (1996), Martin and Ishii (2002), Keese et al. (2006), Rajan et al. (2003, 2004, 2005)	◔	●	◑	Various phases - For planning and designing of product architectures in the early phase, For monitoring existing products/architectures
Assessment of strategic reasons: e.g. Blees et al. (2009), Newcomb et al. (1996), Gershenson et al. (1999), Fixson (2005), Marti (2007)	○	○	◔	Concept phase - To plan and compare product architectures
Assessment of financial aspects: e.g. Martin and Ishii (1996, 1997), Fixson (2002, 2004), Fixson and Blanchard (2001), Fixson and Clark (2002), Siddique and Repphun (2001), Park and Simpson (2005, 2008), Thysen et al. (2006), Farell and Simpson (2010), Hansen et al. (2012), Ripperda and Krause (2014, 2015)	○	◕	◑	Concept phase As long as companies do not have ABC cost information, such approaches are only applied during product concept phase. Mostly in order to select between different alternatives.
Assessment of value: e.g. Kohlhase (1996), Meyer and Lehnerd (1997), Gonzalez-Zugasti et al. (2001), Zacharias and Yassine (2008)	◔	◕	○	Concept phase - In early phases to select between different product architecture alternatives.
Integrated approaches - concept selection: e.g. Otto and Holtta (2004), Orfi et al. (2012), Ericsson and Erixon (1999)	◔	◑	◔	Concept phase - In early phases to select between different product architecture alternatives.
Integrated approaches - continuous measurement: e.g. Döpke et al. (2009), Möller et al. (2011), Schuh et al. (2014), Junge (2005)	◑	●	◑	Various phases - To continuosly monitor possible effects of product architectures

7.4 Development of modularisation metrics

This section presents the metrics which have been developed in the course of this work in much iteration over a prolonged period. Topics like how to make the metrics comparable across different organisation or how to normalise them and what to include into metric calculation filled lengthy and controversial discussions. The section was written by keeping in mind that it is more important for the reader to understand what the metrics shall achieve and how they can be applied. It is not the purpose of this section to go into the very detail of their formulas or factors as this is a problem of the context of their application. It is assumed that once the purpose and application of each metric is properly understood, they can be detailed and amended freely to the specific context of their application.

7.4.1 Application of the metrics based on three levels

The metrics developed for the purpose of this work have been divided to be applied on three levels, see left hand side of Figure 73:

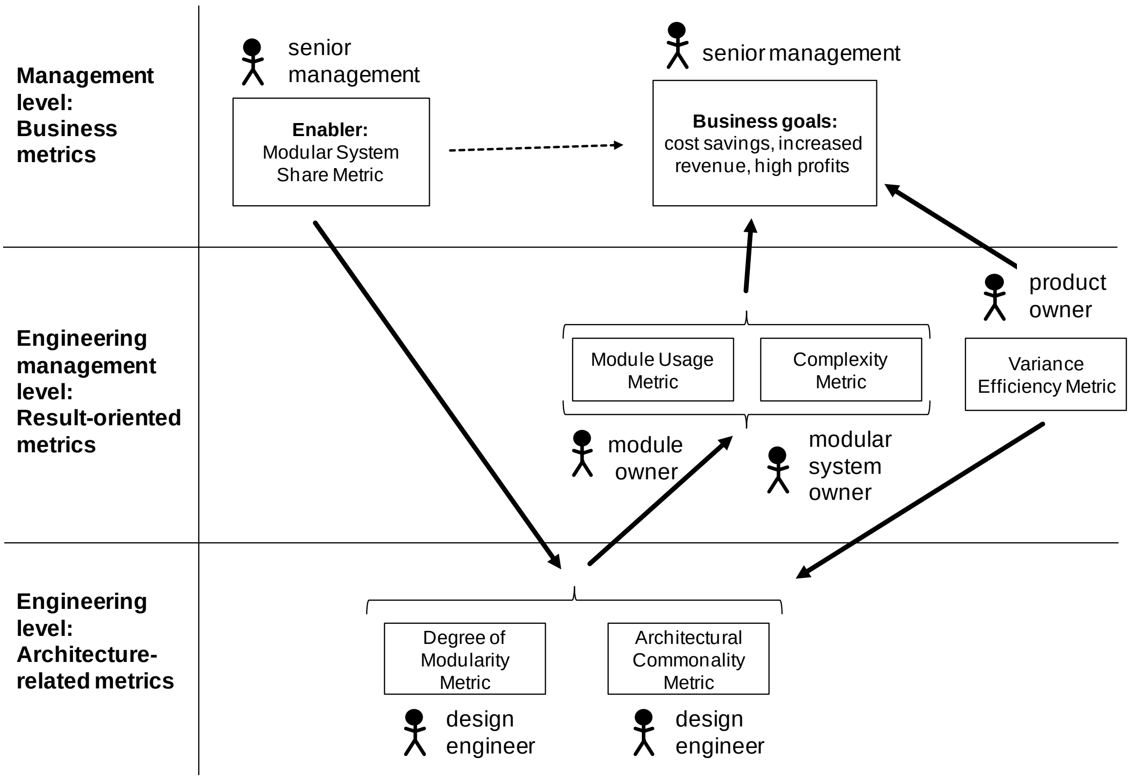


Figure 73: Application of metrics on different levels and interdependencies between metrics, based on Heilemann et al. (2015)

- The first level is the top management level which is measured by business metrics. It is assumed that top management wants to improve its business metrics (e.g. costs, revenue, profits, time to market) by transitioning toward modular system development (see upper right side of Figure 73). In order to achieve this, it is decided to increase the share of products to be derived from the common modular system. Therefore, the development of a modular system is triggered (see Figure 61 of Chapter 6). This development shall be reflected by the metric Modular System Share (see upper left side of Figure 73).

- The second level concerns engineering management. This is measured with result-oriented metrics. For the purpose of this work, result-oriented metrics measure whether the modular system contributes to higher commonality, reuse and variety which in turn positively influence business goals. This level is measured with the metrics Module Usage, Complexity Metric and Variance Efficiency which are assigned to the responsibility of different roles (see middle layer of Figure 73).
- The third level is the engineering level which is measured by product architecture metrics. After the decision of top management to increase the share to be derived from the central modular system, design engineers in derivative product development projects have to develop their local products in line with the common modular reference architecture. This adherence is measured locally with the metrics Degree of Modularity and Architectural Commonality (see lower part of Figure 73). It is suggested that an improvement of these metrics directly contributes to an improvement of the metrics on the other levels over a cause-effect chain.

The next section will give another perspective of the application of the metrics from a modular system development life cycle point of view.

7.4.2 Application along the modular system development life cycle

Figure 61 of Section 6.4 proposed a modular system development life cycle which focuses on all phases and particularly on the stability of the modular system. Figure 74 shows how the presented metrics can be related to the modular system development life cycle. At the beginning of the life cycle, initial goals for the Modular System Share are set. These goals form the basis for the concept and refined concept of the modular system. With more mature knowledge, after the concept of the modular system is established and modular system specifications are available, initial goals can be confirmed and broken down to lower level metrics.

From here on, the application of the metrics can be divided into two parts:

- The metrics Modular System Share and Variance Efficiency are measured and aggregated on a regular base. By comparing goals to actual values, actions for improvement can be derived.
- The Commonality Metric, Module Usage, Degree of Modularity and Architectural Commonality can already be directly measured in derivative development projects in order to influence stability of the modular system directly where it is designed. For instance, the metrics can be calculated after the technical design release (TD), after project end or additionally on a regular base. All metrics from local derivative development projects are analysed in order to influence the project on hand and they are also

aggregated to an overall modular system level in order to gain transparency from an overall perspective.

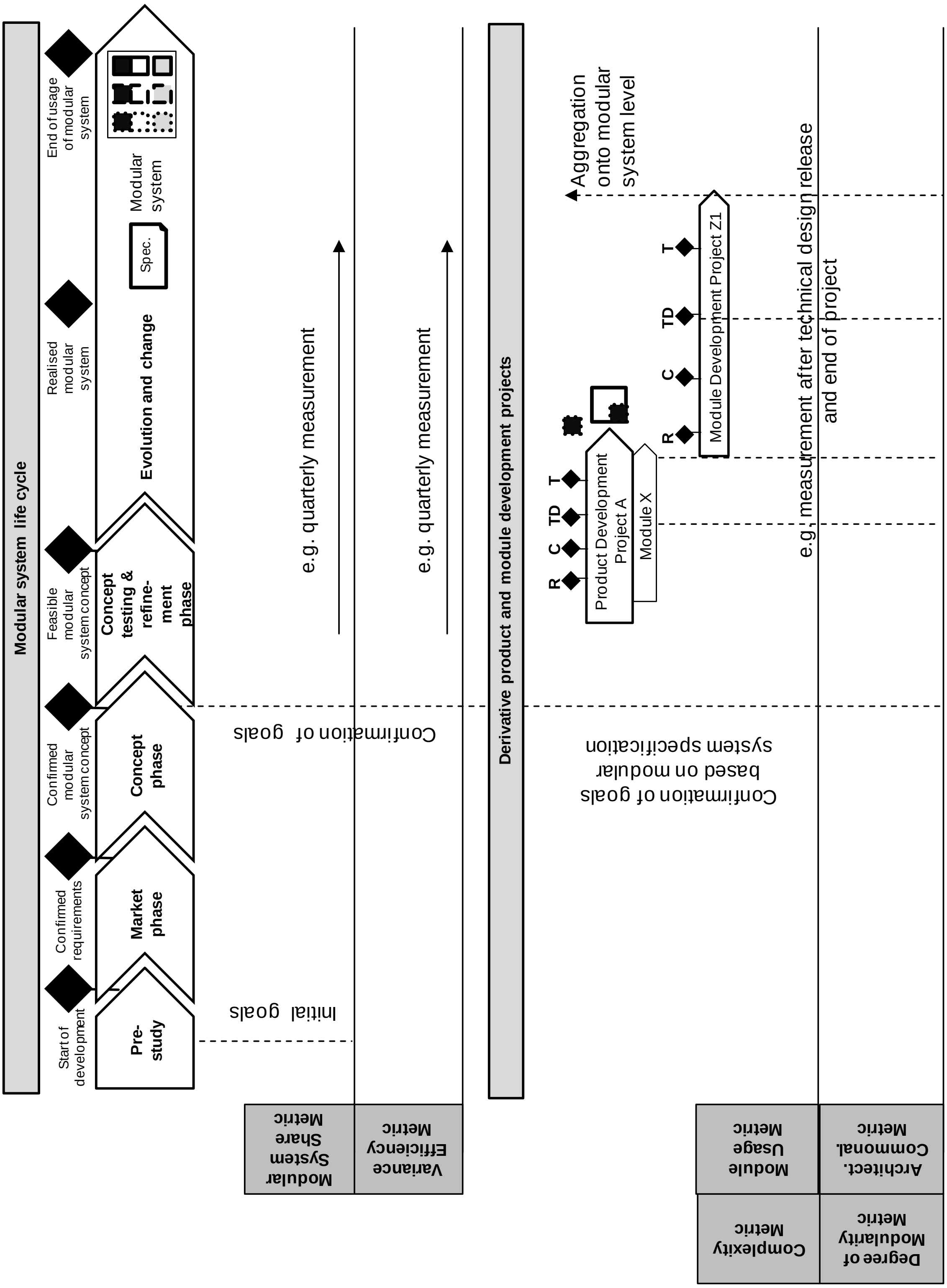


Figure 74: Integration of the modularisation metrics into the modular system development life cycle

- It has to be considered that not all metrics are applicable to all artefacts of a modular system like it is shown by Figure 74. For instance, the figure shows a module development project. During a modular development project it does not make sense to calculate the Degree of Modularity, unless the Degree of Modularity of the module shall be calculated. Moreover, as there are no products involved in the development of the module, it is not possible to calculate the Complexity Metric during such a project. Nevertheless, all metrics are applicable on an aggregated level.

The next three sections will now turn to present the purpose and calculation of each metric of the three levels in detail: business level metrics (see Section 7.4.3), result-oriented metrics (see Section 7.4.4) and architecture-related metrics (see Section 7.4.5). The following sections are based on the work of Heilemann et al. (2015).

7.4.3 Business level metrics

On the highest level, it can be assumed that if a company wants to improve its business goals like reduced costs and increased profits, it can influence it by increasing the share of products that are based on a common modular system. Therefore, there are general company-specific business goals and the modular system share to be measured within this category.

- General targeted business goals

The transitioning company has to permanently measure those business metrics that it wants to be influenced by introducing the modular system. These metrics are usually related to finance, time to market, customer satisfaction or the like. The influence of modularisation on these business goals has not been within the scope of this work for following reasons:

- Such metrics are company-specific and usually well-known by companies
- The influence of modularisation on such metrics occurs with considerable time delay
- The cause-effect relationship between modularisation and these metrics is highly complicated and yet ambiguous. It is not the focus of this work to come up with a scientifically valid impact model. Significant research work on causes and effects of modularisation has been conducted by other researchers who are referenced in Sections 7.2.4, 7.2.5 and 1.2.
- Even though the directly visible influence of architecture decisions on business goals would be helpful, it is the focus of the presented metrics of this work to keep an established modular system architecture stable over time.

- Modular System Share

After a strategic decision for modular system development has been made, the Modular System Share can be used to monitor the progress of transition toward modularisation of a company. Therefore, it measures the share of products that have been derived from the common modular system in relation to all products under scope. For reasons of better comparability of a product's importance, products are measured in terms of their revenue as weighting factor.

- **Purpose:** It is the purpose of this metric to constantly involve senior management and to keep sustained management attention for the topic. If there are deviations from the goal, senior management can establish powerful counteractions to achieve the agreed modular system share. Moreover, this metric can be used as a common vision which can be broken down into lower level metrics, modular system specifications and guiding principles for local development projects.
- **Calculation of metric:** The metric can be calculated as shown below.

$$\text{Modular System Share} = \frac{\text{Revenue of products based on modular system}}{\text{Revenue of all products under scope}}$$

The result can be interpreted as "percentage from revenue". The scope of the products has to be defined in the context of measurement and can be defined company-specific.

- **Comment:** Particularly during the transition phase, it is possible that there are some mixed products which are only partially derived from the modular system. For such cases, it is important to define how to proceed with such products. For instance, the product could be counted as derived from the modular system when its architecture is in line with the specification or it could be distributed proportionally depending on the share of its modules.
- **Organisational role in charge:** Responsible for this metric is a senior management role that is in charge of the overall product portfolio. This could be a product manager, a product line manager or indeed a top manager (see Figure 73), depending on the size of the company.
- **Time of measurement:** The metric can be reported to management on a regular base (e.g. quarterly) or during management appraisals. With its overarching character, it is not necessary to calculate this metric for each development project (see Figure 74).
- **Example:** The example has its background in a real industrial case study. For reasons of confidentiality, all numbers and figures have been disguised.
    - Input:
      Product portfolio in scope: A, Total revenue of product portfolio A: 3755 T€, Revenue of product portfolio A with products derived from the modular system: 977 T€

- Calculation:
  Modular System Share = 977 T€/3755 T€ = 0,26 = 26 %
- Example for reporting and visualisation: see Figure 75

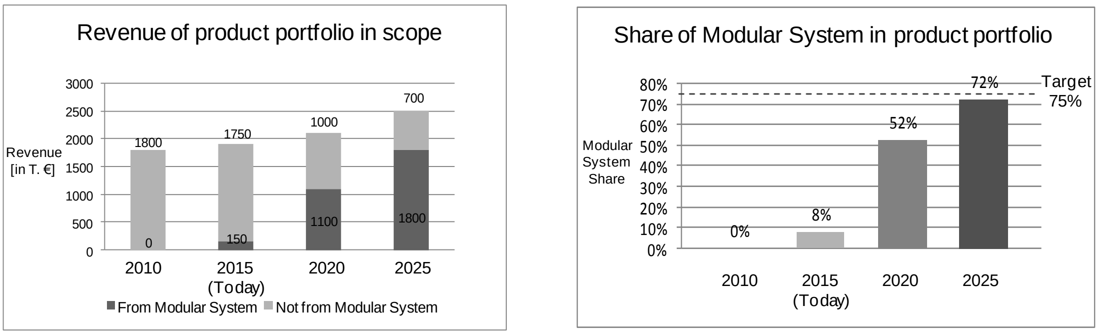


Figure 75: Example for reporting and visualisation of Modular System Share, on the base of Heilemann et al. (2015)

7.4.4 Result-oriented metrics

Within this section, above mentioned metrics are further broken down to an engineering management level with focus on the modular system. On this level it has to be ensured that derivative projects develop their artefacts based on the modular system with the specified amount of commonality, reuse and variety from an overall company perspective.

- Complexity Metric

It is one of the main goals of a modular system to reduce complexity while offering exactly the amount of variety the customer needs. Thus, the Complexity Metric measures the relation between internal complexity that is needed to generate an appropriate amount of external variety and external variety itself. Concretely, it measures the number of parts that are needed to build a certain number of products. In other terms, this is the average number of distinct parts per product. It is the goal of this metric to achieve a low value of the complexity metric.

- **Purpose:** It is the purpose of this metric to show the ability of the modular system to achieve given variety with a minimum amount of internal complexity. Moreover, this metric shall make derivative development projects contributing to an agreed balance between complexity and variety of the overall company, even though this might be contradicting to local goals of the derivative project.
- **Calculation of metric:** The metric can be calculated as follows:

$$\text{Complexity Metric} = \frac{\text{\# of distinct parts in modular system}}{\text{\# of saleable product variants derived from modular system}}$$

- **Comment:** BOMs that can be found in companies are usually not designed for such calculations. For instance, BOMs that have been used for this case study also contained items that were not seen as parts making up a product (e.g. material order specifications, grease). In addition, it was found out that different development sites had totally different BOM rules. In order to provide comparability of the metric, it is necessary that the company defines, for instance, which type of items to include for the calculation. Moreover, companies with strong interface management can extend the numerator of the formula by adding the number of interfaces. The metric can also be adapted to calculate an interface complexity metric. Another aspect to consider is that this metric has to be considered in close relation to its goal value at a specific date because at the beginning of transitioning the metric might indicate bad performance which cannot be directly compared to more mature portfolios who reuse more components or the final goal value.
- **Organisational role in charge:** Modular System Owner (senior engineering manager role, e.g. on a level of a director or head of department) who is in charge of the common modular reference architecture and who tries to achieve an overall optimum for the whole modular system. This role is the counterpart to the local development project manager who tries to optimise one or a small range of products, derived in one single development project (see Figure 73).
- **Time of measurement:** On the one hand, this metric can be reported to the management on a regular base. On the other hand, the metric can be used for more detailed analyses. In accordance with Section 7.4.1 and Figure 74, the forecast of the target value can be given at start of the modular system concept phase. Then, the target value can be confirmed at the end of the modular system concept phase. Moreover, the metric can be used to directly measure the contribution of each project delivering modules and products for the modular system (see Figure 74).
- **Example:** The example has its background in a real industrial case study. For reasons of confidentiality, all numbers and figures have been disguised.
  - Input:
    # of distinct parts in modular system = 620, # of saleable product variants from the modular system = 230
  - Calculation:
    Complexity Metric = 620 parts / 230 products = 2,7 parts per product
  - Example for reporting and visualisation: see Figure 76

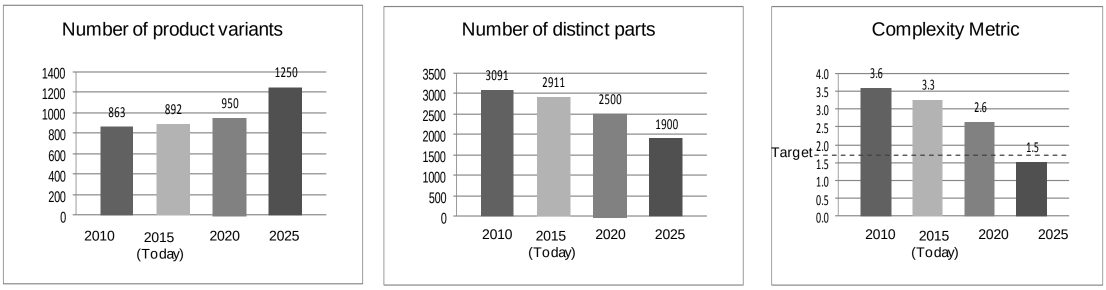


Figure 76: Reporting and visualisation of Complexity Metric, based on Heilemann et al. (2015)

- Module Usage Metric

The metric Module Usage is used to measure whether modules are reused according to their specification. Although it is usually the goal of module variants to be reused in products whenever possible, there are also module variants which have to provide variety and distinctiveness.

- **Purpose:** It is the purpose of the Module Usage metric to contribute to decreasing internal complexity across products through high reuse of module variants which are meant to be common according to their specification.
- **Calculation of metric:** The metric can be calculated as follows:

$$\text{Module Usage} = \frac{\sum_{i=1}^{n} \text{usages of module variant i of respective module}}{\text{\# module variants (=n) of the respective module}}$$

It is assumed that each “abstract” module (e.g. Cylinder Body Module) comprises n distinct module variants (e.g. module variant 1 (i) = Cylinder Body 1 with 45 mm).

- **Comment:** It is not the purpose of each module variant to obtain the highest possible Module Usage. Some module variants have to be explicitly used to generate variety or distinctiveness. For this reason, these modules have to be categorised and a different target value for the metric is assigned to each category. For example, Alizon et al. (2009) differentiate between three module categories: common, variant and unique. Similarly, Jonas et al. (2012) divide modules into “carry-over”, “potential carry-over” and “variant”.
- **Organisational role in charge:** Module Owner (senior engineer or engineering manager, e.g. on a group leader level) who is in charge of all module variants of the respective module. It is the task of this role to make sure that designated module variants can be reused by following the specification of the reference architecture. For instance, module variants can only be combined if they stick to the same interface description (see Figure 73).

- **Time of measurement:** On the one hand, this metric can be reported to middle management and aggregated to modular system level on a regular base. On the other hand, the metric can be used for more detailed analyses. In accordance with Section 7.3.3 and Figure 74, the forecast of the target value can be given after modular system concept phase. Then, the target value can be confirmed at the end of the modular system refinement phase. Moreover, the metric can be used to directly measure the contribution of each derivative project delivering module variants and products for the modular system. There it is suggested that measurement takes place after technical design release (TR) or already earlier when the value of the metric can still be influenced (see Figure 74).
- **Example:** The metric and example have their background in industrial application. However, the following illustration is disguised due to confidentiality reasons (see Figure 77).
  - Input:
    $\sum_{i=1}^{n=3}$ usages of module variant RF = 2 (RF1) + 1 (RF2) + 1 (RF3) = 4,
    # module variants from the respective module RF = n = 3
  - Calculation:
    Module Usage = 4/3 = 1.3 usages per module variant in module RF
  - Example for reporting and visualisation:

Module	Module Category	Σ of Usages	# Module Variants	Module Usage	Module Variants				Overview
Cylinder Body (CB)	Standard	4	1	4,0	CB 1 4 usages				Actual Target min. max.
Valve Unit (VU)	Standard	4	2	2,0	VU 1 1 usage	VU 2 3 usages			Actual Target min. max.
Rod Fastening (RF)	Variant	4	3	1,3	RF 1 2 usages	RF 2 1 usage	RF 3 1 usage		min. max.
Mounting (MO)	Variant	4	4	1,0	MO 1 1 usage	MO 2 1 usage	MO 3 1 usage	MO 4 1 usage	min. max.
Cover (CO)	Unique	3	3	1,0	CO 1 1 usage	CO 2 1 usage	CO 3 1 usage		min. max.

Figure 77: Reporting and visualisation of Module Usage Metric, based on Heilemann et al. (2015)

- Variance Efficiency Metric

Usually, modular systems are related to high variety that can be offered to the customer. However, it is claimed within this work that modular systems are not there to create “endless” variety. Rather, modular systems are used to create a certain degree of variety or flexibility based on a competitive cost level. Modular systems that accommodate too many variants like a one size fits all solution might be prone to failure due to overload of compromises and constant changes. In order to get stable product architectures, it is necessary to get requirements prioritised and fixed as early and stable as possible. Pahl et al. (2007) and Jonas et al. (2012) stress the importance of proper market research and product management in order to be able to focus on profitable market needs as precondition for modular system development. Therefore, the number of requirements has to be restricted by implementing only those requirements by the modular system that promise profitability (Haug, Hvam and Mortensen, 2013). If certain features of a modular system are not needed anymore, they may be phased out in order to reduce internal complexity. For the purpose of getting product management on board of internal complexity reduction, the Variance Efficiency metric shall ensure that the focus of the modular system is on an appropriate amount of profitable products.

- **Purpose:** Fixing interfaces and module specifications is quite an effort for a modular system compared to single product development which only pays off after reuse of the respective interface or module variant. It is the purpose of the Variance Efficiency Metric to prompt product managers to do proper market analyses in order to come up with stable and profitable requirements to be implemented by the modular system. Moreover, by adding a limited scope to product variants to be derived from the modular system, an "over-dimensioned" modular system is prevented. Another point is that by constantly monitoring the product portfolio, product management is triggered to phase out those features from the modular system that are not profitable anymore. In consequence, it is the purpose of this metric to get a lean and stable modular system instead of an overloaded, unprofitable and unstable compromise.
- **Calculation of metric:** The Variance Efficiency metric can be calculated as follows:

$$\text{Variance Efficiency} = \frac{\text{\# of product variants sold with sales volume} > X}{\text{\# of final product variants derived from the modular system}}$$

  The sales volume "X" which can be seen as critical profitability target value for each product variant to achieve has to be determined for the specific context of each modularisation project under consideration. It is also possible to take sales revenue or another value as critical target value X. Moreover, it is possible to define the target value X on the base of an ABC-Analysis or other profitability analyses as indicated by Rennekamp (2013) in figure 2-9 on p. 35 and in figure 2-11 on p. 38. For example, the critical target value could be divided into three different categories where the C-category is tried to be removed from the portfolio while the focus of the modular system is on the A-category. In such a case, the B-category could be a category for unavoidable product variance with relatively low volume that can be generated by efficiently combining existing module variants with some simple "variant" module variants.
- **Comment:** During market and concept phase of the modular system life cycle it is suggested that potential features that shall be implemented by the modular system

have to be backed with expected sales volume or sales revenue. These values can either be taken as target value or baseline for the calculation of this metric.

- **Role in charge:** Responsible for this metric is a role that is in charge of the overall product portfolio. This could be a product manager, a product line manager or the like (see Figure 73).
- **Time of measurement:** This metric can be forecasted at project start of the modular system project and confirmed after conceptual stage of the modular system. Actual values can be measured and reported during yearly product review meetings under close collaboration of derivative product development projects. Data which is taken from derivative projects has to be aggregated onto modular system level (see Figure 74).
- **Example:** The following figures show an example for the application of the metric. The figures have been disguised due to confidentiality reasons. Nevertheless, the background of the example is from an industrial case study.
  - Input:
    Critical target value X for B-Category: sales volume 500 pcs < X < 2500 pcs,
    # of product variants sold with sales volume X = 8,
    # of final products variants derived from the modular system = 34
  - Calculation:
    Variance Efficiency (B-Category) = 8/34 = 0,24 = 24 %
  - Example for aggregated reporting and visualisation of Variance Efficiency metric (see Figure 78)

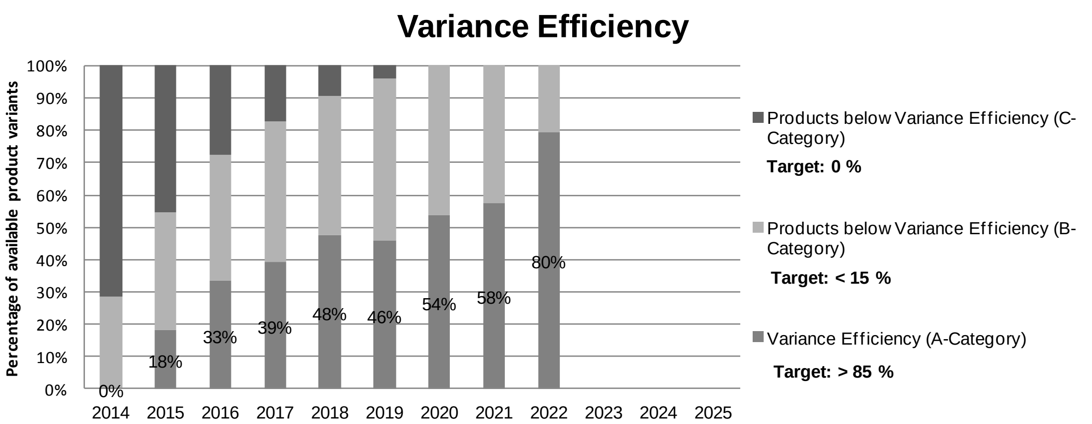


Figure 78: Example for reporting and visualisation of Variance Efficiency Metric, based on Heilemann et al. (2015)

7.4.5 Architecture-related metrics

The literature review of this chapter showed that product architecture metrics have been researched frequently by numerous researchers. At the same time, the usefulness of measuring product architectures or modularity itself has been questioned because modularity should not be an end in itself. Gershenson et al. (2004, p. 37) state that answering how modular a product is, for example with an abstract number, might be unimportant. They state that this is particularly true if the targeted benefits of modularity are measured and if the company follows a dedicated modularisation process with the aim of improved product architectures.

However, differently to previous research, it is the purpose of this section to support stability of the common modular reference architecture by ensuring that local derivative development projects adhere to the overall goals of the company concerning modularisation transition. Therefore, in addition to measuring the results of modular system development on a higher level (which was described in previous sections), it is necessary to support the modular system architecture at the point where it is physically created – in the later modular system life cycle during derivative development projects.

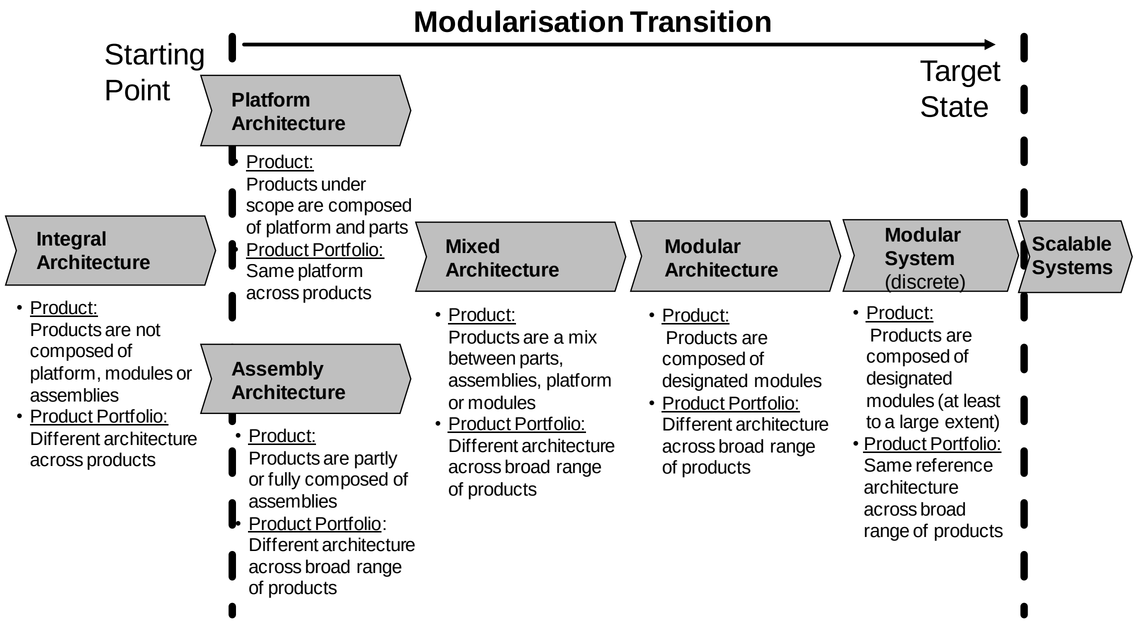


Figure 79: Two dimensions of modularisation transition from a technical perspective

In order to measure modularisation transition, it is important to consider two dimensions. The first dimension deals with the determination of the *degree of modularity* of products. A Degree of Modularity Metric for this dimension is proposed below. The second dimension measures *architectural commonality* across a product portfolio. An Architectural Commonality Metric for this dimension is presented below. It gets evident from Figure 79 that during modularisation transition, the architecture of single products and the common architecture across the product portfolio form the distinct character for each stage. For instance, with integral architectures, single products have no modular architecture and there is no architectural commonality across different products. On the other hand, the modular system architecture requires products to be made up of modules and it requires the architecture to be the same across the portfolio (see Figure 79).

The next sections will show how the *degree of modularity* of products and *architectural commonality* of a product portfolio can be measured.

- Degree of Modularity Metric

This metric determines the degree of modularity by measuring how items are organised within single products or product portfolios. Therefore, it measures whether products are built of *designated* module variants and to what degree items of the BOM are organised into module variants.

- **Purpose:** It is the purpose of this metric to guide engineers of derivative development projects toward the required modularity degree of their products that has been given as target from the modular system development project. This prompts engineers to organise items into designated module variants. Once this metric is aggregated to modular system level, engineering managers can monitor the progress in modularity during modularisation transition. If the metric detects deviations from the target degree of modularity, these deviations can already be fixed on project level.
- **Calculation of metric:** The metric can be calculated on single product level, but also on product portfolio level. If the metric is calculated on portfolio level, the BOMs of the products have to be combined. In practice BOMs are usually not designed to do such kind of analyses. Therefore, the company has to define in detail what it considers as item and what not to consider as item. In particular, this is necessary if BOMs of different development sites are compared. For instance, during case studies it was frequently discussed which parts to consider and whether to consider assemblies and modules themselves as some kind of "virtual" items. The items for assemblies and module variants are rather organisational constructs and not real physical items like parts or components which are actually causing increased complexity levels.

$$\text{Degree of Modularity} = \frac{\text{\# of items in first level module variants}}{\text{\# of items in product}}$$

- **Comment:** The Degree of Modularity metric can be used for calculations on different levels. For instance, it can be used to calculate modularity on system level, on product level or on module level. For example, if the module is applied on module level, it calculates the degree of modularity of the module, separately from the higher level system or product. This is also true for the application of the metric on system and on product level.

- **Role in charge:** The design engineer in charge of the product structure or BOM is responsible for this metric. This role can make corrections to the product architecture if it does not meet overall company degree of modularity targets before the product is launched (see Figure 73).
- **Time of measurement:** This metric can be applied for all products of a product development project contributing to the modular system. It is suggested to take the measurement prior to the technical design release (TD) of the derivative project (see Figure 74).
- **Application note:** For the purpose of modular system development, the Degree of Modularity of a product range is only half the truth. It is important that modularity within products is in line with the reference architecture so that modules can be combined and reused. Therefore, it is strongly advised to combine this metric together with the metric of the next section.
- **Example:** An example of this metric is combined with the example of the next metric.

- Architectural Commonality Metric

The Degree of Modularity like it is calculated by the previous metric shows how the company is transitioning toward modularity. However, it does not show whether products are based on a modular system. Therefore, Figure 79 shows that the second dimension of modularisation transition is Architectural Commonality across products. This metric shows whether a set of given products are in compliance with a predefined common reference architecture which is valid for the product portfolio under scope. Thus, it measures the relation of module variants of a product that are in line to product architecture specifications to the total number of module variants of the product.

- **Purpose:** Design engineers in local product development projects are inclined to optimise products based on the present development project. Taking care of overall modular system specifications is rather seen as obstacle and time consuming. Thus, it is the purpose of this metric to control derivative product development projects that their architecture is in line with the architecture of the common modular system. Even though, this will not help to achieve the local optimum of the project, it will help to improve extrinsic motivation of local engineers to pursue global modular system goals. Moreover, during the case study local engineers claimed that their products are already modular. However, the calculation convinced these engineers that the scope of their modularity is quite narrow and insufficient to generate the required complexity level.
- **Calculation of the metric:** It is possible to calculate this metric for each product separately and to aggregate the results later on or to combine the BOMs of the whole product portfolio.

$$\text{Architectural Commonality} = \frac{\text{\# module variants in line with common modular reference architecture}}{\text{\# of module variants of the product}}$$

- **Role in charge:** The design engineer in charge of the product structure or BOM is responsible for this metric (see Figure 73). This role can make corrections to the product architecture if it does not meet architectural commonality targets before technical design release (TD).
- **Time of measurement:** This metric can be applied for all products of a product development project contributing to the modular system. It is suggested that the targets for this metric are given by the modular system project and that the actual value of this metric is measured after concept phase (C) and technical design release (TD) of the derivative product development project (see Figure 74).
- **Example:** An example of this metric is combined with the example of the previous metric in the following sections.

- Examples for architecture related metrics
  - Example 1: Sample calculation
    - Data:
      # items in first level module variants = 324, # items in product = 350,
      # module variants in line with reference product architecture = 3,
      # module variants of product = 18
    - Calculation:
      Degree of Modularity = 324/350 = 0,93 = 93 %,
      Architectural Commonality = 3/18 = 0,17 = 17 %
    - This sample shows that the considered product seemed to design engineers at already quite modular. However, the Architectural Commonality showed that the product rather possessed local modularity than modular system commonality.
  - Example 2: Figure 80 and Figure 81 show the principle how companies can measure their performance in transitioning toward modular system development from an architectural perspective. The different types of architectures comply with the categories given in Figure 79.

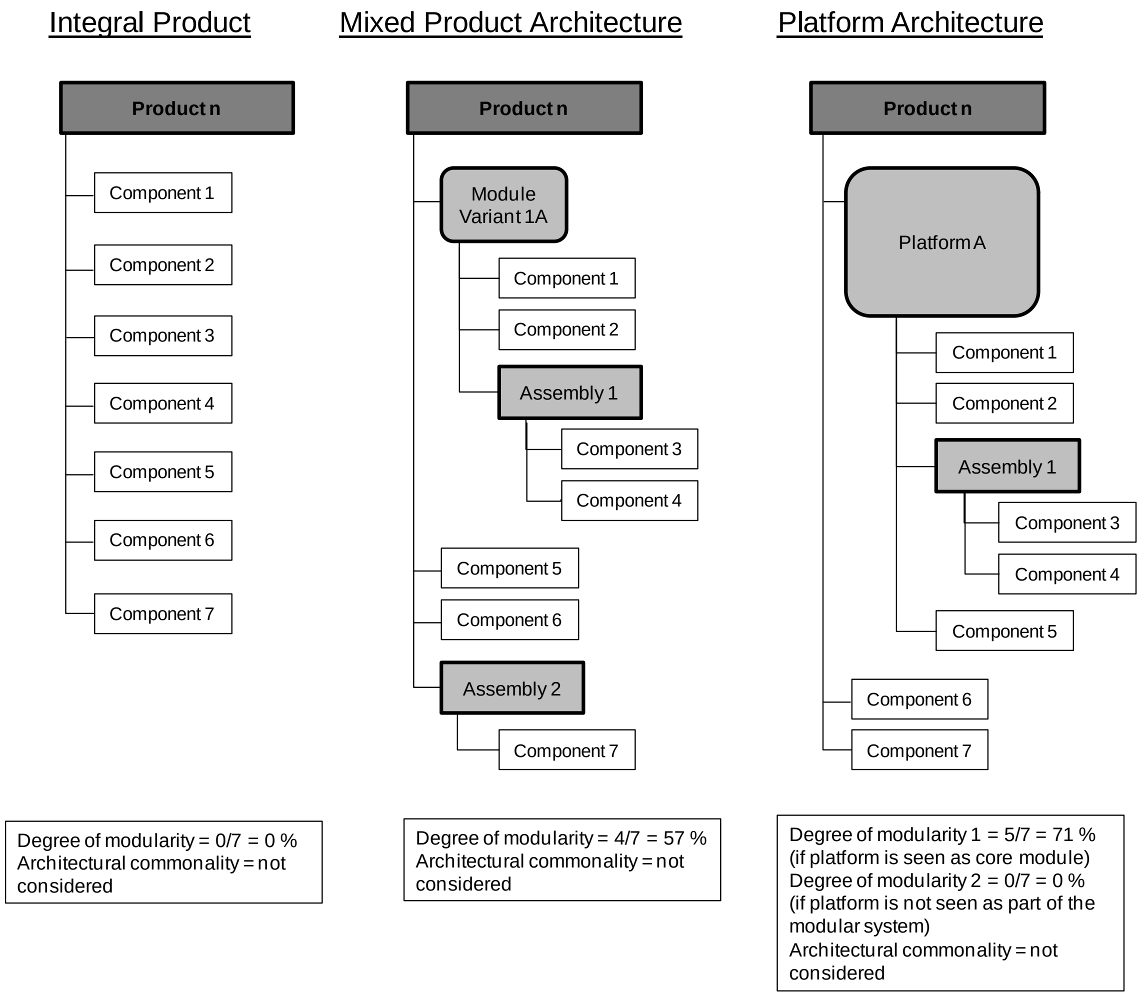


Figure 80: Examples for product architectures and corresponding architecture metrics (part 1)

Figure 80 shows the first part of examples for an integral architecture, for a mixed architecture and for a platform architecture. For all examples, the Degree of Modularity Metric was calculated accordingly. For instance, the integral product is not organised into modules. Consequently, its degree of modularity is at 0 %. The product with the mixed architecture is partly composed of modules. Four out of seven "physical" components are assigned to first level module variants. This makes a degree of modularity of 57 %. The platform architecture can be considered in two different ways. Firstly, if the platform is not seen as part of the modular system, the degree of modularity of the product is at 0 %. Secondly, if the platform is seen as some kind of core module, five parts out of seven are considered as organised into a module. Hence, the degree of modularity of this architecture is 57 %.

As only one product was considered for each product architecture type, it was not possible to calculate the Architectural Commonality Metric for these examples.

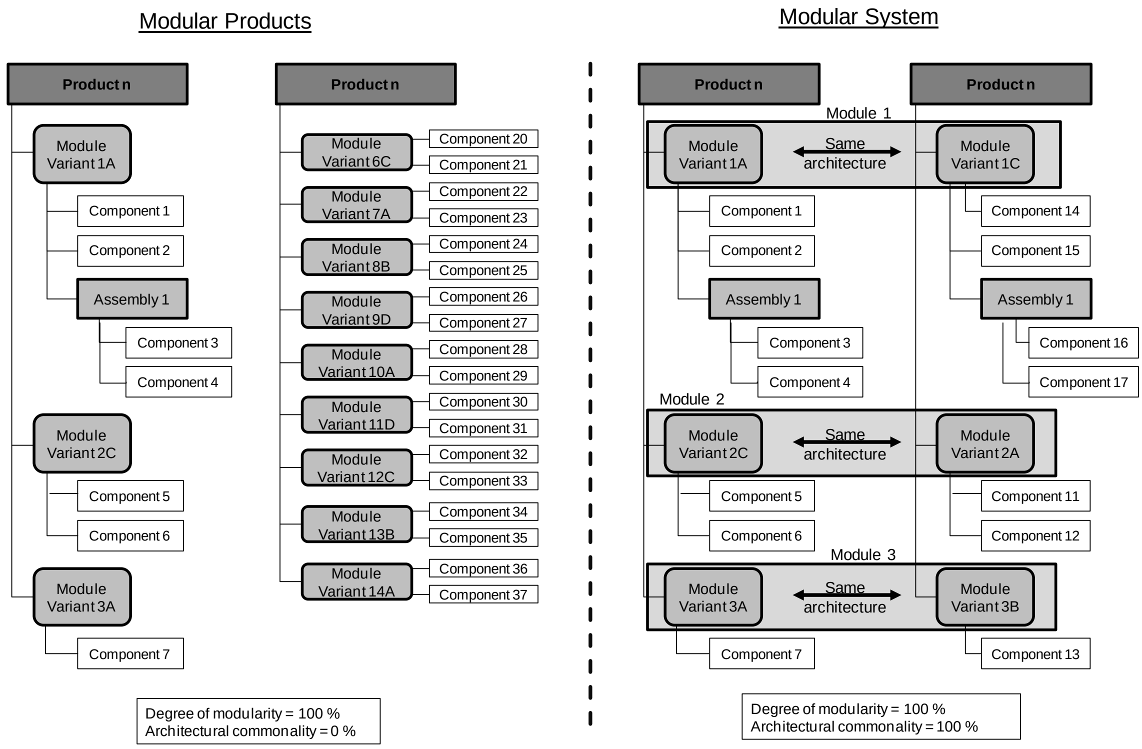


Figure 81: Examples for product architectures and corresponding architecture metrics (part 2)

Figure 81 shows fully modular products. The left hand side of the figure shows two products which are both fully modular, but have different product architectures resulting in architectural commonality of 0 %. The modular structure of these products is not based on the same common modular reference architecture. This gets evident by looking at the description of the modules. For instance, Module Variant 1A is not based on the same architecture specification as any other module of the other product (e.g. Module Variant 5). Thus, modules cannot be interchanged and reused between the two products. It has to be clear that this is a simplified example based on the descriptions made within Figure 81. Usually it is not possible to judge architectural adherence by comparing naming of modules. The ability to perform such analyses (e.g. identifying architectural conformity of modules) depends on the information available in IT-systems. The basis for such considerations will be further described in Chapter 8.

The right hand side of Figure 81 shows two products derived from a designated modular system. The products share the same modular architecture and modules can be easily shared and reused across products. For instance, Module Variant 1A has the same architectural specification as Module Variant 1C. This is also true for other respective modules of the shown products. Thus, degree of modularity and architectural commonality are both at 100 %. In order to being able to perform such analyses efficiently, an IT-integration approach like it is described in Chapter

8 is required. Otherwise, it is not directly obvious whether module variants violate or adhere to their architectural specification.

- Example 3: Another sample visualisation, each metric calculated separately - Figure 82 shows the derivation of the Degree of Modularity Metric with 57 % on the left hand side and Architectural Commonality Metric with 100 % on the right hand side.

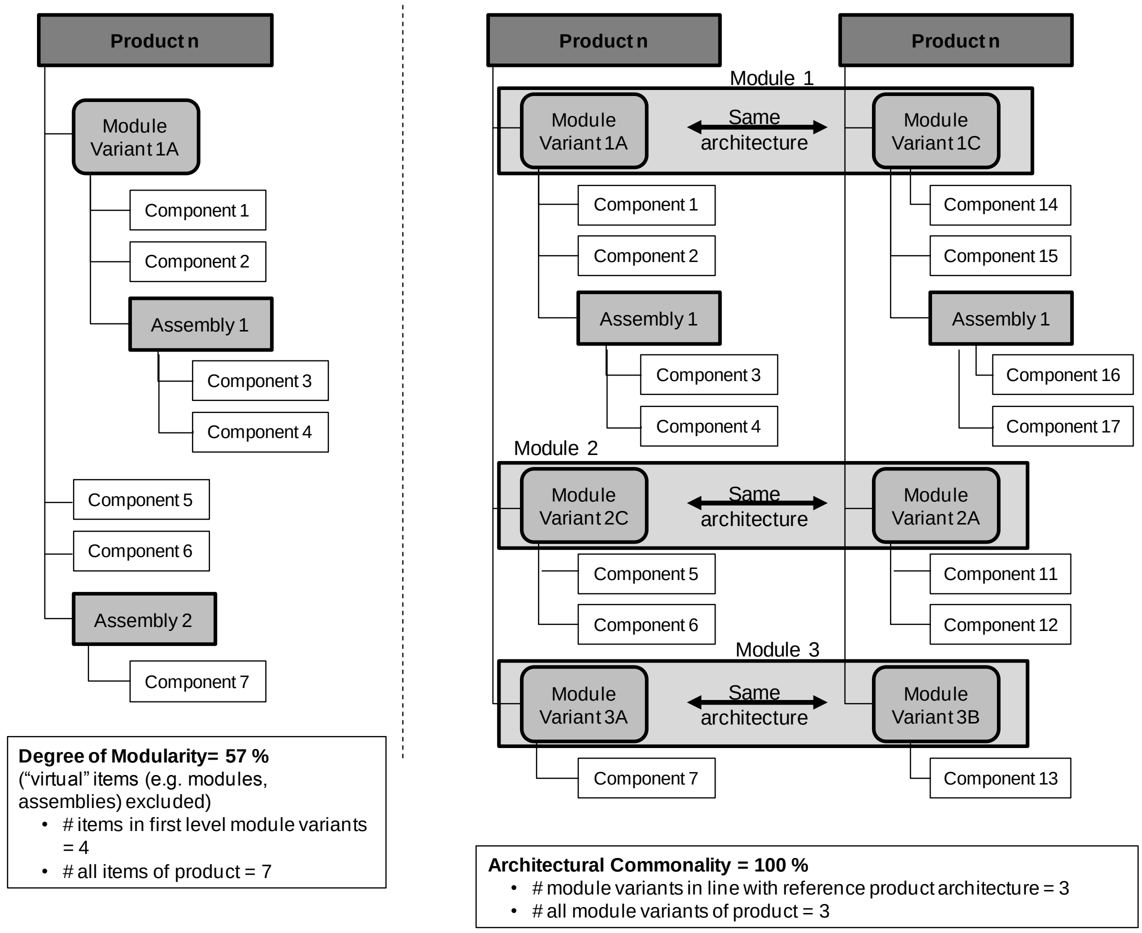


Figure 82: Depiction of rationale behind Degree of Modularity Metric and Architectural Commonality Metric (Heilemann et al. (2015))

- Example 4: Calculation of the modularisation metrics in industry caused various and lengthy discussions about *comparability of different products and architectures*. This is an important factor that has to be considered during metric calculation. For instance, different products might not be directly comparable if modules of one product are in-house built and modules of the other product are outsourced. Thus, there might be a different degree of vertical integration across development sites and regions. Moreover, DFA activities quite often lead to an integration of parts within products or modules. Thus, the overall number of parts decreases. Such kind of complexity reductions are not the result of modularisation activities.

Therefore, products might have to be made comparable or "normalised" in order to obtain fair evaluation.

Figure 83 shows different examples of normalisation factors which can be applied to normalise product portfolios. Factors that can be used for normalisation may include length of BOM, depth of BOM, "virtual" items (e.g. assemblies and modules), outsourced items or integrated items. Other means of normalisation may include omissions or projections. The normalisation factors to be applied and the means of normalisation have to be specified in detail for each company in which the metrics are applied.

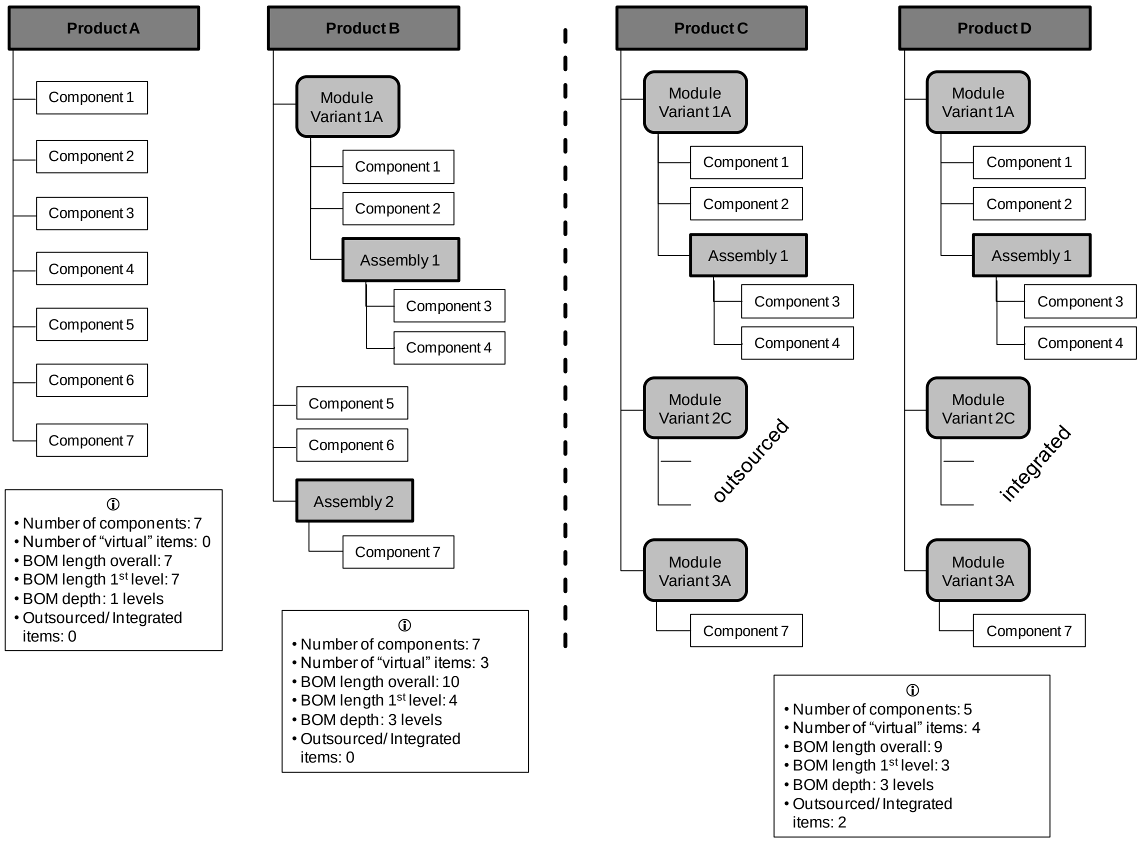


Figure 83: Product architectures, their different characteristics and possible normalisation factors

For instance, the BOM of Product A of Figure 83 consists of seven items of which are all physical components. The BOM of Product B has in total ten items. Out of these ten items, there are seven real physical components and three "virtual items" (i.e. one module variant and two assemblies) which cannot be "grasped" if the physical product is disassembled into its physical components. Thus, the number of BOM-items is different between Product A and Product B, but the number of physical components equals seven for both products. Another comparability problem gets evident if Product B and Product C are compared. Product B has seven physical components. Product C contains only five physical components. However, this reduction is not due to modularisation-related complexity reduction, but due to outsourcing components 5 and 6 of Product B.

All such factors have to be properly considered before drawing conclusions on modularisation transition based on the presented modularisation metrics. Moreover, it might be helpful if specific BOM-rules for the creation and comparability of BOMs are established within companies.

7.5 Validation of the metrics

The metrics were developed and validated iteratively with the primary case company. Means of validation comprised benchmark studies with other companies and other researchers, discussions within academia and industry, sample calculations, experiments decision process for implementation of metrics at the company and application in industry. The metrics presented within this chapter have been validated based on their requirements from Section 7.3.1. Moreover, a specific focus was laid on applicability and relevancy of the metrics as shown in Table 22 of Section 7.3.3: Calculation, Application, Purpose and Modular System Development Life Cycle.

7.5.1 Validation on industrial products

The metrics have been constantly refined and validated based on real products from industry. For this, the metrics have been used to analyse non-disclosed BOMs and other material master data of the primary case company. This was helpful to gain insights on the calculability of the metrics and whether the metrics are supportive to understanding product characteristics based on architectural considerations.

Table 23: Disguised example for calculating modularisation metrics

Metrics	Dimension	Target (of) Modular System	Development Project 1	Development Project n	Overall Portfolio of Modular System (time: n)
Modular System Share	**[%]**	**80%**	**88%**	**61%**	**67%**
Revenue of all products based on modular system	Mio. €	-	1,4	3,3	4,7
Revenue of all products in scope	Mio. €	-	1,6	5,4	7
Complexity Metric	**-**	**1,5**	**1,5**	**2,2**	**1,9**
# d i s t i n c t parts in modular system	-	-	112	263	375
# of all saleable product variants derived from modular system	-	-	74	120	194

Metrics	Dimension	Target (of) Modular System	Development Project 1	Development Project n	Overall Portfolio of Modular System (time: n)
Module Usage: Module Engine (Standard Module)	**-**	**37**	**37**	**22**	**26**
# usages of all module variants of the module	-	-	74	110	184
# module variants of the module	-	-	2	5	7
Variance Efficiency	**[%]**	**85%**	**88%**	**68%**	**76%**
# product variants produced with sales volume > 2000 pcs.	-	-	65	82	147
# final products under scope	-	-	74	120	194
Degree of Modularity	**[%]**	**90%**	**93%**	**77%**	**84%**
# items in first level module variants	-	-	392	403	795
# all items of products	-	-	421	522	943
Architectural Commonality	**[%]**	**100%**	**100%**	**45%**	**69%**
# module variants in line with reference product architecture	-	-	83	48	131
# all module variants of product	-	-	83	106	189
Complexity Cost in project	**[€]**	**n.a.**	**301000**	**700000**	**n.a.**
# number of new parts in the project	-	n.a.	86	200	n.a.
Complexity Cost per part number	[€]	3500	3500	3500	n.a.

Table 23 shows the calculation of the presented modularisation metrics. The figures have been disguised and sanitised to maintain the confidentiality of the raw data, but the process has been done by applying confidential multipliers. The original spreadsheet that was derived in industry and used to conduct experiments with the metrics contains more than 12000 items which are related to 34 different products from different business areas. The original spreadsheet is more extensive than Table 23 as it contains further numerical fields which are used to answer following questions:

- What is the degree of modularity and how can it be calculated?
- How is the resolution or granularity of modularity?
- How is the profile of the product?
- How can BOMs be controlled or normalised in order to achieve comparability?
- How to measure architectural similarity of products?
- How to measure functional independence of modules?

- What are factors to be considered during modularisation IT-Integration?

The main findings of validation via experiments are twofold. Firstly, the metrics as they are presented in Table 23 are indeed easily calculable by using exports of standard PLM or ERP systems and some standard spreadsheet programmes. This is particularly true if the IT-Systems are adapted according to the procedures described in Chapter 8. Secondly, the metrics are indeed relevant and lead to correct and objective “statements” for modularisation decision makers.

Table 23 shows that for each metric a target which has to be achieved by the overall modular system has been set. These targets have to be achieved by different derivative development projects. In this case, the results of derivative “Development Project 1” and “Development Project n” are shown. The right column shows the metrics for *all* products under scope of the modular system. In order to obtain these metrics, data and BOMS of all derivative development projects of the modular system have been combined. Therefore, Table 23 can be used to illustrate two further findings.

Firstly, it shows how different development projects influence the performance of the overall modular system. The first derivative project “Development Project 1” performs quite well and achieves the targets set for the modular system. After some time, “Development Project n” underperforms and achieves poor modularisation results, not only for its isolated project, but also for the overall modular system. Thus, engineering designers can see at first glance from Table 23 that the first development project contributes to the stability of the modular system and the “Development Project n” deteriorates the stability of the modular system by deviating from its global goals.

Secondly, experiments have shown that it is always necessary to bring the development of derivative products into the context of the overall modular system and its targets. While single projects may well achieve good results for most of the metrics, it is not ensured that this will foster the modular system. For instance, if two derivative projects achieve 50 % commonality for their isolated projects, this does not mean necessarily that the overall modular system at a certain time also achieves 50 % commonality. This is only the case if the commonality of different projects is based on the same reference architecture. If the commonality of the projects is considered separately and if there is no match across projects, this could mean a significantly lower commonality down to 25 % for the overall modular system. Therefore, the summarising cross-project column “Overall Portfolio of Modular System (time: n)” has been introduced in Table 23 to show cross-project performance and its development toward the target. This example highlights the significant importance of the newly introduced Architectural Commonality Metric.

In sum, the metrics have been validated to be capable for a) comparing BOMs of different products, b) showing the difference between non-modular and modular products (i.e. before and after transitioning), c) showing to what extend the development project contributes to the targets/plans of the modular system and d) showing how product variants are contributing to the performance of the overall modular system.

7.5.2 Validation of applicability and relevancy for industry

After initial modularisation metrics were developed and scrutinised, they were first tested and implemented into Primary Case Company – Project C (see Appendix B). The following three metrics have been validated in expert interviews and implemented into product and project assessment of the business unit in this case: *Variance Efficiency Metric, Complexity Metric, and Module Usage Metric*.

Then, the metrics were again validated during several cycles of expert interviews within the primary case company before they were introduced in the company over the overarching modularisation implementation program (Primary Case Company – Project A). Following metrics were implemented: *Modular System Share Metric, Module Usage Metric and Complexity Metric*. These metrics were set out to be implemented within the Primary Case Company.

The product architecture related metrics (*Degree of Modularity and Architectural Commonality Metric*) have not been officially designated and implemented within the Primary Case Company. However, in the context of a so-called product profile that was used to analyse existing products, the architecture related metrics have been evaluated by industry. The reason why they were not officially implemented into the reporting scheme was that they are not on a business- or result-oriented level (see Figure 73). Managers were rather interested in the results to be expected of modularisation and not in the technical architecture details. Thus, the architecture-related metrics have been seen as less relevant for reporting purposes in industry by managers. Nevertheless, the value of the metrics for design engineers was well-recognised. These metrics help to change the understanding of engineers and to make them thinking in terms of product-overarching modular system architectures instead of thinking in terms of single integral products. Moreover, the metrics were supportive in convincing engineers that their alleged modular products have nothing to do with a modular system that is reused across a wide range of products.

Given these points, the metrics have been proven to be both applicable and relevant for industry. Moreover, the metrics are seen as capable from an industrial point of view to fulfil the requirements of Section 7.3.1.

In the final analysis, the metrics presented within this work fulfil the combined research and industry criteria from Table 22 of Section 7.3.3: Calculation, Application, Purpose, Modular System Development Life Cycle.

7.6 Discussion

7.6.1 Fulfilment of metric creation requirements

Table 20 shows requirements established from the shortcomings of existing approaches and from industry on “what” to measure and “how” to measure and monitor progress during modularisation transition. All requirements of “what” to measure are fulfilled by modularisation metrics established and detailed above. The assessment is either done on modular system level or it is described how aggregation from derivative development projects onto modular system level can be done. The direct results of modularisation transition increased external variety, internal complexity and reuse are fully covered by the metrics. In addition, the description of the modularisation metrics advises how modularisation roles and different phases of the modular system development life cycle can be assessed.

The second requirement category of Table 20, "how" to measure modular systems can also be seen as fulfilled. It is proposed that the metrics are applied several times during the life cycle, either regularly or during events such as milestone reviews of the modular system life cycle. It is of special importance to the purpose of this work that there is a strong focus of the developed requirements on post-architecting phases of modular system development. It was these phases were identified as most vulnerable and prone to failure. Moreover, it has been shown that the metrics can be easily combined into a metric system that uses quantitative criteria instead of estimations or qualitative data. The assessment can be done by local business units themselves, but the results can be aggregated onto higher level in order to achieve a compressed overview. Although the metrics can be applied as standalone solution, it is strongly advised to integrate them into already existing company assessment like milestone reviews or quality gates in order to contribute to the sustainability of modularisation transition.

It has been seen as important precondition by industry that the metrics can be calculated automatically with data from standard IT-systems. All modularisation metrics suggested within this chapter have been designed to be computed efficiently and tested if they are computable with data available from standard PLM and ERP within the case company or IDES.

The next section will move on to describe the differences and similarities to existing research.

7.6.2 Implications for other researchers

- Application and purpose of metrics

Table 22 shows a rough overview of the relation of existing metrics to the requirements of this research work. It is quite evident that most of the metrics struggle to be efficiently computable. It is assumed that the prevailing research direction tries to include more and more factors to be measured with modularisation metrics (e.g. see Appendix F). While this might be advantageous for purely academic purposes, requirements of industry are not met by this development. Metrics that shall be applied in industry have to be efficiently computable, especially if they are applied on a large range of items. For industry, there is no use to introduce metrics that would need a considerable amount of additional resources just to calculate them regularly. The metrics developed within this chapter might not be as "fancy" and intricate as other existing metrics, but these metrics are well applicable and helpful which has been proven in industry. Surprisingly, this is a new finding compared to contemporary research where this has been identified as a clear gap. Thus, it is suggested that this research work helps to redirect current research toward the real needs of its customers in industrial practice.

Another innovative aspect of the modularisation metrics presented within this work is their purpose of application. Most of existing metrics from literature help to improve products during architecting in the concept development phase. Thus, they are used to support designers in selecting between different product architecture alternatives or during reengineering of existing products. In contrary to existing work, the metrics of this work start to be applied after concept phase, i.e. after the architecture for the modular system has been established. Hence, it is the purpose of the modularisation metrics to enforce the central modular system architecture in derivative development projects.

Other existing metrics like those that aim at commonality or on financial aspects are used to monitor general performance of a modular system. This is similar to the metrics of this work. However, unlike other studies the metrics of this work have a clear focus on whether development projects align to the common modular reference architecture, how stable the modular architecture is and how the modular system develops compared to its plans.

- Different modularisation metric categories

Modularisation metrics can be divided into three different categories: architecture-related metrics, result-oriented metrics and business level metrics.

Architecture-related metrics from literature tend to determine modularity of a product based on functional structures, functional-physical relations and physical interaction between elements. Thus, it is the purpose of these metrics to determine the degree of modularity of a product based on the abstract definition of “modular”, in order to tell engineers how modular their product is. In comparison, architecture-related metrics of this work measure the product architecture based on the concrete organisation of items within products. This has the purpose to tell engineers how far is their current product architecture away from the common modular reference architecture, based on data that is in PLM or PDM systems. Moreover, it tells engineering how stable the architecture is and how it develops toward modular system development over time.

Result-oriented metrics either focus on commonality, variety or strategic intent of the product architecture. Strategic factors have not been considered by this work, as they are mainly used to establish product architectures and are rather no driver to monitor the stability of common modular reference architectures. The result-oriented metrics of this work are similar to commonality and variety metrics, they heavily depend on those metrics from literature that take data from standard IT-systems as input. A remarkable difference in the usage of variety metrics of this work is as follows: Literature suggests that modular systems are used to generate a high variety of products. This is in fact true and considered by the Complexity Metric of this work. However, the Variance Efficiency metric was introduced to make it clear to product management that they cannot have whatever they want but that they have to focus on the most promising product variants in order to get a lean and stable modular system. Further reasons for the difference to classical commonality metrics from literature are as follows:

- Possibility to distinguish between really advantageous commonality and possibly unhelpful commonality that was created for the sheer sake of commonality
- Categorising modules whether they are designed to create commonality, variance, flexibility or options
- Focus on reuse targets which in turn aims at commonality
- Comparability problem when a module can be defined as common: For instance, if it is common across all products, if it is common across 80 % of the products or if it is common in more than 50 % of the products? As it is the main aim of modularisation to combine different module variants and not installing a common platform across all products, the focus of the metrics presented within this work has been slightly different compared to the commonality metrics of literature.

Business level metrics either focus on financial aspects, value or other factors that are of importance for the business. While the modular System Share presented within this work is just an indicator of the progress of the Modularisation Programme, literature has a high number of publications that deal with the link between modularisation and its impact on the financial situation of the company. This work recognises the importance of financial metrics to measure modularisation transition, but such metrics would have their origin in the world of finance. Moreover, even those metrics that establish the link between the modular world and business objectives which are popular in literature have been beyond the scope of this work due to following reasons:

- Actual linkages between product architecture characteristics and costs have to be determined case-specifically.
- Proposed cost models for linkages can only be established in tedious and time-consuming processes if they shall be of appropriate value.
- If cost estimation shall be reasonably correct, the product must be at an appropriate maturity level.

In addition, there is another reason why this work handles the link between modular systems and costs cautiously. Although it is seen as highly practical important, this study found out that the links between transitioning and business objectives are unpredictable and intricate. Moreover, it was found out that engineers could not be convinced with cost estimations for the far future that do not affect their own project. For example, for an engineer it is sufficient to understand that he shall contribute to higher commonality and he is appraised based on generated commonality. If the engineer is told that it is better not to use the optimised part that would give direct cost savings to his project but to use an oversized standard module variant that is more expensive for him, but cheaper for the overall company, this is of no benefit for him as long as this is not reflected by his project budget. Thus, as long as there is no change in the costing system, there is only moderate additional value in linking modularity to cost.

7.6.3 Implications for industry

It is assumed that the presented modularisation metrics have considerable implications for industry. First, an analysis of existing modularisation metrics has shown that most of the metrics from practice could not be calculated efficiently due to the input information they require. Moreover, it seems that some metrics that were already available have mere academic purposes. In contrast, the presented modularisation metrics of this work were developed based on requirements form industry. This included efficient computability and high relevancy for modularisation transition. The presented metrics constitute innovative and validated support for companies that transition toward modular system development and seek stable product architectures over a prolonged period. Exactly this is the field where previous companies embarking upon modularisation transition failed and where support from literature or other benchmark organisation was rare. Thus, it is suggested that companies which apply the presented metrics with a particular focus on post-architecting phases have a more determined modularisation transition than without those metrics.

7.6.4 Limitations of metrics

Application of modularisation metrics in industrial projects showed that the value of the metrics even could become worse after modularisation transition. This is not because of the metrics themselves but because of the facts that a) the modular system is developed in parallel to other products which have so far not been phased out, b) the overall complexity of the company or business environment is increasing at a rate that is more than the modular system can reduce (in such a case, the modular system would just lower the level of growth) and c) metric values are worse for new products than for mature products because more mature products already had the chance to incorporate reuse. This matter has not been seen as problem by practitioners but it has to be considered and communicated. Moreover, that is the reason why to set overall goals (according to the roadmap) and strictly pursue them during transitioning. Thus, it is not the mere purpose of these metrics to improve products or to show comparable benefits of modularisation but to achieve the goals which have been set during the modular system life cycle.

Another limitation of this study is that its direct application has been limited to one case company, even though it has been validated within different departments and products from different industries within that case company. Thus, future research should find out how far the results of this chapter are generalisable and applicable to other industries.

7.7 Summary

This chapter aimed at achieving research objective RO3 by developing metrics for transitioning toward modular system development in industry.

The literature review of this chapter showed that there is indeed a large body of literature available on modularisation metrics. If these existing metrics are compared to the requirements of industry, which have also been derived in this chapter, it becomes evident that existing metrics can be improved concerning their computability, applicability during different modular system life cycle phases and purpose to support stable modular reference architectures.

In order to give improvement in exactly these fields, the developed modularisation metrics have been designed for efficient computability and application during modular system evolution and change phase in order to make derivative development projects pursuing goals of the common modular system. Therefore, metrics have been developed for three different levels. Firstly, the Modular System share involves senior management to constantly control the transition of derivative development projects toward modular system development from an overall perspective. Secondly, result-oriented metrics are applied on an engineering management level and influence projects to work for overall communality, reuse and variety goals of the modular system. Finally, product architecture metrics measure whether design engineers adhere to modular system specifications and therefore contribute to stable architectures during the modular system life cycle.

Altogether, it is suggested that the presented modularisation metrics enhance transparency & information and extrinsic motivation of involved managers and engineers to strongly contribute to stable modular systems during the modular system life cycle, especially during post-architecting phases.

The Qualitative Study in Chapter 5, the Assessment Framework in Chapter 6 and requirements for computability of metrics within this chapter indicated that information availability about the modular system is a critical factor within transition companies. As a consequence, the next chapter will study information representation of common modular reference architectures.

8 Modularisation information provision

The Qualitative Analysis in Chapter 5, (see e.g. Figure 55) has shown that the evolution and change phase of the modular system development life cycle is the phase which is fraught with fundamental issues which endanger the stability of the modular system. Figure 55 in Chapter 5 also shows that one major reason for this eroding stability is a lack of information and transparency in derivative development projects about global modular system specifications. Figure 56 in Chapter 5 (Qualitative Analysis) and Figure 59 in Chapter 6 (Modularisation Assessment Framework) have shown that a part of these issues can be removed by providing company-wide information about modular systems and the corresponding common modular reference architecture (i.e. modularisation information).

While the previous chapter has described how modularisation metrics help to gain motivation and transparency, this chapter will fully focus on transparency and information provision of the modular system during post-architecting phases. How this chapter relates to the other chapters and research objectives of this research work is shown in Figure 84.

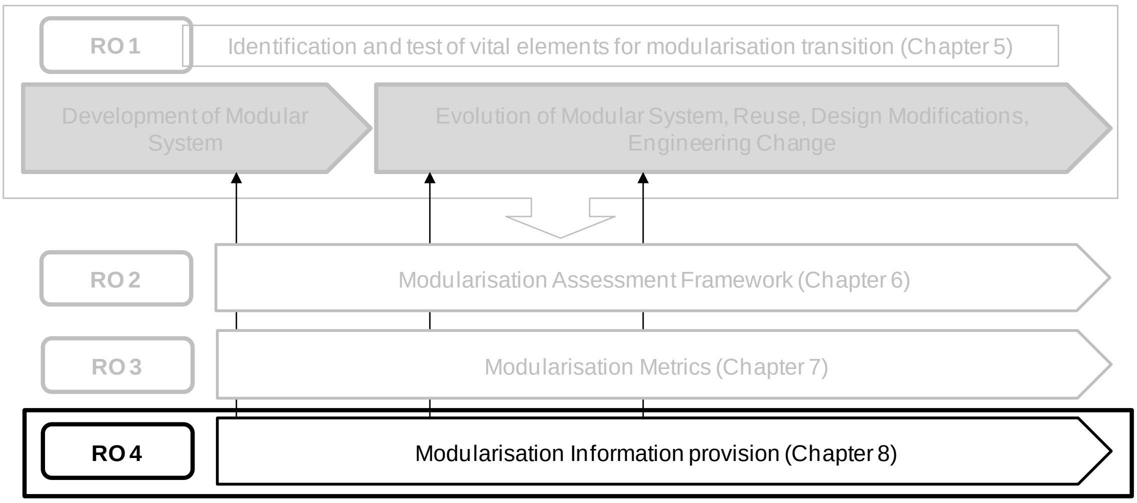


Figure 84: Relation of Chapter 8 to other chapters of this research work

8.1 Chapter overview

It is the purpose of this chapter to develop an approach for provision of modularisation information in companies. Therefore, it is first necessary to derive requirements for such an approach and to identify relevant information that has to be represented. Secondly, the approach has to be developed and tested in industry.

8.1.1 Research setting and methodology

The development of the approach for modularisation information provision is the third contribution of the Prescriptive Study of this research work. This is done by applying following steps:

1. Analysing the literature about state of the art about modularisation information provision
2. Collecting and analysing requirements of industrial practitioners
3. Iteratively developing the approach by moving from a concept and detailed design towards realisation, application and intervention in standard IT-systems of the company.
4. Validating the approach based on tests, application and expert interviews
5. All these steps were conducted in the primary case company.

8.1.2 State of the art in literature and industry

Previous research in the field of information provision for modular systems is scarce. However, first calls for research in the field also came from other researchers. Arnoscht (2011), LaLande (2013), Beckmann et al. (2014) and Gebhardt & Krause (2015) identified need for research in this field quite recently. Moreover, Karius (2011), Ponn (2015) and Kehl et al. (2015) have made the same calls from an industrial perspective.

All existing attempts have in common that they either have their origin in visual modelling approaches or in standalone software implementations. For instance, Stone et al. (2000) use functional flow structures, Ericsson & Erixon (1999) take use of Module Indication Matrices (MIM) and Eppinger & Browning (2012) use Design Structure Matrices. Harlou (2006), Mortensen et al. (2008) and Kvist (2010) have developed a Product Family Master Plan (PFMP), Pedersen (2010) uses CAD skeletons, Parslov and Mortensen focus on interface representation (2015), Gebhardt et al. (2014) represent a Module Interface Graph (MIG) and Bruun & Mortensen (2012) an Interface Diagram Formalism which are more sophisticated but still "manual" approaches for visualisation of product families. Software tools such as LOOMEO (Lindemann, Maurer and Braun, 2009), Complexity Manager (Schuh & Co., 2015), METUS (ID-Consult, 2015) or SOLEY (Kissel, 2014) are all proprietary standalone solutions to visualise and analyse complexity.

In the meantime, several researchers have made first attempts to come up with theoretical concepts or industrial case studies for integrating these concepts into standard IT-systems.

Kissel et al. (2012) present a framework for product structure management which contains data and information requirements. Kreimeyer (2012) suggests to store product architecture information within the PLM system of a truck manufacturer. His framework contains different layers, a product portfolio layer, a product architecture layer, a property layer, a configuration layer and an embodied solution space, in order to get a closed link from properties to embodied design of truck modules. Bruun et al. (2015) enhance the Interface Diagram in order to make the generic architecture of a product family explicit. This Interface Diagram has been imported into the PLM system of a construction vehicle manufacturer which helps to get a unified product structure for all vehicle variants of the product family.

8.1.3 Chapter elements

The chapter overview which is described in this Section 8.1 will be followed by Section 8.2 that describes requirements for the approach from industry. The core of this chapter are Sections 8.3 and 8.4 which deal with the description of the approach for modularisation information provision and the resulting information model within standard IT-systems of industrial companies. Section 8.5 describes the validation of the approach and the information model. Section 8.6 presents the specific discussion section of this chapter and Section 8.7 briefly summarises this chapter. Following figure illustrates the elements of this chapter:

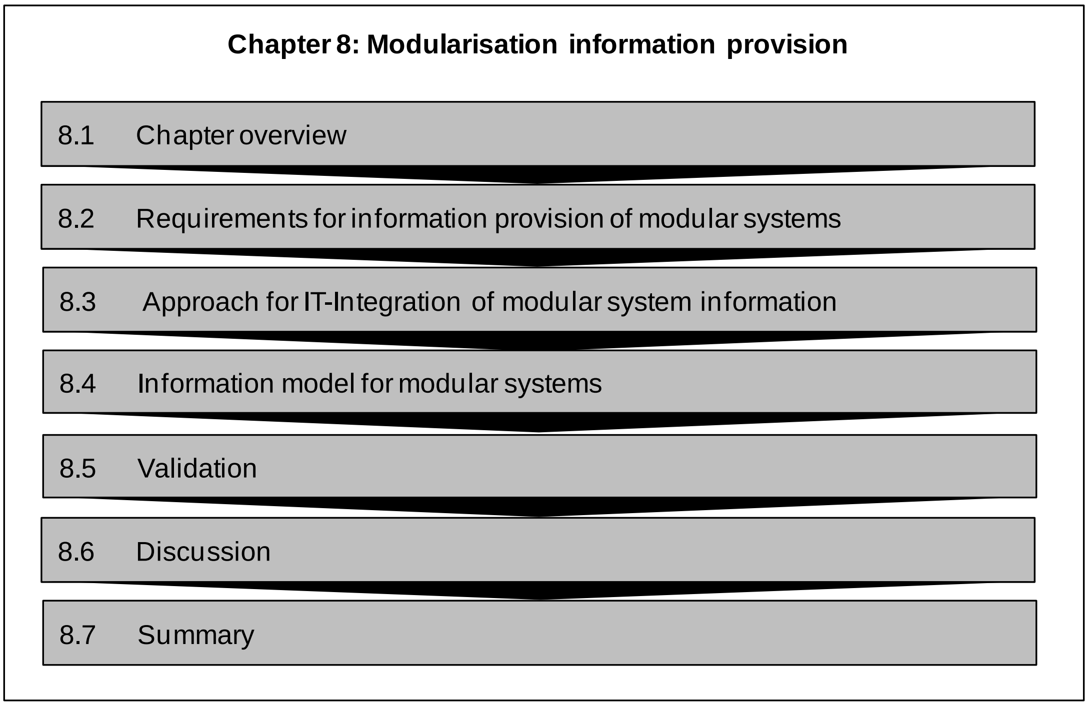


Figure 85: Elements of Chapter 8

8.2 Requirements for information provision of modular systems

Given the literature review in the previous section, it can be inferred that research in the field is still at the very beginning. Therefore, it has been seen as vital to first collect requirements in industry before developing support.

Requirements for the modularisation information provision have been collected from three different sources: literature, case studies in modularisation transitioning projects and semi-structured interviews.

The findings of this section and the following sections have been structured and refined based on the findings of Heilemann et al. (2014).

8.2.1 Requirements from literature

Kissel et al. (2012) find it as important to establish a generic architecture from which architectures for derivative products can be derived. Kreimeyer (2012) points out that it is important that the “product architecture is collision-free for the intended variant models”. Bruun et al. (2013) specified that modules and interfaces should be easily indentified, that standard and customised elements can be distinguished and that relational properties can be controlled within derivative projects.

8.2.2 Case studies in modularisation projects

The following requirements were collected during participatory field research in Case A and C within the primary case company:

- Information about the modular system must be available at the point of use. This means that module specifications have to be available in derivative local product and module development projects.
- Information has to be stored centrally and neutrally so that it can be reused by multiple local projects.
- Central items of the modular system are maintained by central organisational roles in charge of the modular system in order to provide best possible synergies across projects.
- Direct comparability between artefacts of local projects and specifications of the modular system.

8.2.3 Semi-structured interviews

Semi-structured interviews during a series of workshops and follow-up meetings took place in the primary case company over a prolonged period. The collected requirements were prioritised and checked for feasibility in an additional workshop with 15 engineers, IT-experts and engineering managers. The collected requirements are as follows (based on Heilemann et al. (2014)):

- Elements of the modular system like modules, module variants, interfaces and the modular system should be identifiable.
- Specifications of the modular system which are valid for multiple elements should be stored centrally and linked to respective items in derivative development projects.

- For reuse purposes, elements of the modular system should be easily searchable.
- Ownership and responsibility for modular system elements has to be indicated.
- Elements of the modular system should be linked so that their relationship gets traceable.
- Configuration management and data maintenance (e.g. identification, naming, and versioning) should be clear-cut, neutral and done centrally.
- Elements of the modular system undergo special treatment during the embedded engineering change process due to their higher impact. Thus, deviations from the architecture specification should be identified easily.

In addition, preconditions were identified that had to be considered during development of the concept:

- Low effort for data maintainers and design engineers
- Integration into existing standard core IT-systems, i.e. into integrated CAX chain (e.g. CAD, PLM, ERP)

8.2.4 Summary: requirements for IT-integration

The requirements that were established above have been processed and aligned to the goals of modularisation transition. Figure 86 shows a multi-layered cause-effect diagram that connects the goals of modularisation transition with solution-neutral requirements (derived from above), detailed solution requirements (derived from above) and the steps of the presented IT-Integration approach (presented in Sections 8.3 and 8.4). The figure shows that the approach that was derived from above mentioned requirements indeed contributes to the goals of modularisation transition under stable architectures during post-architecting phases.

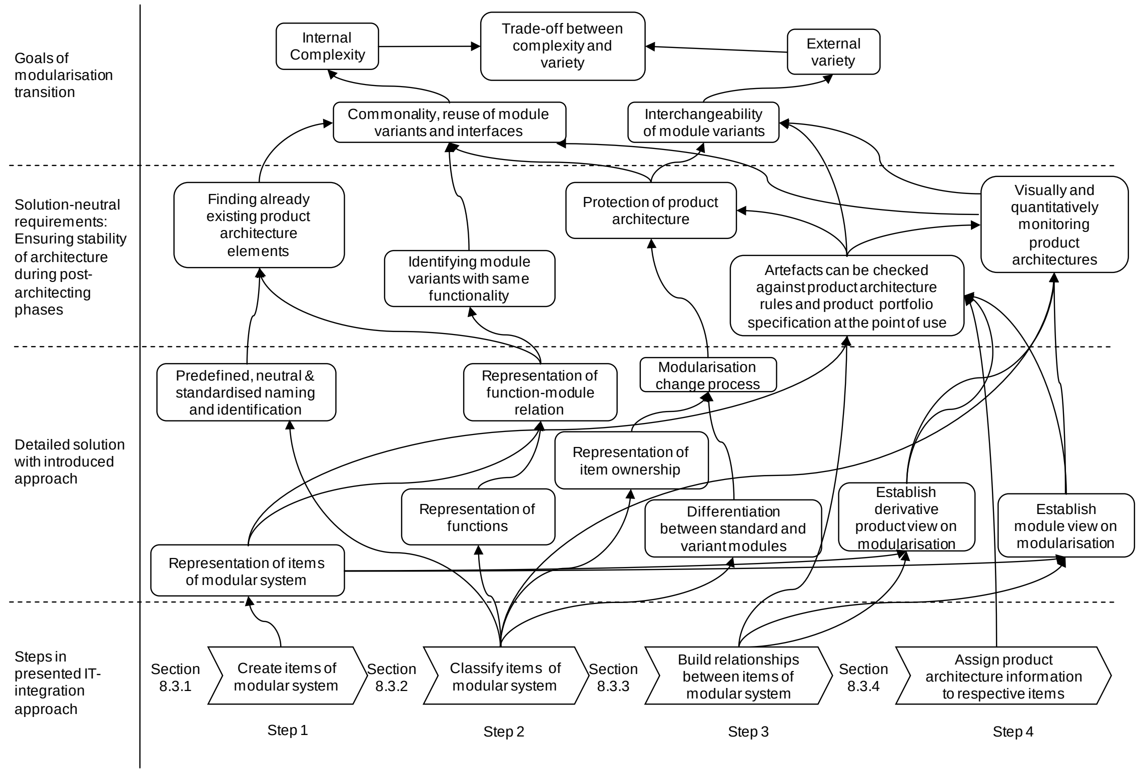


Figure 86: Cause-effect relationship between goals of modularisation transition, requirements and steps in presented IT-integration approach, Heilemann et al. (2014)

8.3 Approach for IT-integration of modular system information

It is the purpose of this section to look at the IT support issues in relation to making the transition from single product development towards the development of modular systems within existing products. Traditionally, such companies have been building products that consist of the items *parts* and *assemblies*. With modular system development, the research has shown that these companies have to make the shift such that their products consist of *module variants* in the first level and of assemblies or parts on the second level. In addition, derivative development projects have to make sure that their products satisfy the requirements of the derivative product specification and of the modular system specification which are passed on from higher level items. These higher level items are modules and the modular system itself (see Figure 87). In consequence, the IT-system which is used to store these items has to handle two perspectives: a derivative product perspective (see lower part of Figure 87) and a modular system perspective (see upper part of Figure 87).

Figure 87 is based on the proposed modular system development life cycle presented in Figure 61 of Section 6.4. It gets evident form Figure 87 that the focus of the IT-integration approach is on the later phases of modular system development and, thus, on the stability of the modular system.

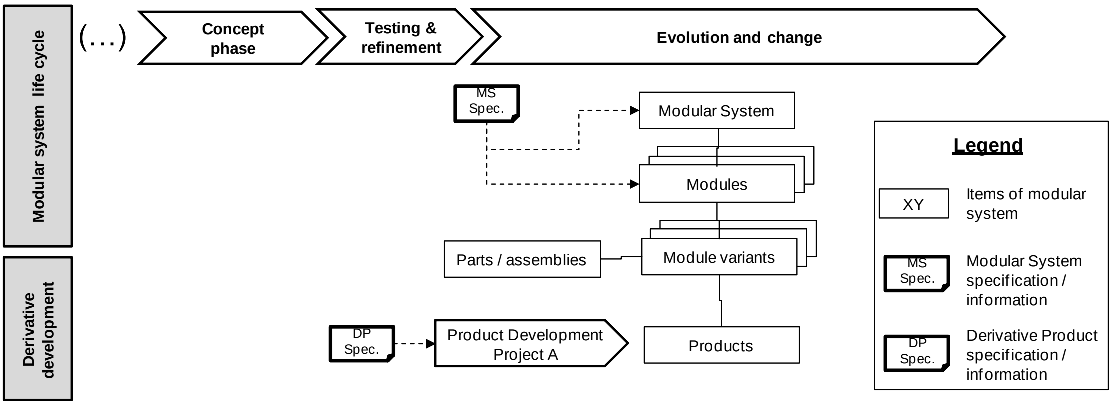


Figure 87: Modular system and derivative development developing items for products and the modular system to be stored in IT-systems.

As already indicated by Figure 86, the approach for providing information about modular systems in standard IT-systems is divided into four steps (Heilemann et al. (2014)). Firstly, items of the modular system are created (see Section 8.3.1). Secondly, the items of the modular system are classified (see Section 8.3.2). Thirdly, items of the modular system are linked (see Section 8.3.3). Finally, relevant modular system information is assigned to modular system items (see Section 8.3.4). These steps that have been developed in the research are explained in more detail below.

8.3.1 Step 1: Create items of modular system

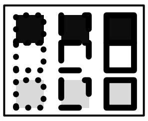
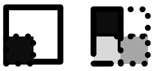
After the product architecture has been established and modular system specifications defined, engineers can start to represent the modular system within standard IT-systems. This means that information about the modular system from concept phase (e.g. in spreadsheet, document or graphical format) has to be transferred to CAD, PLM or ERP and represented within those systems with specific items. It depends on the specific characteristics of the company which IT-systems to use. However, it is strongly recommended to integrate the modular system into the whole CAX process chain on core IT-systems. For instance, the items of the modular system could be created in CAD, afterwards transferred to PLM and later automatically synchronised with ERP. For items without geometric information, it is also possible to create "plain" items in PLM or ERP and to attach non-geometrical information to these items later on.

The items that have to be created are depicted in Figure 90 of Section 8.4 and labelled with characters from A to H. Section 8.4 will give a detailed description of each item. In standard IT-systems, a BOM or a product structure consists of products, assemblies and components. Therefore, in step 1, *product items*, *component items* and *assembly items* have to be created like it is done with single product development.

The big difference with modular system development is now that products are created by combining different *module variant items*, which are made up of components and assemblies. In order to cluster associated module variant items, it is necessary to introduce *module items* for each module of the modular system. In order to reflect the derivative product-perspective of the modular system, a *modular system item with a specific focus on products* is introduced. The central counterpart to this item is the product-overarching module perspective. This is represented by a *modular system item with a specific focus on modules*. Finally, the last item to introduce is the *modular system item from an overall perspective* which represents the whole modular system construct in the IT-system.

In summary, the items proposed to be created within the IT-system with this approach are as listed in Table 24. These items have been established after analysing the data and information flow of a modular system development life cycle in the primary case company as it is described in Figure 87 and in Figure 61 of Section 6.4.

Table 24: Proposed items to be created in IT-systems

Items as illustrated in in Figure 61 of Section 6.4	**Proposed items to be created in IT-systems**
	Item: Modular System – Overall View (A), Item: Modular System – Product View (B), Item: Modular System – Module View (C)
	Item: Products (D)
	Item: Module Variants (E), Modules (F), Assemblies (G), Components (H)

The items are described in detail in Section 8.4. It is the purpose of these items to function as classification information carrier (see Section 8.3.2), as nodes for establishing a structure within the whole modular system (see Section 8.3.3) and as general information carrier like for product architecture "specifications" as illustrated in Figure 87 and in Figure 61 of Section 6.4 (see Section 8.3.4).

It has to be considered that the identifiers (e.g. part numbers or naming) for the items within the IT-system have to be neutral so that they can be reused across different development projects.

8.3.2 Step 2: Classify items of modular system (modularisation classification)

Once the items of the modular system from Step 1 have been created in the IT-system, they can be classified with modularisation-relevant attributes. This has two purposes:

1. These items have to be better protected due to their overarching modular system impact.
2. These items can be better searched for reuse purposes and are, thus, easily identifiable throughout the company.

In general, standard PLM and ERP systems can accommodate such classification functionalities. It depends on following factors which IT-system to use for modularisation classification purposes: a) standard classification system of the company and b) where the creation and change process for items is handled.

Figure 88 shows an example for a modularisation attribute list. This attribute list with the data structure and logic behind it has been prototyped in MS Access (see Figure 88) within the course of this research work. Moreover, the prototype was transferred as standard entry mask into Teamcenter PLM by IT-Experts of the primary case company. This modularisation mask has to be used mandatorily by all design engineers of the primary case company who create a new modularisation-relevant item in PLM.

In order to ensure that all classification attributes that are used by different derivative development projects follow the same logic and rules, the database behind the attribute list (i.e. classification options) is prefilled centrally by a role in charge of modular system data maintenance. Data that is used for classification has its origin in the modular system specification from concept phase (see Figure 61 or Figure 87) of the modular system development life cycle. Thus, all fields in the attribute list (see Figure 88) have a unique, company-wide agreed and standardised entry and are dependent on each other. This dependence amongst attributes, like configuration restrictions between characteristics, ensures that only those entries are feasible that are in line with specifications of the modular system that have their origin in a modular system development life cycle such as described in Section 6.4. Thus, only items that are compliant with designated plans of the modular system can be created.

- Classification for reuse purposes

For reuse purposes and to prevent creation of redundant items, the attributes “Item Name” and “Characteristic 1” to “Characteristic N” are used. Figure 88 shows an example for a predefined and dependent classification of a module variant. When the design engineer classifies the item, its corresponding data is already given centrally from data that were initially created after the concept phase. In the example of Figure 88, the design engineer chooses the name for his module variant “Servo_Pneumatic_Positioning_Module”. In order to uniquely identify his module variant, the design engineer assigns specific functionality or other characteristics to this module variant. In the example of Figure 88, he had the option to assign specific values for stroke length, output signal and piston diameter to his module variant. Thus, all items of this modular system follow the same naming and classification scheme. With this transparency, they can be easily found and reused. Moreover, it is not possible to generate duplicated items or items that are not predefined by the modular system specification.

- Classification for protection purposes

While change requests are more likely to be raised and the impact of changing a modular system item is higher than for “single product items”, these items have to undergo careful treatment in order to protect the stability of the modular system. Thus, two characteristics are assigned to respective items of the modular system: the “Modularisation Item Owner” and the “Variance Classification” (see Figure 88). The item owner is the person in charge for this item. In case of a new engineering change request or any discrepancies, the item owner has to decide how to handle the situation because he is the person with appropriate overview and expertise about the impact on the modular reference architecture. For example, if a design engineer wants to create a module variant item which does not adhere to its specifications, such a violation will be easily detectable by the modularisation item owner during the engineering change process. Thus, the owner can reject the request with the demand to rework the module variant. In addition, the “Variance Classification” attribute is used to control the change process within the IT-system. For instance, if a module variant with “High Impact” classification is requested to be changed, the change process will be much more strictly than for a “Low Impact” module.

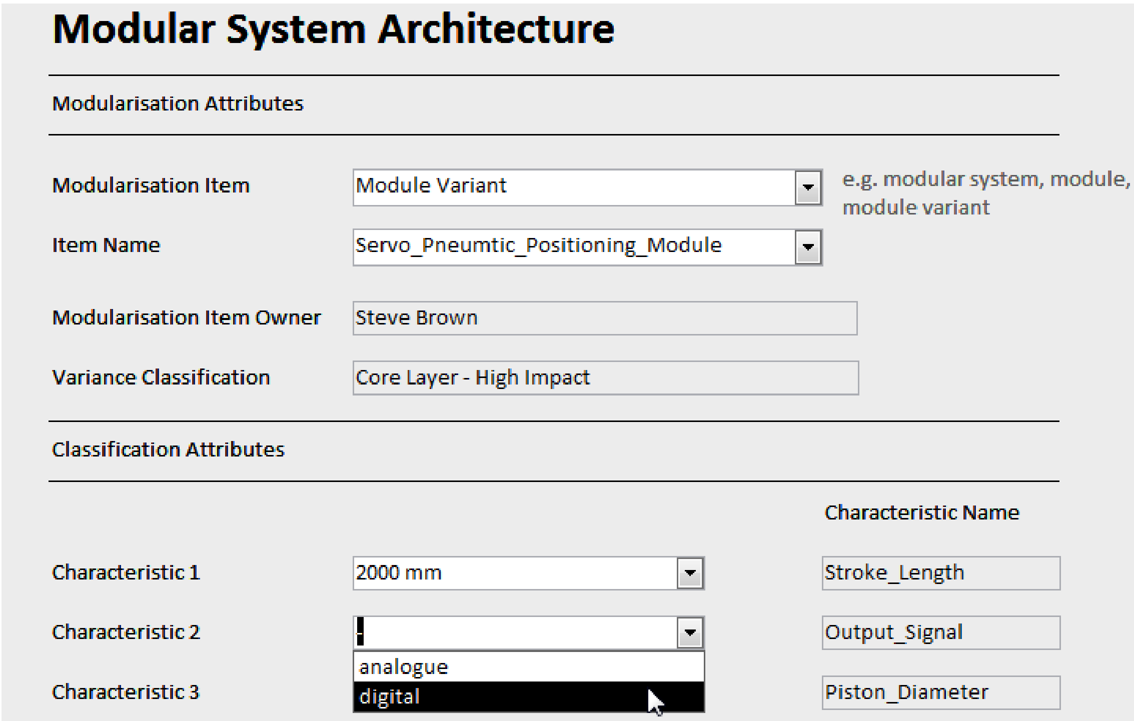


Figure 88: Example for modularisation attribute list for standardised classification of modularisation items in a PLM system

8.3.3 Step 3: Build relationships between items of modular system

After creation and classification of modularisation items in Step 1 and Step 2, the items can be set into relation by linking them. With single product development, there are usually only links between product items, assembly items and part items in the structure of standard IT-systems. With modular system development, it is required to show the relationship and hierarchy of all dependent items of Step 1. This has a few practical implications (Heilemann et al. (2014)). First, it shows the relation of “slave” items to “master” items. Second, it can be ensured that all “slave” items follow architectural specifications of their “master” items. For instance, all module variants have to comply on the one hand with the specification of their master-product but also with the architectural specification of their master-module. All module items have to comply with the architectural specification of their master item, the overall modular system item, and so forth. Table 25 illustrates the direct relations between all slave and master items of the modular system. It gets evident that all slave items have to adhere to the architecture specification of their master item.

Table 25: Concept of linking slave items to master items to obtain architectural adherence

Slave Items	... has to adhere to architecture spec. of ...		**Master Items**
Item: Modular System - Module View	→	Spec.	Item: Modular System - Overall View
Item: Modular System - Product View	→	Spec.	Item: Modular System - Overall View
Item: Module	→	Spec.	Item: Modular System - Module View
Item: Product	→	Spec.	Item: Modular System - Product View
Item: Module Variant	→	Spec.	Item: Module
	→	Spec.	Item: Product
Item: Component, Item: Assembly	→	Spec.	Item: Module Variant

Figure 89 shows a class diagram that includes the items of the modular system with modularisation classification information and relationships between the items summarising Step 1 to Step 3. Moreover, it shows the chain of relations between all items of the modular system. Therefore, it gets evident that there is an indirect link between modular variants and the overall modular system. Hence, it is ensured that all design operations on the "lowest" level (i.e. module variant, component or assembly level) can be controlled so that they contribute to the targeted overall modular system from a product point of view and from a module point of view.

In addition, Figure 89 illustrates the "traditional" items and links (in grey colour) with single product development and the new items and links (in white colour) that have to be established with modular system development. By looking at the new items and links at the upper left part of Figure 89, it becomes clear that the new approach creates an additional view on the generic, product-overarching module perspective of the modular system.

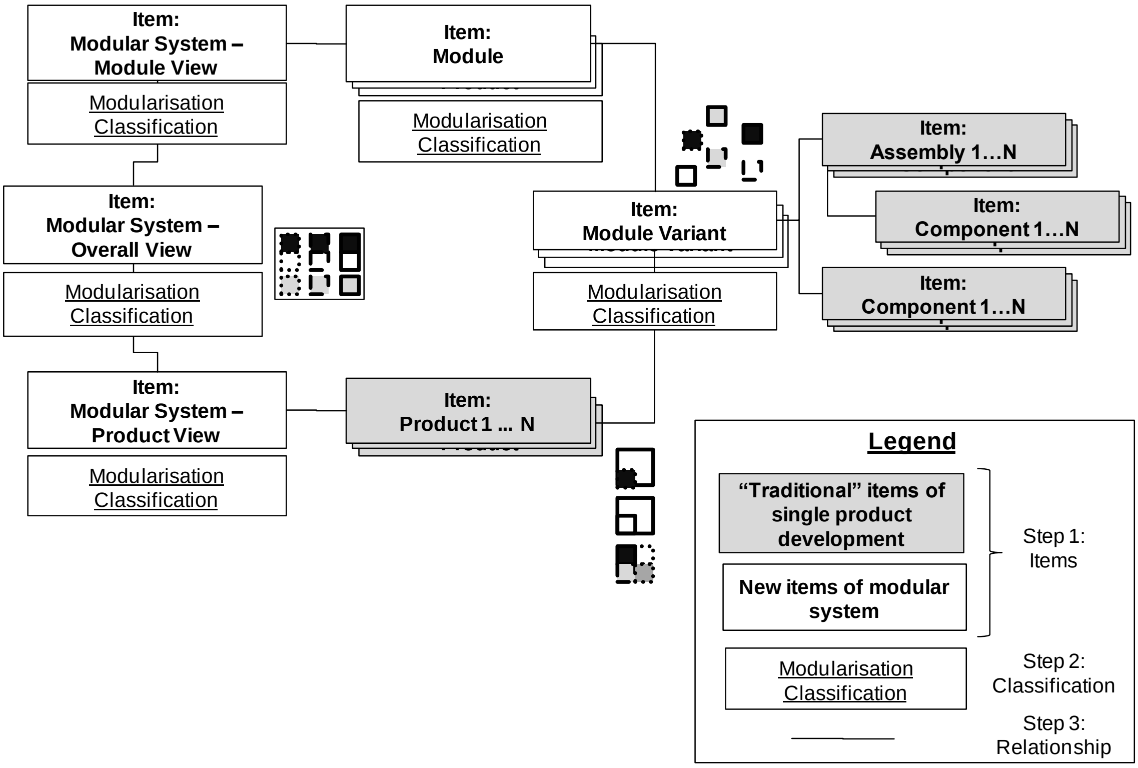


Figure 89: Class diagram showing the items of a modular system with modularisation classification information and relationships between the items summarising Step 1 to Step 3

8.3.4 Step 4: Assign relevant information to respective items

Step 4, the last step of this approach deals with the assignment of relevant information (e.g. CAD templates, drawings, documents, plans) to modularisation items stored within the IT-system. A large amount of information and numerous documents are created during the requirements and concept phase, but also before and after these phases in the modular system life cycle (see Figure 61). Among these documents are binding plans and specifications for the modular system. This kind of information has to be maintained throughout the whole modular system life cycle. Moreover, the information has to be available centrally and updated simultaneously within different derivative development projects. For instance, the module specification has to be maintained centrally because it is valid for all module variants of the module and, thus, it is valid for all derivative product development projects developing module variants of the module. Therefore, the module specification is assigned to the module item that is linked to all module variant items which are developed or used in derivative development projects. This link supports design engineers in derivative projects to directly check their compliance with modular system specifications. Moreover, owners of a "master" module items or "master" modular system items can control their "slave" items on adherence to overarching architectural specifications.

As a rule it can be said that information which is valid for a number of "slave" items (e.g. module variant item) has to be stored with the master of the slave item (e.g. module item). Hence, all module variants have to adhere to the same interfaces and space requirements stored with the module item in order to ensure interchangeability of module variants. This central storage of “master” information has the advantage that information is free of duplicates, automatically connected and traceable to slave items. Figure 90 shows that each item of the modular system contains a set of additional modular system data and documents.

A more detailed description about what information to attach to each item of the modular system is given within the next section.

8.4 Information model for modular systems

The result of processing Step 1 to Step 4 of the previous section can be thought of as an information model for the modular system which completely represents and describes the modular system in a standard IT-system. It is described in more detail below.

8.4.1 Detailed description of the modular system information model

The modular system information model as a result of the IT-integration approach of Section 8.3 consists of the following elements: modularisation items, modularisation classification, relationships between the items and assigned modular system information and documents. A detailed description of the elements for each modularisation item will be described in the remaining part of this section. Figure 90 gives an overview of the modular system information model in a UML-based class diagram. Each element of the model is labelled for better referencability within the text. Each item is labelled with characters from (A) to (H) and each relation is labelled with a number from 1 to 9. The different relations are further described in Figure 90 using UML notation. For instance, a module variant belongs to exactly one module (1..1) whereas a module comprises one to an unlimited number of module variants (1..*).

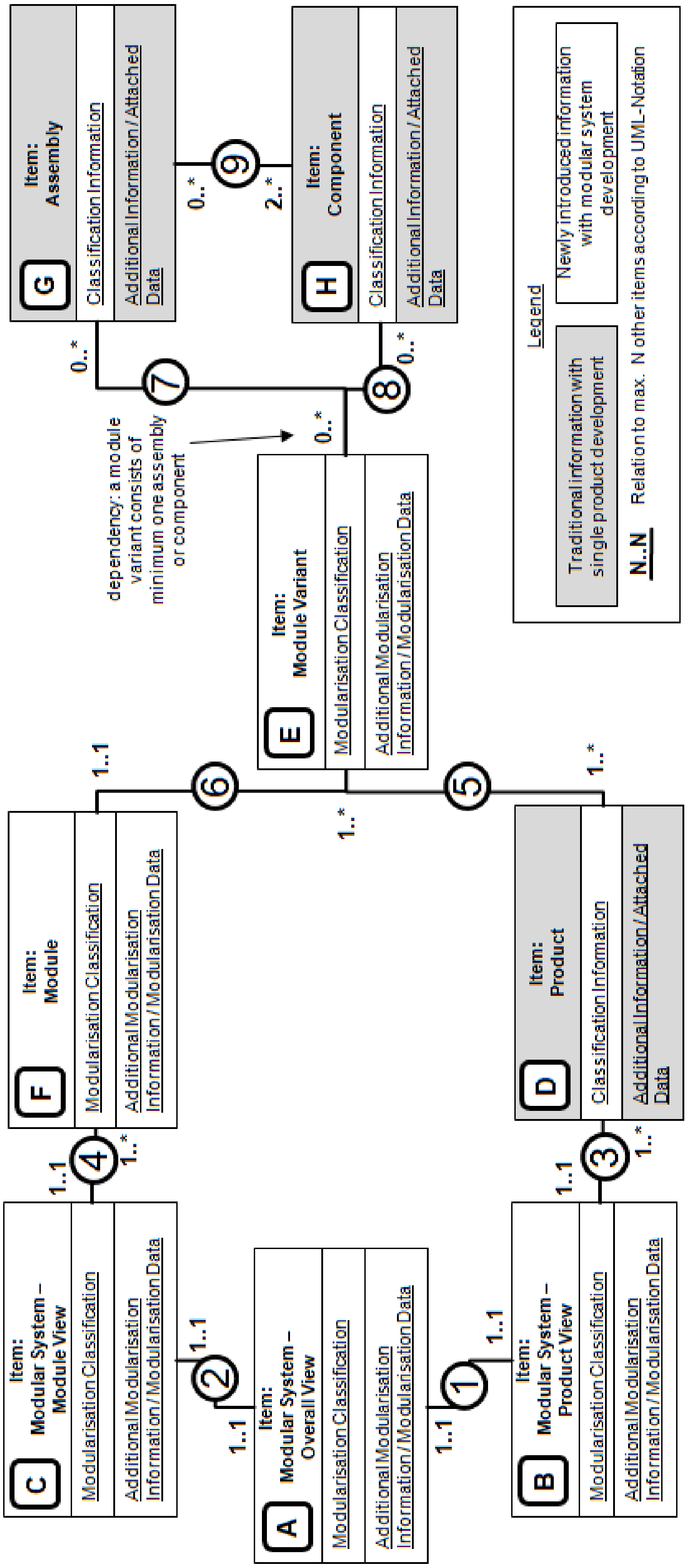


Figure 90: Information model for IT-integration of modular system development (Heilemann et al. (2014))

- Item: Modular System – Overall View (A):
  - Purpose:
    It is the purpose of this item to serve as representative for all matters that impact the overall modular system. This item relates to the derivative product perspective and to the common module perspective of the modular system. Therefore, this item bridges the gap between external variety to be generated within product development projects and internal complexity that is represented by the internal module perspective.
  - Modularisation classification:
    Modularisation-specific attributes that are used to classify this item are as follows. The modularisation item has to be marked as "Modular System – Overall View", the item name is classified with the unified name of the modular system and the modularisation item owner has to be assigned to the item. The modularisation item owner is the role in charge for the overall modular system. For instance, he manages the trade-off between commonality and external variety and ensures that the modular system specifications from an overall perspective are met.
  - Relational information:
    This item is linked to the item "Modular System – Module View" (relation 2) and to the Modular System – Product View" (relation 1). The UML notation shows that a modular system from the overall perspective has exactly one derivative product perspective and one internal module perspective.
  - Attached Modularisation Information / Data:
    It is recommended that following specifications of the modular system are attached to this item:
    - The **modular system requirements specification** with architectural-relevant requirements should be assigned to this item due to its validity for all other items. The specification can later be broken down into module and product level.
    - The plan that shows how derivative products to be developed are equipped with module variants from the modular system (**module-product matrix**). This plan can later be compared to the actual links between module variants and derivative products in the standard IT-System. Moreover, the plan shows when a certain module variant has to be ready for usage in a derivative product.
    - A **Feature Tree** might show variant-driving features that the modular system has to fulfil. These features can be broken down onto product level, module level and later be compared to actual functional attributes of modularisation classification (see Step 2, Figure 88). Moreover, a **Variant Tree** might indicate

where variance is created during the production flow of all products and modules of the modular system (e.g. to postpone the point of variance creation).

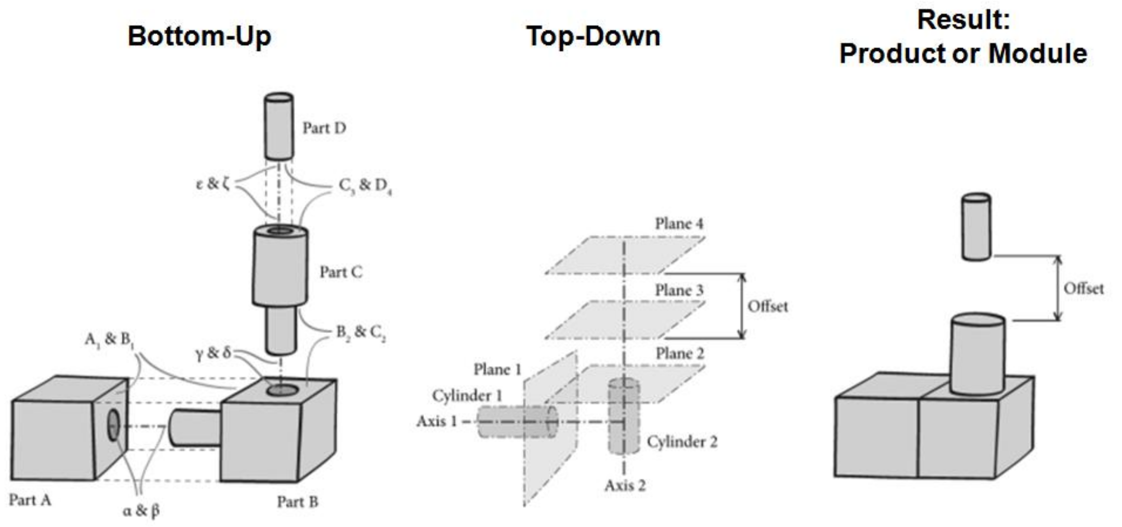


Figure 91: Top-Down approach where generic interfaces and spatial requirements are used to guide the creation of parts in order to derive products or modules (based on Pedersen (2010, p. 241))

- "**Architecture template**" which defines positions, interfaces and arrangement of modules within products to be derived. The product template could be in CAD format implementing all necessary information for combinability of module variants. This kind of information can be further broken down into lower level specifications like module specifications. The introduction of an architecture template follows the principles of Pedersen's (2010, p. 241) **Top-Down Design** approach. In this approach, the structure is designed before elements. Thus, interfaces, spatial relations and eventually key features are designed before "the physical embodiment of all parts is known". This is the opposite to Bottom-Up Design where elements guide the structure (see Figure 91).

- Item: Modular System – Product View (B):
  - Purpose:
    It is the purpose of this item to centrally store all information that is valid for the whole product portfolio to be derived from the modular system. Thus, this item reflects that the modular system is not only built for single products but for a whole range of parallel and future products which all draw upon the common modular reference architecture. Moreover, it stands for external variety that is generated from the modular system.
  - Modularisation classification:
    Several modularisation-specific attributes that are used to classify this item have to be used. The modularisation item has to be categorised as "Modular System –

Product View", the item name is classified with the unified name of the modular system and the modularisation item owner from a derivative product perspective has to be assigned to the item. The modularisation item owner from a product view is the role in charge for all products to be derived from the modular system. For instance, he ensures that derivative product specifications are met and that the required external variety is generated.

- Relational information:
  The item is linked to the item "Modular System – Overall View" to which it has exactly one link according to the UML notation (relation 1). Moreover, this item is linked to all actual product items that are derived from the modular system (relation 3). All products of the modular system have exactly one relation to the modular system – product view item while one to an unlimited number of products can be assigned to this item.

- Attached Modularisation Information / Data:
  All modularisation-relevant information or documents which are valid for all derivative products have to be attached to this item. It is recommended to attach at least following information.

  - A **product portfolio roadmap** which shows the timeline for the development life cycle of derivative products. This roadmap can be checked against actual development of products available in the IT-system.
  - **Requirements and feature specification** for the products to be implemented which can be compared to actual implementation of features in the IT-system which is classified by the modularisation attribute list.
  - **Generic product structure** which has to be used by all products to be derived from the modular system.

- Item: Modular System – Module View (C):
  - Purpose:
    It is the purpose of this item to centrally store all information that is valid for all modules derived from the modular system. This item represents the goal of the modular system to build modules not for single products but for a wide portfolio of the generic modular system. Thus, the module view stands for the internal complexity which shall be reduced with modularisation transition.
  - Modularisation classification:
    Several modularisation-specific attributes that are used to classify this item have to be used. The modularisation item has to be categorised as "Modular System – Module View", the item name is classified with the unified name of the modular system and the modularisation item owner from a module perspective has to be assigned to the item. The modularisation item owner from a product view is the role in charge for all modules to be derived from the modular system. For instance,

he ensures that the reuse and commonality targets are met by each module owner and he is involved in high impact changes to the modular system.

- Relational information:
  The item is linked to the item “Modular System – Overall View” to which it has exactly one link according to the UML notation (relation 2). Moreover, this item is linked to all actual modules (relation 4). Module items have exactly one relation to the modular system – module view item while one to an unlimited number of modules can be assigned to this item.

- Attached Modularisation Information / Data:
  All modularisation-relevant information or documents which are valid for all modules is stored with this item. For instance, these documents are as follows and can later be compared to actual data that is available in the IT-system.

  - A **module roadmap** with the timeline of the development life cycle for each module variant.

  - A description of module variants and their features with the maximum number of variants per module (**module-variant matrix**).

  - **Interface matrix** which specifies interfaces between different modules.

- Item: Product (D):

The product item is handled almost in the same way like with traditional single product development. However, the difference is that unlike single product development where a product consists of assemblies and components, the product consists of module variant items. This is illustrated by relation 5. Moreover, the attribute list for modularisation classification could be used for classification of product functionalities. All product items to be derived from the modular system have to follow the specifications (e.g. generic product structure) given by their master item “Item: Modular System – Product View (B)”.

- Item: Module Variant (E):

  - Purpose:
    It is the purpose of the module variant item to realise the generic modular system and to enable derivation of local products by combination. Thus, the modular system is realised with the accomplishment of module variants which are in line with architecture specifications given by their master item “Item: Module (F)”.

  - Modularisation classification:
    This item is classified with the attributes for modularisation classification “Modularisation Item”, standardised “Item Name”, the modularisation item owner, variance classification attributes and predefined classification characteristics with their value (see Step 2 of Section 8.3.2 and Figure 88).

  - Relational information:
    Each module variant consists of assembly or component items which is illustrated

by relations 7 and 8. A module variant has to be built for the generic modular system for reuse purposes and for concrete derivative products. Thus, the module variant item is set into relation with the module item (relation 6) which stores master information for architectural conformity (e.g. interfaces, module boundaries, requirements) to which all module variants have to adhere to. This adherence is a vital part for the stability of the modular system. On the other hand, the module variant is linked to each product variant where the module variant is actually used (relation 5). This relation is a vital part for the actual external variety to be generated from the modular system.

- Attached Modularisation Information / Data:
  Information like drawings or models of the module variant. For instance, if information is stored in CAD format, it is possible to make automated collision analysis with the module item if interfaces and space boundaries of the module item are stored in CAD format as well.

- Item: Module (F):
  - Purpose:
    Interchangeable and reusable module variants have to adhere to the same architectural specifications. Thus, it is the purpose of the abstract module item to store binding information that connected module variant items have to fulfil. For instance, the module item “Cover” specifies interfaces and architecture relevant requirements for the module variant items “Cover_45mm” and “Cover_85mm”.
  - Modularisation classification:
    This item is classified with the attributes for modularisation classification “Modularisation Item”, standardised “Item Name”, the modularisation item owner, variance classification attributes and predefined classification characteristics. The modularisation item owner is usually a module owner who is in charge for the master module and all corresponding module variants. For instance, if a module variant shall be newly created or if an existing module variant shall be changed, the module owner has to approve the request based on adherence to module specifications. The attribute “Variance Classification” controls the engineering change process for respective module variants. If a high impact module variant shall be changed, the change process should be much more restricted than the change process for a low impact module.
  - Relational information:
    The module item is linked to its master item, the modular system – module view which bundles all modules of the modular system (relation 4). Moreover, all modules are connected to the respective module variants which have to fulfil their master’s specifications (relation 6).
  - Attached Modularisation Information / Data:
    In order to ensure adherence to architectural rules of the modular system and to

compare actual realisation of module variants to their plan, following information is stored with the module item:

- **Interface specifications** and drawings to which all module variants have to adhere to (e.g. textual or in 2D or 3D CAD format).
- Specification of **geometric or spatial requirements** which module variant design has to consider (e.g. textual or in 2D or 3D CAD format).
- **Module specifications** which textually describe the module and its architectural requirements.
- Geometrical, spatial and interface requirements can be used to generate a so-called **module skeleton** which is used in Pedersen's Top-Down Design (2010, p. 243). A module skeleton is a generic structure in which interfaces, spatial relations, geometry or key characteristics are controlled. The skeleton can be used as an architectural specification repository which can be linked to a number of different module variants. Figure 92 shows the principle of module skeletons.
- Information about tolerances and requirements for reliability, safety and failure

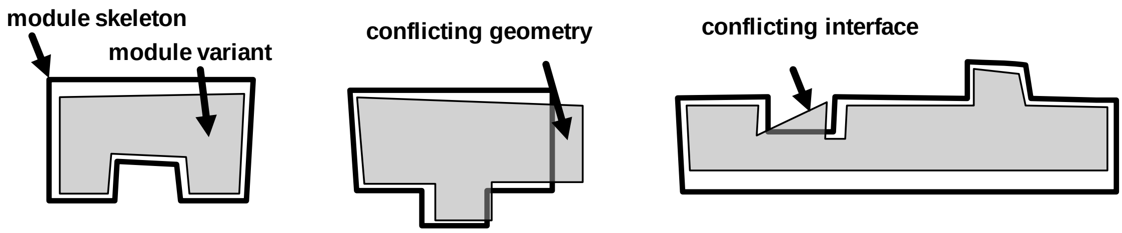


Figure 92: Principle of a module skeleton and its controlling factor for module variants

- Item Assembly (G) and Item Component (H):

Besides the fact that these items are organised into module variants, they are not handled differently as with traditional single product development (relations 7, 8, 9). It is possible to use the attribute list for Modularisation Classification for any ordinary classification purposes of parts and components.

Figure 93 shows an example for a concrete information model of a modular cylinder system. It also shows traditional information from single product development and additional information from modular system development. From a module point of view, the cylinder system consists of the modules cylinder body, valve unit, rod fastening and mounting. These modules are embodied by linked physical module variants on the lower level. For instance, the figure shows examples of different mounting module variants (MV) which all have to be in compliance with the illustrated space requirements and interfaces of the master mounting module. Thus, these modules are compliant with common modular system specifications. Moreover, Figure 93 shows how different compliant module variants can be combined to build different cylinder products (e.g. "Cylinder 1 (Hydraulic)" or "Cylinder 2 (Pneumatic)"). This branch of Figure 93 is the "Product View" on the overall modular system.

Figure 93 illustrates in the style of Figure 90, based on the modular cylinder system, how information about each level of the modular system can be efficiently retrieved, either by using classification information (see Step 2, Section 8.3.2), by using the links which have been established between the items (see Step 3, Section 8.3.3) or by consulting additional data or documents which are directly attached to an item (see Step 4, Section 8.3.4). Each distinct module item, product item and module variant item comprises further classification information (see Step 2, Section 8.3.2) and "additional information" (see Step 4, Section 8.3.4) like module skeletons or interface documents. Therefore, it gets easily evident if there are items which are not in line with reference modular system specifications. For instance, this can be found out by looking at the Module Item (F). The figure shows under "additional information" a module skeleton for the Module Mounting and further space requirements and interface information. It is obvious that there is a direct link from the Module Mounting to its module variants MV Mounting 1 to n. The "additional information" section of these module variants MV contains the geometrical information for each MV Mounting. The direct link of these items to their master module item allows to directly evaluating whether the geometries of each MV mounting are compliant with their master's geometrical module skeleton.

The next section will provide brief insights from the application of the modular system information model in practice.

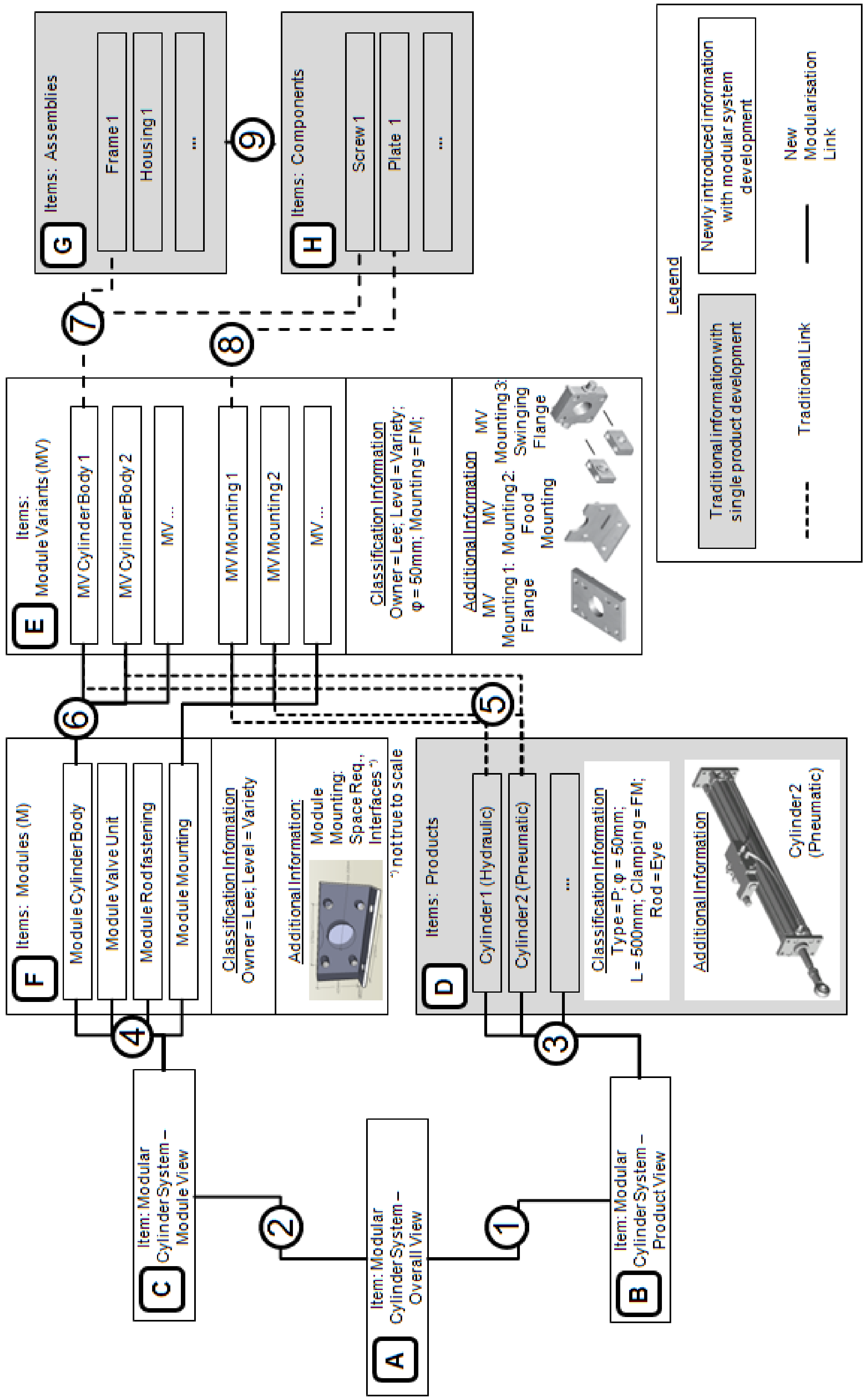


Figure 93: Example for an modular system information model for a modular cylinder system (Heilemann et al. (2014))

8.4.2 Examples from practice

The described modular system information model (see Section 8.4.2) has been introduced and established based on Teamcenter PLM as leading IT-system in the primary case company in order to fulfil the requirements of Section 8.2. The IT-integration approach (see Section 8.3) and the modularisation information model (see Section 8.4.2) are still in practical operation there. For confidentiality reasons, these operationally-driven examples could not be used for the purpose of this thesis.

From a research-driven point of view, the IT-integration approach (see Section 8.3) was also applied on a modular cylinder system in another PDM environment. Figure 94, Figure 95 and Figure 96 show an exemplified and disguised modular system information model (see Section 8.4.2) of a cylinder system based on the standard PDM System PTC Windchill PDMLink. It is the purpose of these figures to show how the modularisation information model, which was illustrated in the form of theoretical class diagrams in Figure 90 and Figure 93, has been realised in a practical PDM system.

Figure 94 shows how the item of the Cylinder Body Module of Figure 93 has been created in PDM according to Section 8.3.1. Moreover, the lower part of Figure 94 shows the attribute section where modularisation classification information according to Section 8.3.2 can be entered. This example shows a case where the module was created as a CAD item which contains geometrical specifications for respective module variants (as some kind of module skeleton).

Figure 95 indicates how the structure of the modular system is created according to Section 8.3.3. The figure shows the example of the cylinder body module of Figure 93. The module item has been used to link seven different concrete module variants and their lower structures to the same overarching module.

Figure 96 presents examples of architectural documents which specify modular system relevant parameters of the cylinder body module and its linked module variants. The left part of the figure shows how architectural documents are ordered so that they can be easily retrieved and attached to relevant items. The right part of the figure shows the following documents: a module-relevant requirements specification, the module roadmap, variant drivers of the module, a detailed module specification and various modular system relevant interface specifications. These documents can either be used as own "controlled" items and linked or as "uncontrolled" attachments (see "attachment" function within Figure 94) for the central module item. In sum, Figure 96 shows how the activities of Section 8.3.4 have been implemented into practice.

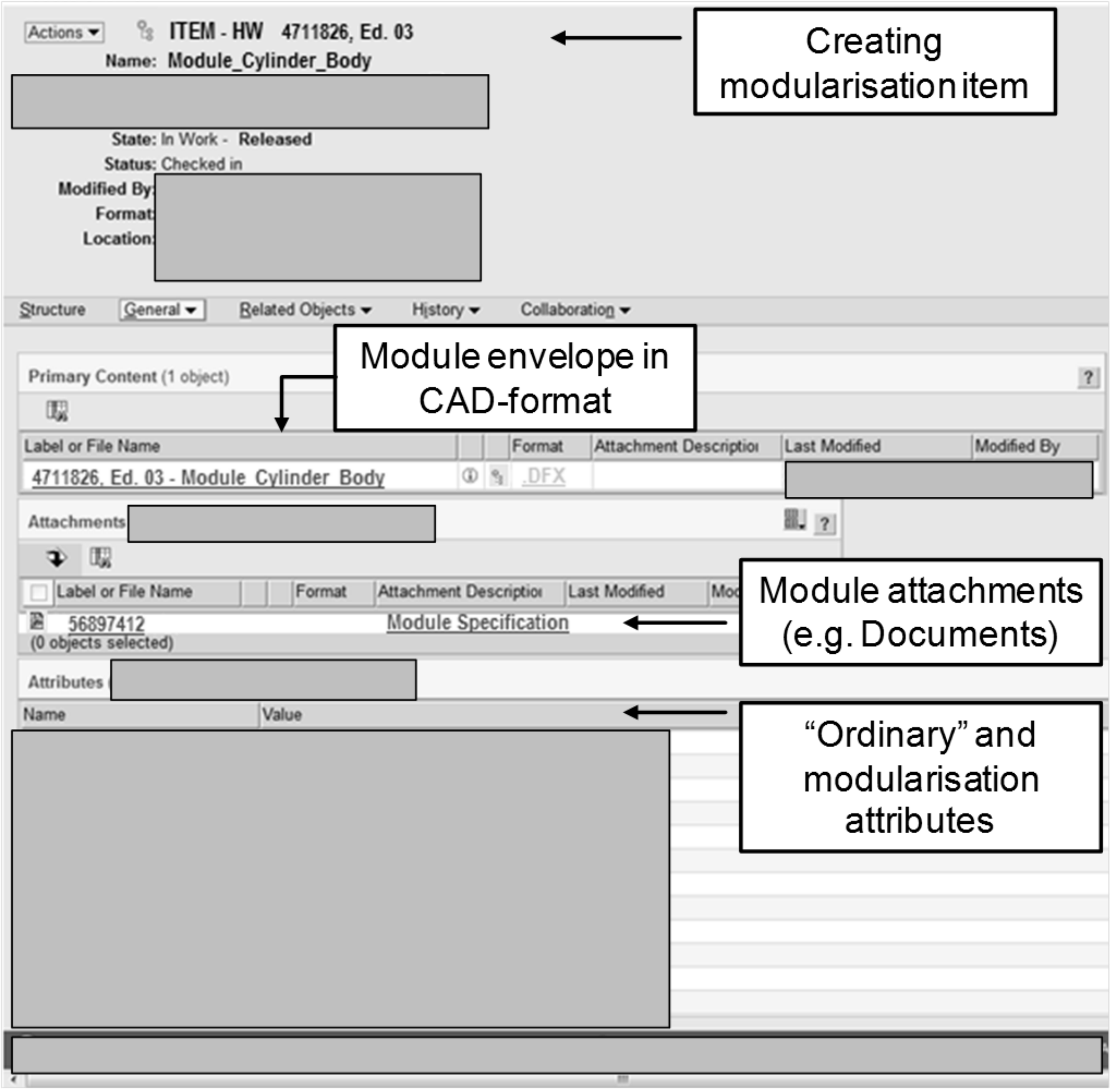


Figure 94: Example for creating and classifying a cylinder body module within PTC Windchill PDMLink

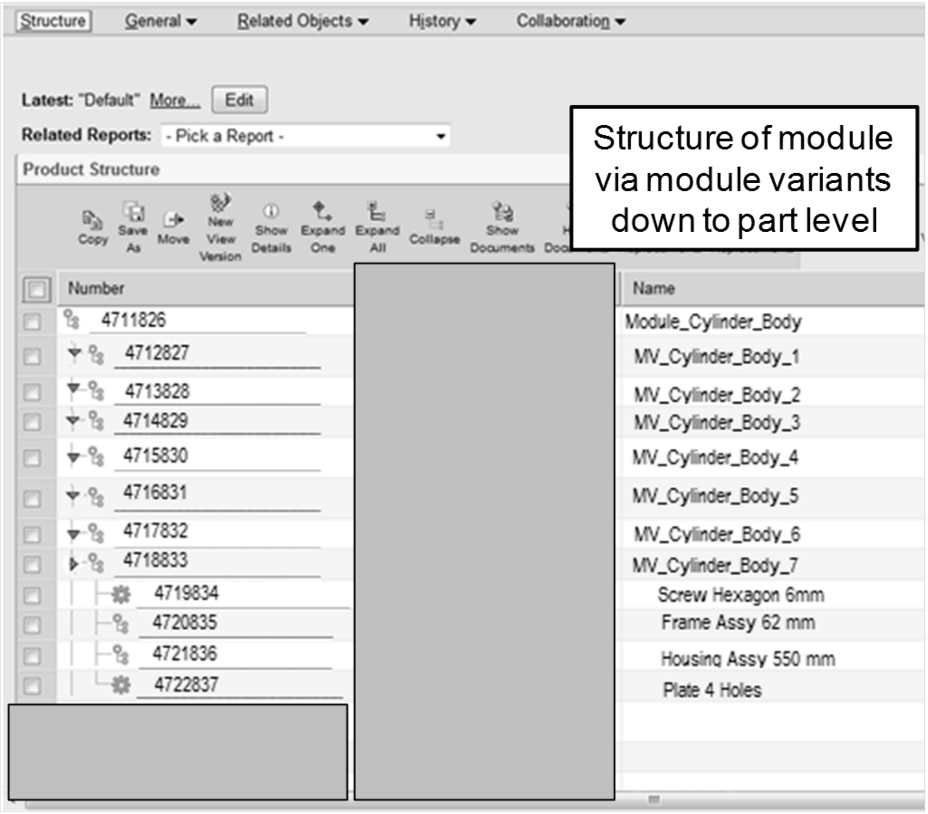


Figure 95: Linking items of the modular system exemplified at the case of a cylinder body module in PTC Windchill PDMLink

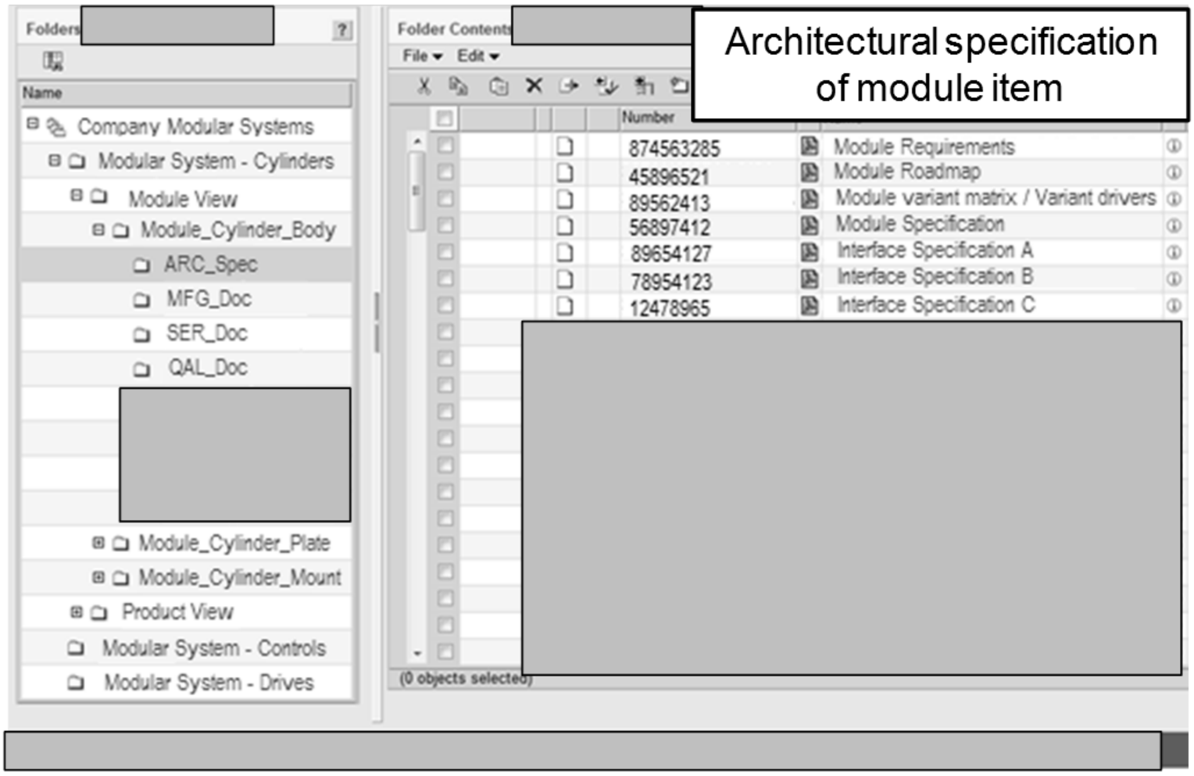


Figure 96: Assigning modularisation information (e.g. documents) to an overarching module in PTC Windchill PDMLink

8.5 Validation

The approach for modularisation information provision has been validated in two different ways: *Validation against the requirements* of Section 8.2.4 and validation of *applicability and relevance* of the approach.

8.5.1 Validation against requirements

The approach has been validated against the requirements given in Figure 86 and Section 8.2.4 respectively. Figure 97 shows solution-neutral requirements which shall ensure stability of the modular reference architecture during later stages of the modular system life cycle. It has been shown that these requirements are fulfilled by the approach presented within this chapter. It is suggested that this will help to achieve the goals of modularisation transition which are an improved trade-off between internal complexity and external variety (see Figure 86).

The approach and the resulting modular system information model have been validated in Teamcenter PLM in an industrial setting to be capable to fulfil following requirements:

- Finding already existing product architecture elements
- Identifying modular variants with same functionality
- Protection of the product architecture
- Artefacts can be checked against product architecture rules and product portfolio specification at the point of use
- Visually and quantitatively monitoring product architectures

In sum, the validity of the cause-effect relationship presented in Figure 86 has been proven to be valid, except concrete effects on high-level goals of modularisation transition which are expected to be harvested not before one or more development life cycles.

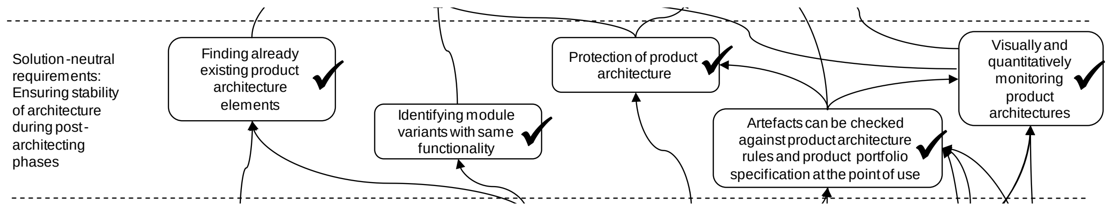


Figure 97: Fulfilment of solution-neutral requirements (excerpt of Figure 86) by the approach for modularisation information provision

8.5.2 Application evaluation

The approach developed for the provision of modularisation information has been evaluated in industry in order to find out whether the approach is usable and relevant. The PLM system used for this evaluation was Teamcenter PLM and the iterative evaluation cycle took place in the primary case company.

Before the approach was implemented, it was evaluated several times during its development as listed below.

- Two workshops were organised with engineering designers and IT-experts in order to find out the relevancy of IT-support for modularisation.
- The support was further developed and iteratively evaluated on relevancy and usability during a series of ten follow-up meetings with engineering designers and PLM key users.
- The support was tested on its usability in the PLM test system before it was transferred to the PLM productive system of the primary case company.

After these evaluations, the approach was implemented into the company's PLM system, it was presented and discussed with engineering experts and PLM key users of a development site. Focus of the evaluation was on usability and relevancy. Afterwards, usability and relevancy of the approach were again tested by applying it to a pilot modular system of the primary case company C. Finally, the approach was presented to engineering designers and engineering managers of the primary case company in order to prepare it for company-wide implementation. In terms of research, it was the main purpose of these meetings to evaluate usability and relevancy of the developed IT-integration support.

Transfer of the approach to an ERP system and the actual implementation across the company has not been covered by this research work. However, it can be assumed that such continuing efforts of the primary case company of the approach indicate strong usability and relevancy of the approach for modularisation information provision presented within this chapter.

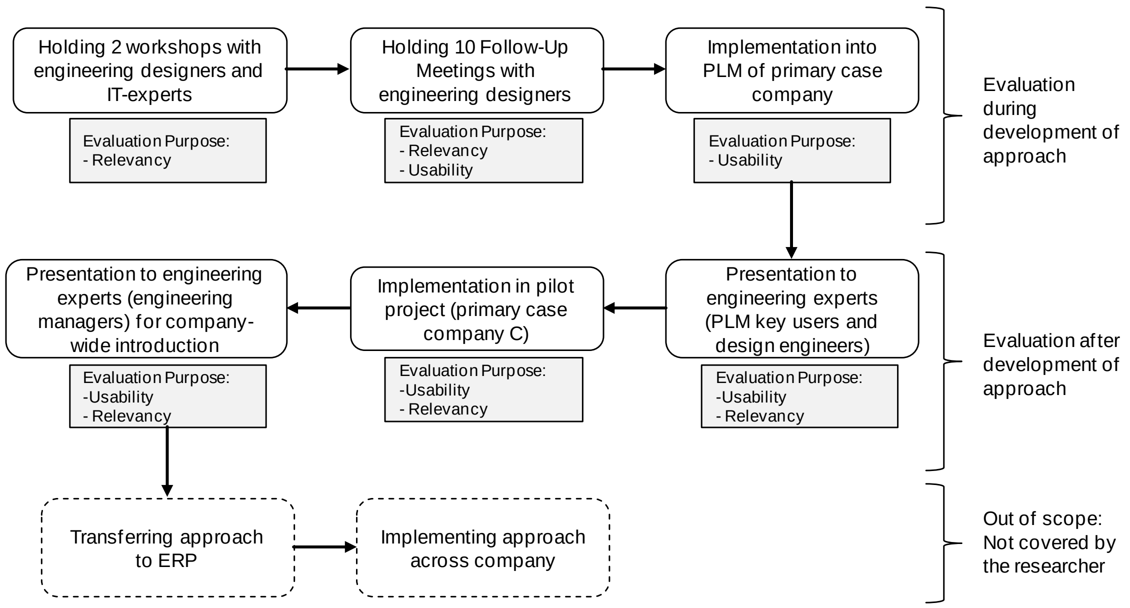


Figure 98: Evaluation of the approach for modularisation information provision

In the end, the approach has been evaluated successfully and proven to be an appropriate means of support to guide engineers toward more stable common modular reference architectures during later stages of the modular system development life cycle. This has been particularly achieved by providing central information and transparency about the modular system. An overview of the evaluation process is given in Figure 98.

8.6 Discussion

8.6.1 Research implications

Even though there is only a small number of publications researching the field of IT-integration for modular systems, the results of this study show that it is in fact an important field. Three parallel and similar studies have been compared to the results of this work. The study of Kissel et al. (2012) points at the importance of systematic product structure management and points at the importance of documents on different layers like accompanying specifications. However, there is no focus on actual representation of the modular system within standard IT-systems like it is extensively described within this work. The work of Kreymeier (2012) rather focuses on the representation of the market side of the modular system in order to efficiently configure derivative product variants. Even though, the market side has been considered through Modularisation Classification within this chapter, there has been no focus on this aspect.

The focus of this chapter was on internal aspects like protection of the modular reference architecture and efficient reuse for design engineers, but not on efficient configuration for the marketing or sales organisation. Thus, the work of Kreymeier (2012) can be seen as an extension or complementary to this research work. While this research work rather handles internal engineering aspects of modularisation information and complexity reduction, the work of Kreymeier (2012) focuses on creation of external variety. The research of Bruun et al. (2015) comes quite close to the work presented in this chapter and can therefore in its principle be confirmed by the findings of this study. As was done in this work, Bruun et al. (2015) use relational items like modules and interfaces to rebuild the conceptual modular system within a standard IT-system. However, there are also significant differences between the two works:

- Bruun et al. (2015, p. 102) apply their model during the development phase of the modular architecture and use it to evaluate module concepts and to generate process reports. Compared to the underlying work, there is no focus on stability of the modular architecture during later evolution and change of the modular system.
- Interfaces are a central element of the work of Bruun et al. (2015) and they are reflected as distinct items within the IT-system. While the importance of interfaces has been well recognised in the course of the underlying work, their separate treatment in standard IT-systems has been rejected after initial experiments. While there are definite advantages of this approach, it has been argued that separate treatment of inter-

faces increases complexity inherent in IT-systems. This increased complexity may not outweigh the benefits of handling interfaces separately in IT-systems. Nevertheless, in order to make sure that interfaces can be efficiently protected and managed, interface descriptions have been attached as vital modularisation information to module items for the purpose of this work. It has been argued that with this approach, the same benefits can be achieved without overloading the company with a large amount of additional items and relationships.

- Sometimes, the product architecture between different domains like design engineering, manufacturing engineering and service is defined differently. Hence, the approach of Bruun et al. (2015) offers the possibility to handle a design structure and a manufacturing structure within the standard IT-system. Such an approach is also possible with the underlying approach by extending the modular system from an overall, module or product view by the perspective of manufacturing or service. However, such a possibility has not been implemented in the course of this work due to its high complexity. Therefore, following further possibilities have been identified for such cases:
    - Initiating a programme for unification of company-wide product structures
    - Handling different structures with the help of different classification attributes
    - Taking advantage of offered “design-BOMs” and “manufacturing-BOMs” in available ERP-systems
    - Handling design structures within CAD and PLM while handling manufacturing structures within ERP

Altogether, all the approaches identified from literature have slightly different purposes. Thus, it would be interesting to see whether the approaches could be combined in order to get powerful IT-support for modular system development. This would mean that the market phase (Kreymeier (2012)), product architecture development phase (Bruun et al. (2015)) and the post-architecting phase (Heilemann et al. (2014) & this thesis chapter) would be covered by powerful modularisation information support.

8.6.2 Implications for practice

The Qualitative Analysis of this research thesis (see Chapter 5) has identified that there is strong need to increase provision of information and transparency during evolution and change phase of modular system development. So far, no research approach and no case company has been identified which has taken such an IT-integration approach for modular system development during that phase. Due to its development and validation in industry, the underlying approach is suggested to have high relevancy for industry. In essence, there are two use cases for practitioning design engineers that can be derived from the modular system information model:

- Ensuring reuse of already existing modular system elements

- Ensuring compliance of modular system elements with the common reference architecture specification

The use cases are enabled by following features of the modular system information model:

- Searching structural elements in IT-System:
  Queries in the IT-system based on the predefined attribute list guide engineers through the search process. The system structure represented in the IT-system helps engineers to get overview of all elements of the modular system.
- Protecting structural elements in IT-Systems is ensured through following points:
  - Architecture attributes mark structural elements of the modular system.
  - Module boundaries, geometry specifications and interface specification help to ensure stability of the modular system (master items control slave items).
  - The engineering change process in the IT-system is controlled according to the defined attribute “Variance Classification” under involvement of the “Modularisation Item Owner”.
  - Ownership and responsibility for modular system elements is defined across development projects.
- Administrating and linking product architecture information
  - Modularisation relevant data and documents are linked to respective modular system items throughout CAX-chain.
  - Central availability, maintenance and control of modular system information.
  - Successive and coordinated update of modular system information amid input from different derivative projects.

While all these points can be seen as effectively validated in industry, the application of the approach in industry raised some concerns. The first concern, the effort needed to establish the modularisation information model, could be rejected due to following facts: The information model can be established by creating a few additional items. The creation of each of these items takes less than a minute. For instance, for a modular system with 30 modularisation items, it takes approximately half an hour to create the items of the whole modular system. If modularisation classification is done with a predefined attribute list, it takes approximately 20 seconds to classify each item. Establishing relationships between modularisation items can be efficiently done by using the drag and drop functionality of standard IT-systems. Moreover, information that is attached to modular system items is not created for the purpose of the IT-integration of modularisation. Such kind of information (e.g. interface specifications, module specifications) should ideally be established already during concept phase of modular system development.

However, this study found out that the approach presented requires a new mindset and new project-overarching way of working. During application in industry, considerable time had to be spent on gaining understanding of involved design engineers for the new approach. This has to be considered by other companies embarking upon a similar approach.

While the underlying principles of this chapter might be generally applicable, particularly technical realisation of IT-integration of modular system information highly depends on the settings within the primary case company. Therefore, additional industrial settings have to be covered by further studies.

8.7 Summary

It was the aim of this chapter to develop an approach for provision of modularisation information in companies. Thus, requirements for such an approach were defined and relevant modularisation information that has to be represented within standard IT-systems of companies was identified. The approach was developed by taking into account that modular system items have to be created, classified and linked. Moreover, modularisation-relevant data and documents have to be attached to the relevant modularisation item. By following the approach, a modular system information model is established within the IT-system of the company. This helps to protect the architecture of the modular system, it fosters reuse of modular system elements and it supports companies in making powerful analyses and assessments on the modular system.

The presented modular system information model is an innovative approach which extends the traditional product-centred view in standard IT-systems to a product-overarching modular system view. This does not only support the understanding of engineering designers that they design module variants for the common modular system in addition to single derivative products, but it also supports them in doing so. Therefore, the presented approach supports companies to keep the modular system architecture stable during pre-architecting phases of the modular system life cycle, particularly by providing the required level of information and transparency.

9 Overall Discussion

The key active research reported in this thesis is included in Chapters 5 to 8. Each chapter has a specific individual discussion section in order to establish new and significant aspects of the research findings. This chapter will aim to show how these detailed findings relate to previous research work and contribute to the overall findings. Hence, this chapter elaborates what is new and significant from an overall perspective.

9.1 Synthesis of presented modularisation support

It is the purpose of this section to synthesise the active research Chapters 5 to 8 of this thesis and to summarise how the chapters link together. Figure 99 has been created to illustrate the connections. It shows that there are three different possibilities to look at namely a support-oriented thread (Thread A), a "gap"-oriented thread (Thread B) and an evaluation-oriented thread (Thread C).

9.1.1 Thread A: Support-aspect-oriented interconnection – focusing on two support aspects

Chapter 5 provides a support framework for modular system development with focus on stability (e.g. see Figure 57). The support framework of Chapter 5 includes the four support aspects "evaluation", "processes", "organisation" and "information". All of these aspects are covered broadly by Chapter 6 which frames them into an assessment framework which is set out to remedy the issues which have been encountered during modularisation transition in the past. As not all of these four aspects could be dealt with in depth for the scope of this work, the focus of this work was mainly delimited to the support aspects "evaluation" and "information". Figure 99 presents an overview of the synthesizing Thread A.

9.1.2 Thread B: Gap-oriented interconnection – focusing on three questions of the assessment framework without existing means of support

Based on the findings of Chapter 5, Chapter 6 presents a modularisation assessment framework with 22 questions which shall guide companies toward development of stable modular systems. While answering all of the 22 questions is important for modularisation transition, not all of the answers to these questions require novel scientific research. However, the three questions from Figure 99 (see Sections 6.5.5 and 6.5.6) have been identified as significant and as requiring novel scientific research in order to being able to answer them. In other words, a significant research gap has been identified behind these questions. Therefore, Chapter 7 and Chapter 8 were created to provide detailed support for answering the three questions of Figure 99 or Sections 6.5.5 and 6.5.6 respectively.

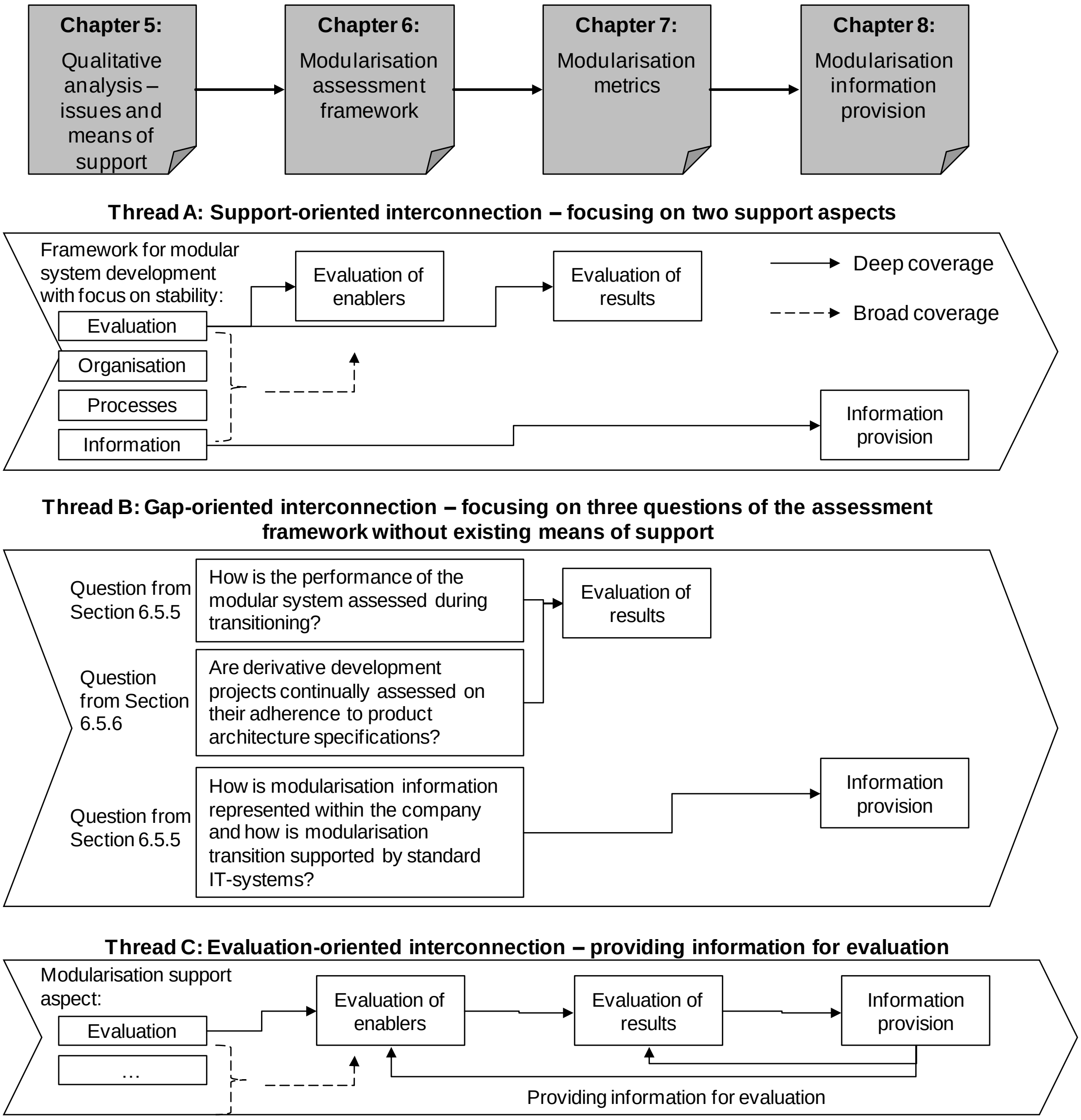


Figure 99: Synthesizing descriptive and prescriptive chapters of this research work

9.1.3 Thread C: Evaluation-oriented interconnection – providing information for evaluation

Thread C of Figure 99 can be seen as "evaluation-oriented" thread where Chapters 6, 7 and 8 all aim at the support aspect "evaluation" of Chapter 5. In this thread, Chapter 6 is created to evaluate "enablers" for modular system transition with a stable reference architecture. These enablers can either be assigned to the support aspects evaluation, process, organisation or information. After the assessment framework of Chapter 6 shall ensure that the important enablers are in place, it is the aim of Chapter 7 to measure the effect of the enablers on the actual performance of the modular system. In other words, it is the task of Chapter 7 to evaluate the results of modularisation transition. This interconnection between evaluating enablers (Chapter 6) and results (Chapter 7) is illustrated by Figure 59 at the beginning of Chapter 6. In this thread, it is the linked purpose of Chapter 8 to provide information for the evaluation of modular system development for the support developed in Chapters 6 and 7. The provision of information is a vital element for evaluating modular systems which has been found to be neglected by existing research (see Section 7.3).

9.1.4 Synchronisation on a higher level

Regardless whether Thread A, Thread B or Thread C is taken to synchronise Chapters 5 to 8, the thesis can also be synchronised on a higher level and simplified level which is shown by Figure 100.

The upper part of the figure shows the situation "before", i.e. the situation without the thesis. In this situation, Company A wants to launch modular products in order to achieve the benefits of modularisation. This works well during goal setting and during definition of the modular product architecture. However, after some time the company finds its architectures breaking apart and its products less modular than planned. In fact, after initial efforts the progress of transitioning vanished as Company A tumbled over too many issues.

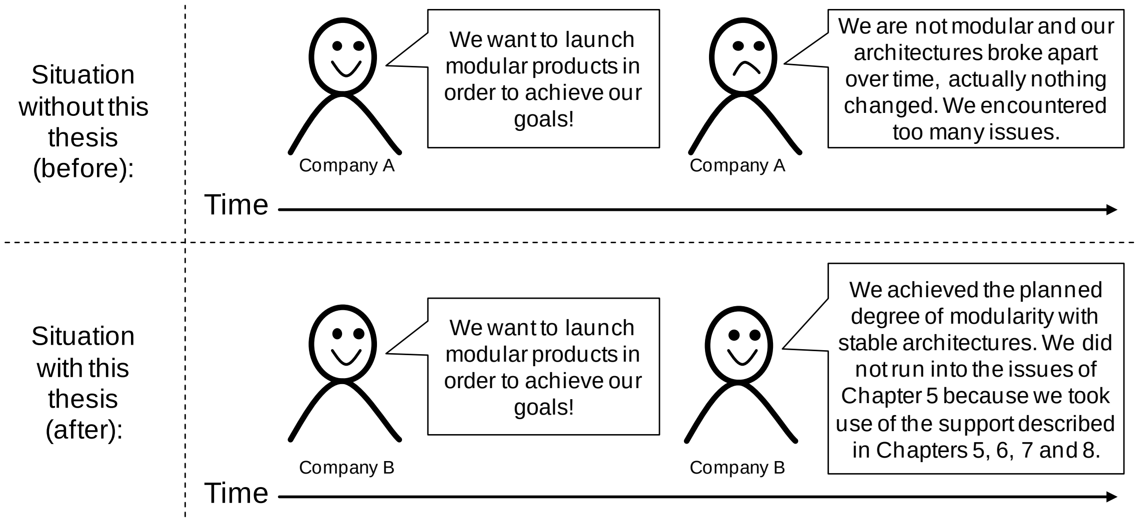


Figure 100: Situation with and without the support of Chapters 5 to 8

The lower part of Figure 100 shows the situation "after", i.e. the situation of Company B which is aware of the findings of this thesis. The situation at the beginning is the same as with Company A: It is the goal to launch modular products in order to achieve the promised benefits of modularisation. After some time, Company B is still successful with modularisation transition as it achieved the planned degree of modularity with stable architectures. Thus, it is likely that the company achieves the benefits of modularisation. It is the advantage of Company B that it is well-aware of the issues that a company is likely to encounter during modularisation transition. These issues are described in Chapter 5. Moreover, Company B is also savvy in overcoming these issues during all phases of the development life cycle after consulting this thesis. Chapters 5, 6, 7 and 8 of this thesis describe important factors, support and threads in order to overcome issues during modularisation transition.

9.2 Reflections on the specific contributions

9.2.1 Reflections on modular system development life cycle phases

- Pre-study phase before starting with modular system development

It is a major finding of this research work that it is necessary to first prepare the company for modular system development before starting with any other development activity (see Assessment Framework in Chapter 6). It has to be made sure that modular systems are not only implemented within development projects, but that they are implemented company-wide in order to achieve synergetic effects. Therefore, it has been suggested to work on management commitment, motivation of employees, cost-benefit analyses, commitment for the programme and planning the programme's implementation into the organisation during this phase.

The prevailing stream of existing research does not relate modularisation transition with such a phase (see Literature Review Chapter 3). Thus, this study is one of the first studies researching this phase of the modular system life cycle. The findings of this thesis directly address the call for future research in the field by Simpson et al. (2014, p. 777–787) where two requirements for this phase are mentioned.

Firstly, it is important to introduce extended "platform thinking to the entire continuum of product fulfilment, including customer platforms, brand platforms, product platforms" and the like (Simpson et al., 2014, p. 781).

Secondly, the future "platform extension" requires the involvement of "corporate-level product platform support" as it is not sufficient to see modular system development as product strategy. Rather, it is important to introduce it as "corporate strategy" (Simpson et al., 2014, p. 779) and "to get support and involvement from the entire organization to this major change" (Simpson et al., 2014, p. 780). How this can be done has been shown in the Assessment Framework in Chapter 6.

In short, the findings concerning the pre-study phase add to the current body of knowledge and can be seen as novel and significant.

- Importance of market phase, architecting phase and planning of the modular system

Given the literature review in Chapters 3 and the Qualitative Analysis in Chapter 5, it gets evident that existing research on modularisation transition follows a certain pattern which has been described earlier by Hofer and Gruenfelder (2001) in the context of product family and platform development. It seems that engineering design researchers embarking on modular system transition follow the same pattern. Usually, they start with the definition of the product/market strategy from which the definition of the product variant range is derived. This is achieved by taking market demands, technology trends and product functions into account. Afterwards, the product architecture is defined by describing modules, interfaces and design rules. The last step that is frequently considered concerns the plan of the modular system. The plan of the modular system usually contains the variability range of module variants and their alignment to products.

The results of this thesis (see Qualitative Analysis in Chapter 5 and Assessment Framework in Chapter 6) confirm literature that the phases in the modular system development life cycle from market strategy until planning of the modular system are vital and important enablers for stable modular system architectures. Additional support for the importance of research activities in these front-end phases comes from Simpson et al. (2014, p. 777–787).

- Back-end aspects after architecting and planning of the modular system

The literature review in Chapter 3 has shown that factors like manufacturing, production, the supply chain, serviceability or recycling have been frequently considered by modularisation support methods. In addition, Simpson et al. (2014, p. 787) stress the importance to further involve such back-end issues in modularisation research.

The importance of these back-end issues has been confirmed by this research thesis. However, this thesis strongly claims to include post-architecting phases of the development life cycle like concept testing, evolution and change of the modular system and the interplay between derivative and the central modularisation project (see Qualitative Analysis in Chapter 5 and Assessment Framework in Chapter 6). Exactly these post-architecting phases have been identified as most vulnerable to failure and as vital phase for the stability of the modular system. Surprisingly, this is the phase which has only raised very little attention by previous researchers. Thus, scarce research outputs like the need to synchronise platform and derivative development of Ponn (2015), Arnoscht (2011) and Nielsen (2010) can be confirmed. Moreover, Simpson et al. (Simpson et al., 2014, p. 787) end their outlook into future product family design research with the statement that "product platforms tend to have lifetimes that exceed the lifetime of the variants which makes the problem both challenging and relevant".

How this phase can be supported has been shown in the Assessment Framework in Chapter 6. Moreover, Chapters 7 to 8 deal with concrete support for these phases. Section 9.3 will go deeper into implications of this support.

In summary, the identified need for an emphasis on aspects after architecting and planning of the modular system contributing to stability is a strong, significant and novel finding. It is suggested that this finding will lead future research into a new direction.

9.2.2 Reflections on differences to the existing body of knowledge

In the course of this thesis, it has become clear that this work has moved the understanding of modularisation transition forward. Also the work is very extensive and covers some of the key design phases. It is thus not surprising that several interesting differences compared to existing research have been seen. These can be summarised as awareness about the overall issue of transition, stability of the modular system architecture during post-architecting phases and support during evolution and change of the modular system.

The following points will examine how the differences to the current body of knowledge can be explained.

- Applied research methodology

It is claimed that the main findings of this research thesis have their origin in the applied research methodology. According to DRM (Blessing and Chakrabarti, 2009), it was a Type 5 research project with a review-based Research Clarification, a comprehensive Descriptive Study I (see Chapter 5 Qualitative Analysis), a comprehensive Prescriptive Study (see Chapters 6 to 8) and an initial Descriptive Study II.

Most of the approaches from the literature review (see Chapter 3) apply a research project of Type 3 according to DRM (Blessing and Chakrabarti, 2009). This is a review-based Research Clarification followed by a review-based Descriptive Study I. The main core of such projects evolves from the comprehensive Prescriptive Study which is initially tested afterwards. While the applied research type is good to develop a tool, method or algorithm, it is less suitable to come to new findings that are really relevant for industrial practitioners due to neglecting overall understanding. Pirmoradi and Wang (2011, p. 1051) confirm that "most of the developed approaches and strategies are applied to a sample product family or an existing product in the market" and that among the developed approaches are "algorithms for handling large-scale platform selection and optimisation problems (...)". Therefore, a different research approach to the major stream has been applied.

It has to be mentioned that the research of this work started as Type 3 with the goal to develop a method for establishing modular architectures. However, after an explorative phase the research Type was switched to Type 5 because the overall issue of transitioning was seen as more significant an novel than developing another modularisation method.

The research approach of this work set out to research the overall issue of modularisation transition by studying "variabilities" of practice, implementation into daily practice and the long-term stability of the modular system. This was achieved by following elements that substantially differed from existing research approaches:

- The comprehensive Descriptive Study I was of explorative nature which was most suitable to get a differentiated view on the field of research.
- The longitudinal and extensive field study with industrial practitioners including observational, participant, action and intervention elements and the overall research setting allowed to place the real customer of engineering design research (i.e. design engineers and managers) into the centre. This allowed for focusing on those issues and means of support that are really relevant. For instance, existing support implemented in industry proved not to be relevant or unsuccessful after initial phases. This was the point where most of other research was terminated.

Based on these insights, new means of support could be developed (see Assessment Framework in Chapter 6, Modularisation Metrics in Chapter 7 and Modularisation IT-Integration in Chapter 8) and a new research direction could be suggested to other researchers.

- The extensive collaboration with industry allowed gaining the trust of industrial practitioners. This helped to get real insights into daily practice and to get unique opportunities to develop and test knowledge and support.
- The content of this work evolved by studying a primary case company with complementary secondary case companies. Thus, the focus of support development and tests was on a multitude of real industrial projects instead of developing support in an isolated environment prior to initial validation.

The differences to the existing body of knowledge could also be explained by the distinct focus on existing products of this research.

- Transitioning within existing products

The findings from the Qualitative Study in Chapter 5 and the developed means of support from Chapters 6 to 8 are suggested to be novel and relevant. However, it could be argued that there are dozens of examples from industry that present successful implementation of modular systems. Thus, it could be questionable why this research thesis challenges existing support and identifies significant issues during modularisation transition.

In contrast to existing research, there has been an adapted research focus of the research reported in this thesis. Firstly, the research focuses on projects without "given" common architecture. Thus, existing products had to be switched over and aligned to a new common modular reference architecture. Numerous examples from industry are in the context of "given" architectures. For cases of such given architectures the issues of this work focusing on transitioning might only be relevant to a certain extent. An example for a given architecture can be found in computer industry where the architecture has been virtually an industry standard (Baldwin and Clark, 2000, p. 9).

Secondly, most of the existing design research assumes that product development strictly follows the rules of "new product development". In new product development the issues of this work might not be relevant to the full extent. Starting from scratch on a blank sheet of paper would allow to build the environment of the modular system life cycle around the common modular reference architecture. However, the focus of this work was on a "brownfield" approach focusing on modularisation transition within existing products. Thus, the constant interplay between the modular system and derivative development projects had to be considered. This was the phase that attained particular attention of this research. Environments with "greenfield" approaches do not have to take care of this phase as the modular system can be properly established before modules are combined to derive new product variance. Such scenarios do not require taking care of the interplay between derivative and central development. However, such "greenfield" scenarios can rather be seen as exception than as the "brownfield" standard.

Thirdly, while there might be successful examples for platforms and modular products established in industry, the dimension to create commonality, synergy and variety out of current modular systems are set out to reach out for new heights.

The differences of the findings of this thesis compared to existing research can be explained to a certain extent to have their origin in an amended research methodology and focus. It is claimed that the amendment led to practical findings from daily practice and for daily practice of design engineers. It is suggested that this adds value to this thesis' significance, novelty and strengthens the validity of its implications.

9.3 Implications of the overall study

From an academic and practical point of view, the main overall contribution of this work might be that it changes the inherent thinking of researchers and design engineers. If this new understanding is used to change their behaviour, engineering design research will move a step forward based on the findings of this thesis. Moreover, based on the findings of the Qualitative Study in Chapter 5 and the Modularisation Assessment Framework in Chapter 6, researchers get a new research direction that really will have an impact in industry.

Given the relevancy of the research topic, it can be assumed that a considerable number of companies will embark upon modularisation transition in the future. However, this study pointed out that there might be major pitfalls on the path toward modular system development.

It is assumed that industrial practitioners are highly interested how they can avoid the issues that were detected during the Qualitative Study in Chapter 5. Moreover, it is significant for practitioners to know which kind of support has a positive "cost-benefit" analysis. For instance, it has been shown that industrial practitioners should not overdose traditional modularisation methods. Rather, practitioners should spend their resources on above mentioned aspects to prepare the organisation for modularisation transition and on support for the evolution and change phase of the modular system. Therefore, a modularisation support framework as guidance for further support has been developed (see Section 5.5).

The Modularisation Assessment Framework presented in Chapter 6 gives guidance to industrial practitioners on what to consider during modularisation transition in the entire modular system life cycle. For most of the steps described in the Modularisation Assessment Framework, it is up to industry to decide how they implement the suggested step in detail. This has to be done accordingly to their respective situation because research in the field has shown that existing modularisation methods are sometimes overly detailed and not suitable to the real needs of practitioners. Thus, it is the clear message of the Modularisation Assessment Framework to say "these are the steps that you have to consider, how you will implement them in detail depends on your specific situation".

However, the Modularisation Assessment Framework has also shown that there are post-architecting phases concerning the evolution and change of the modular system, where there is no sufficient pre-existing support and where application of detailed support is not straight forward. Nevertheless, for the stability of the modular system, the post architecting phases are of significant importance. Therefore, it has been found out that detailed support during these phases has to be developed. Following support is suggested for this phase:

- Modularisation processes: This kind of support for stable modular system development is treated within Chapter 6 (Modularisation Assessment Framework) and also researched by Arnoscht (2011), Nielsen (2010) and Munk (2011).
- Modularisation organisation is briefly introduced within Chapter 6 (Modularisation Assessment Framework) and researched in detail by Arnoscht (2011) and Oosterman (2001).
- Modularisation metrics: Chapter 7 gives detailed support on Modularisation Metrics with a particular focus on later phases of the modular system development life cycle. These metrics have been developed and validated in industry. In consequence, it is suggested that they can be directly used or slightly adapted by other companies for their modularisation transitioning purposes. The application of the presented modularisation metrics will help other industrial practitioners to enhance transparency and extrinsic motivation of involved employees to support the stability of the overall modular system. This can be achieved by avoiding the limitations of existing metrics which have also been presented in Chapter 7. In consequence it is claimed that the modularisation metrics of this work are novel and highly relevant.
- Modularisation information: Chapter 8 gives detailed support on how modular system relevant information can be made explicit by integrating it into standard IT-systems. The particular focus of modularisation IT-integration is on later phases of the modular system development life cycle. The presented approach gives guidance for industrial practitioners on how they can improve transparency about modular system development by storing modularisation-relevant items, attributes and documents in standard PDM and ERP systems. In their outlook for further research, Simpson et al. (2014, p. 779) point out that there are "opportunities abound for enhanced techniques for effectively capturing, storing, retrieving, and delivering information in support of product platform strategies". Moreover, Simpson et al. (2014, p. 779) add what is currently missing in industrial practice:

  > *There is a need to explore how documents can become primary vehicles for manipulating an information model in support of platforms, implying the broader opportunities for knowledge management to support platforms.*

It is claimed that this need is met by the approach presented in Chapter 8. In consequence, the findings can be seen as significant and innovative for both, academia and industry.

Altogether, this thesis has shown ways how fellow researchers and practitioners can remove the issues during modularisation transition which have been presented in Chapter 5.

9.4 Limitations

There are still some limitations and unanswered questions in the field.

Firstly, it has to be tested in other industries whether modularisation transition within existing products is indeed the superior choice or if a costly "greenfield" transition within central modular system development is in general the better choice.

Secondly, all findings of this work have their origin in a case-study with multiple cases. Thus, complete generalisability within industry cannot be provided which makes room for further validation in other industrial cases.

Thirdly, even though the presented Modularisation Metrics cover evaluation of modularisation transition, direct and "graspable" financial impacts of transitioning have not been studied within the course of this work. However, this field was identified as important enabler to "boost" the extrinsic motivation of involved employees to contribute to the overall modular system. This field is also mentioned as important "pillar" for further research by Simpson et al. (2014, p. 782).

9.5 Concluding remarks

The practical element of this research uncovered some of the shortcomings of existing research which mainly focuses on the market phase, architecting phase and planning of the modular system.

This thesis revealed totally new understanding about modular system development through focus on overall corporate aspects and emphasis of post architecting phases of the modular system development life cycle. These are the phases where existing engineering design support left the modular system vulnerable to diverging and, thus, to failure.

The new understanding generated in Chapter 5 aims at changing the behaviour of engineering practitioners to care for the stability of the modular system during evolution and change. Moreover, the presented support in Chapters 6 to 8 directly helps practitioners to keeping the modular system architecture stable by considering different aspects during all phases of the life cycle.

The findings of this research would have to be validated in other cases within different industries. Moreover, other research will have to show whether it is more beneficial to generally make the transition in the presented "brownfield" approach or if it is more beneficial to transition from scratch within a "greenfield" approach.

The overall conclusion of this research thesis will be presented in the next chapter.

10 Conclusions

This has been a long and wide-ranging research programme taking more than five years. It has involved intimate research in 30 distinct cases in ten industrial groups and eight countries. A "case" can be seen as either a separate company or as a distinct organisation of the same "group". For instance, different business units or divisions with a different product portfolio or industrial setting of the same group have been considered as distinct case. There have been a variety of studies ranging from a large and quite rare approximately three year longitudinal industrial study to a variety of medium-size and short studies over the overall period. This has given a very rich foundation to the findings of this work. It has been broadly based on the core Design Research Methodology (DRM) approach. The extensive nature of the collaborating case companies are detailed in Appendix B

This chapter thus presents the main conclusions that can be derived from the research presented in this thesis. This section starts with a review of the research background and drivers for this research work. Second, a review of research objectives and how they have been achieved is presented. Third, the key research contribution and its significance are given. Fourth, implications on both, the field of knowledge and on industry is presented. Finally, potential for further research is outlined.

10.1 Research background and drivers

Establishing the right type of product architecture is seen as an important lever to balancing the trade off between internal complexity and external variety. Reports from literature and industry promise competitive advantage and boost in profitability through modular system development (see Chapter 2).

As the concepts of platform development and modularisation have been established in engineering design since years, there is plenty of support in this field. State of the art methods that support design engineers in establishing a certain type of product architecture work well for new product development projects with limited scope, if modularisation is seen as a step or overarching phase in the concept phase of a NPD project (see Chapter 2).

However, available literature does not consider that modularisation takes place in ongoing product development across different development projects where the character of the development projects varies from new product development to minor modifications on existing products. Establishing an extended modular system across several different development projects in such an environment is only weakly described in literature. Moreover, modularisation methods mainly aim at the market or concept phase of the modular system life cycle. If they consider back-end factors, they mainly focus on factors like manufacturability, supply chain issues, recyclability or serviceability. After clustering modularisation methods and allocating them to different life cycle phases, it got obvious that all phases of the traditional life cycle are directly or indirectly considered. However, the period of the development life cycle where existing artefacts are evolved, changed and modified has not been directly covered by any existing modularisation method (see Chapter 3).

There are only very few, limited or yet uncompleted studies that suggest how a company can make the *overall transition* toward modular system development where a high amount of derivative products can be derived by combining a common set of module variants. At the same time, more and more industries are urged to achieve higher commonality levels across products while still being able to differentiate their products. Therefore, the number of companies that are on the verge to transition towards modular system development is increasing. Prominent examples that intend to introduce cutting-edge modular systems include the MQB of Volkswagen or the platform of BMW with the goal to launch millions of highly customised cars based on only two different platforms (see Chapter 3).

Although such transition plans in industry are at the beginning or early implementation phase, problems and costly setbacks have been reported. This is also the case for the primary case company of this research, it has an extensive, diverse, complex and non-modular range of products. It is also trying to bring some “order” to its activities. Thus, it has been the main rationale of this research to investigate the issues during transitioning and to support companies in having a less cumbersome modularisation transition.

10.2 Review of research objectives

The aim of this research is to identify and test critical issues and important factors associated with support for the transition towards modular system development with stable product architectures.

Based on these findings, it is the aim to develop engineering design support for the transition.

This aim has been achieved by attaining the objectives outlined below.

10.2.1 Review of RO1 – elements of transition

In order to understand what problems companies encounter and what kind of existing support they can apply, it was the first objective of this research to identify and test vital elements for modularisation transition.

The research activities that were applied to achieve these objectives had their origin in participating in an industrial case study comprising mixed methods like participant-observer approach and semi-structured interviews. Moreover, field studies directly in industrial development projects have been conducted to test the suitability of existing support in order to remedy issues during modularisation transition (see Section 5.2).

Firstly, this objective was achieved by identifying critical issues that companies encounter during transitioning toward modular system development. These identified issues have been classified and analysed in order to present a coherent list of modularisation transitioning issues within this research work (see Section 5.3).

Secondly, based on these analyses, it was possible to establish important factors that must be in place for transitioning toward modular system development. These important factors to overcome above mentioned issues are classified and listed within this work. After critical issues and important factors for modularisation transition were identified and analysed, it was possible to evaluate existing support for transitioning toward modularisation with a focus on development life cycle models and modularisation methods (see Section 5.4).

Based on identified issues, important factors to be established and weaknesses of existing support, a modularisation support framework with focus on stability during all phases of the modular system life cycle has been developed (see Section 5.5). This support framework serves as guidance for the development of further modularisation transition support.

After achieving research objective one, there was still a gap between the identified problems that companies encounter during modularisation transition and available support. This gap can be summarised by the finding that modularisation transition is always jeopardised by more tempting short-term and project-individual goals. Therefore, “mechanisms” had to be installed that guide the engineering organisation toward development of modular systems amid prevailing short-term goals, product-individual goals and temptation to fall back to more individualised product development. These “mechanisms” have been addressed by research objectives 2 to 4.

10.2.2 Review of RO2 – Modularisation Assessment Framework

Research objective 2 was to develop a Modularisation Assessment Framework for companies that transition toward modular system development. This was achieved by identifying an appropriate scheme for assessment of modularisation transition and by developing and testing the framework for modularisation transition in industry.

The Modularisation Assessment Framework presented in this work is based on state of the art assessment frameworks that are applied by process auditors in industry. The actual content of the assessment framework is based upon the achievement of research objective 1 by addressing potential issues that can be avoided by making sure that important factors for modularisation transition are in place. The modularisation assessment framework presents modularisation “enablers” along the modular system life cycle (see Chapter 6). The framework gives hints but does not prescribe detailed support for modularisation transition. Validation of the audit scheme took place by applying it in industry and by doing expert evaluation with industrial practitioners.

10.2.3 Review of RO3 – Modularisation Metrics

Because the Modularisation Assessment Framework, as result of the previous research objectives, mainly addresses the "enabling" and overall side of modularisation, it is research objective 3 to focus on the results of modularisation transition. Therefore, modularisation metrics for transitioning toward modular system development were objected to be developed. First, this research objective has been addressed by deriving requirements for modularisation metrics applied in industry. The requirements were derived based on semi-structured interviews in industry and based on a survey study. Already existing metrics have been tested in order to find out use and limits of existing modularisation metrics. It is the main flaw of existing metrics that they do not consider information requirements that are necessary to calculate them and that they are mainly design to assist engineering designers in establishing a certain kind of product architecture instead of supporting a company in modularisation transition. In order to overcome this flaw, this research developed and tested metrics for modularisation transition in industry (see Chapter 7). These metrics were validated in different development projects in industry and by recurring expert interviews.

10.2.4 Review of RO4 – transition and IT-systems

In order to create transparency about modular systems and to provide modularisation information for the results of research objectives 2 and 3, it is research objective 4 to develop an approach for provision of modularisation information in companies. This research objective was achieved by first identifying requirements for provision of modularisation information in a survey study. Based on these requirements, it was possible to identify relevant information for modularisation transition that must be captured within standard IT-systems of companies. This profound base enabled to come up with the core achievement of this research objective which is to develop and test an approach for integration of modularisation information into standard industrial IT-systems (see Chapter 8). The IT-integration approach was validated by a constant feedback cycle and implementation in industry.

10.3 Key research contributions and significance

The key contributions in the four research areas are summarised below. Underpinning data are included in Appendix D.

10.3.1 Deeper understanding about modularisation transition (see Chapter 5)

When considering modularisation transition, it is not enough to look at the step „modularisation" in the design process. Traditional models of the product development life cycle fall short of providing a sound base for the process of modular system development. Therefore, such a process model was introduced in the course of this work. Three different models how modularisation transition can take place were derived from literature and field studies in industry. The first alternative is similar to the NPD approach while giving special attention to the step modularisation during the design process. The second transitioning alternative takes place in a setting where a dedicated team and a dedicated span of time is devoted to modular system development. This alternative is less prone to failure but highly resource-intensive. Consequently, because most companies cannot afford to dedicate additional resources, they have to make the transition in parallel to ongoing product development within existing products (alternative 3). The three identified modular system development alternatives have the same phases and comprise a market phase, a product architecture planning phase, a product architecture design phase and evolvement of the modular system (see Section 5.4.1). The phase evolvement of the modular system can be further broken down into development of modules, development of derivative products and derivation of products from the modular system. It has been shown that the evolvement phase after architecting is the phase which gained only little attention by existing research and by existing development life cycle models.

- Issues during modularisation transition (see Section 5.4)

The common scheme among the development life cycle models of the modular system development process has been used to classify issues that arise during modular system development. The phases of this compressed model comprise the market phase, concept phase, architecture planning phase and evolvement and change of the modular system. It has been shown that while all phases contain the risk of considerable issues, most issues arise during evolution and change of the modular system.

An issue of modularisation transition is that the company has to initially invest more while benefits of modularisation do not arise before several product life cycle of a company's products have taken place. There are no direct benefits of the modularisation project itself. In contrary, there are only costs for the modularisation project in the accounting books. The "winners" of a modularisation project are in turn product development projects that rely on a modular system but that did not contribute to the modular system. Thus, motivation to contribute to the modular system instead of solely focusing on single derivative projects has been seen as a major issue during modularisation transition.

In transitioning companies, products were originally designed to cover the scope of only those requirements derived in the direct product development project, but not for a broader scope of requirements for other development projects. Hence, the architectures of existing product portfolios are vulnerable to unexpected market changes and cannot accommodate a large variety of products. Moreover, while planning the modular system it was found out that companies frequently struggle to capture the market requirements across the extended range of products due to lack of motivation and resources. Another issue is that in companies transitioning within existing products there are time-constraints in derivative development projects and pressure to bring new solutions as fast as possible to the market. This is detrimental for the initial modularisation project. There are always short-term goals that beat the long-term and costly concept of modularisation from the very beginning. Hence, lack of time and resources was identified as major obstacle to successful modularisation transition.

The study also found out that in addition to a different view on the product architecture across development projects, different company functions may have different views on the product architecture. Moreover, engineering departments often miss the overview of the overall modular system and information about cross-project specifications they have to adhere to. This intensifies the problem of establishing and deploying common modular architectures across the company. In consequence, lack of information and transparency has been identified as major issue during modularisation transition.

It has been shown that above mentioned main issues and diverse other disturbing factors like the cost reduction of single products, last minute wishes from product management, a lack of feasibility of the planned concept and time constraints to pursue the modular concept endanger the stability of the originally planned common modular product architecture, particularly during post-architecting phases. This means that the phase evolution and change of the modular system with least existing support is the phase where most of the issues occur that undermine the stability of the modular system.

In sum, engineers and managers always find reasons why not to stick to the originally planned common modular reference architecture. It is claimed that in product development with existing means of support, they neither have enough time, resources, motivation, information or transparency to appropriately contribute to the modular system during all life cycle phases. This endangers the stability of the modular system, especially during later phases of the modular system development life cycle (see Section 5.3).

- Important Factors for modularisation transition (see Chapter 5)

Transitioning companies need factors and support in place that advocate the overall and long-term goals of the modular system in opposition to short-term goals of single products and derivative development projects.

The transition towards modularisation requires a new way of working for engineers. This can be summarised through following points:

- Planning the product architecture, i.e. requirements phase and design of the product architecture requires to be on a fixed and much broader base than engineers and managers are originally used to.
- A dedicated and standardised process how the product architecture is created across different products has to be in place for transitioning.
- It is vital that a process is established how the modular system is evolved and how the interplay between single derivative products and the modular system takes place.

  - Regular assessments have to take place that guide a company from its natural default setting in single product thinking toward thinking in common modular systems with all accompanying measures that this entails.
  - Information has to be provided that broadens the scope of engineers and managers from single product thinking toward thinking in common modular system.
  - It has to be ensured that information for efficient modularisation assessment is provided.
- Support for modularisation transition (see Section 5.4 and 5.5)

Prior to transitioning to modular system development, there must be a well-grounded decision to pursue such a strategy. This decision has to be prepared by conducting a cost-benefit analysis how it was suggested within this work in order to highlight whether there is potential for modular system development or not.

If there is potential for such a strategy and if this potential cannot be achieved by applying less costly measures like defining standard parts, modularising existing platforms, agreeing upon common platform cores, or implementing part catalogues, modularisation transition can start with the implementation phase.

If a company transitions toward modularisation, one has to look at support from following fields: processes, evaluation, organisation and information. The use and limits of available support from these fields was tested.

Only little use was found in optimisation algorithms for product architecting. It was found that rather than a single method, many factors must be established within companies and that optimisation methods do not yield the promised results.

It is an important precondition for successful modularisation transition that the role of product management is emphasised and that market-knowledge about all products derived from modular system is well-known, proved and accepted by all involved roles.

It was shown that newly applied modularisation measures like modularisation methods consume a considerable amount of time and nerves of engineers so that they are prone to failure from the very beginning. Therefore, modularisation support must be integrated into existing company processes and monitored for appropriate application. Novel holistic support has to be established and implemented in the course of the modularisation programme.

In the course of this work, the new concept of stability has been established. Stability is important for the evolvement and change of the modular system. It means compliance to product architecture rules across projects and low deviation from architecture plans under constant update of architecture plans. If companies fail in managing stability of their modular system, they endanger the success of the whole modularisation project. As a result, a modularisation support framework with focus on stability has been developed as guidance for further research.

As soon as appropriate transitioning support is available to overcome the issues and to make sure that important factors for modularisation transition are in place, it has to be constantly monitored. Thus, enablers that ensure stability of the modular system have to be assessed and how the performance of modularisation transition develops during the modular system life cycle has to be monitored. Installed modularisation measures have to be taken as advocates for long-term modularisation goals compared to short-term goals of single products and single projects. In sum, the company has to be constantly pulled toward the overall goals of the modular system.

10.3.2 Modularisation Assessment Framework (see Chapter 6)

Modularisation transition requires making modifications to the engineering design processes and organisation if the transition shall lead to stable product architectures. If these modifications are not constantly measured and defended, there is the risk that they are not adequately lived or removed because of cost pressure.

A comprehensive view on the understanding of the modular system development life cycle within this work has been presented in Section 6.4 as base for the assessment framework. The developed modularisation assessment framework of this study supports companies to find out where their processes have gaps that endanger the transition process.

Moreover, the introduced framework helps companies to constantly monitor whether important support for modularisation is stringently lived. If there are gaps, the assessment framework shows how the flaws of current processes can be removed adequately.

The assessment framework covers all phases of the modular system development life cycle. It aims at reducing the issues that have been identified during each phase. According to occurrence of issues during the phases, the Modularisation Assessment Framework has a particular focus on the later evolution and change phase of the modular system. This makes the assessment framework a novel and relevant means of support in the researched field.

10.3.3 Modularisation Metrics (see Chapter 7)

The proposed metrics of this study measure how modular the product architecture of the considered product portfolio is, how the products stick to established product architecture rules, how stable the modular system is and how the trade-off between internal complexity and external variety develops over the transition period and afterwards.

The newly developed portfolio of metrics gives adequate transparency in order to physically track modularisation transition, but also the trade-off between short-term company goals and long-term goals related to modularisation. From a technical perspective, the metrics assess adherence of derivative products to the specifications of the common modular reference architecture. By integrating the metrics into the company evaluation system, they help to improve the motivation and understanding of involved employees to contribute to the overall modular system. It is suggested that this directly supports the stability of the modular system. Moreover, in contrast to existing metrics, the modularisation metrics of this work are efficiently computable and have a particular focus on the evolution and change phase where the modular system is most vulnerable to diverge. Thus, the metrics are a major contribution to the researched field and are novel in their application.

10.3.4 Modularisation information provision (see Chapter 8)

In order to shift the focus from single products toward the modular system, to gain transparency about product architectures and to assist efficient modularisation assessment by providing sound input, this support concerns the provision of modularisation information in companies. Therefore, the study introduced an approach for IT-integration of modularisation information into core IT-systems of companies.

The approach consists of four steps. First, the items of the modular system are created. Second, the items of the modular system are classified with modularisation attributes. Third, relationships between items of the modular system are established. Fourth, overarching product architecture information is assigned to respective items. The result of the IT-integration approach is an innovative modular system information model which represents a generic module view in addition to a derivative product view.

Validation activities of the method showed that the approach helps engineers to make sure that their products do meet the same architecture specifications across products and generations. Moreover, validation showed that the provision of modularisation information is a supportive enabler for reuse of artefacts, protection of the modular system and for architecture-relevant assessments. The distinct feature of this support is its focus on post-architecting phases of the modular system life cycle.

Summarising this section, this research is different to existing work as it covers following innovative aspects:

This work goes beyond the thinking that modularisation support ends with establishing the product architecture. Rather, this work takes into account that the evolvement of products and modular systems takes place in an interplay between different parallel and succeeding projects where engineering work mainly comprises engineering changes, modifications, reuse and adaption. Thus, this work strongly contributes to the stability of modular systems during later phases.

10.4 Implications

At the beginning of this research, no support for modularisation transition could be identified in literature. In the meantime, very few studies emerged that handle modularisation transition of companies. It is suggested that this thesis raises awareness about this topic in the academic and industrial world. Consequently, it can be used as base for transitioning in industry and for further research about modularisation transition and stability of modular systems.

10.4.1 Implications for the field of knowledge

While literature describes modularisation in the context of the traditional new product development processes, this work advocates the usage of an adopted modular system development life cycle (see Section 6.4).

Categorising issues and support along this modular system development life cycle allowed to come to new insights for modularisation research. While previous work strongly focused on the market and concept phase, this work focussed on the real demands of industry and came to the result that researchers should rather focus on the post-concept phases for modular system development (e.g. evolution and change of the modular system).

The approach to support companies in making the transition with a modularisation assessment framework is unprecedented in this field of research, though, it has been proven as appropriate means of support in the course of this work. In addition to the fact that it is a new kind of means of support for modularisation transition, the focus of its content on a pre-study phase and on evolution and change of the modular system should change the understanding of contemporary researchers on what is really important in industry.

Even though, there is already a plethora of modularisation metrics available in literature, this work has shown that there is still a gap in computing the right metrics for modularisation transition efficiently and in the application of the metrics during later post-architecting phases. It is assumed that this will give a new direction to the prevailing research stream about abstract modularisation metrics. It is suggested that research takes into account more concrete needs of industry like proposed in this work, especially the need to support the stability of the modular system during later development life cycle phases.

For academia, it is important to notice that the best support for modularisation is of no use if it cannot be applied in practice, e.g. because of lack of resources or because of lack of available information. The IT-integration of modularisation presented in this work shows how modularisation support can be assisted with provision of information for efficient application in practice.

Altogether, it is suggested that this work changes the understanding and behaviour of contemporary researchers toward improving time, motivation, information and transparency of engineers throughout the whole modular system development life cycle. Moreover, it is suggested that their attention is directed during later phases of the development life cycle (e.g. evolution and change phase) in order to provide stability of the modular system.

10.4.2 Implications for industry

The suggested support framework for modular system development with a focus on stability shows practitioners where issues during the development life cycle are to be expected and how these issues can be avoided efficiently by applying appropriate means of support.

The presented modularisation assessment framework can be used by industry in different ways. First, the industrial study showed there was the intention to use it itself as driver for the modularisation initiative by reporting to top-management where the weak spots of a certain business unit concerning modularisation are. Second, in industry it is also intended to include parts of the modularisation assessment framework into milestone review meetings along the modular system and product development process.

The modularisation metrics are applied and further developed in the primary case company. Their use is ongoing while they present the main means of measuring the overall modularisation programme within the company.

The proposed IT-integration approach has been successfully implemented into the PDM system of the primary case company. This concept is currently extended and implemented to the ERP environment.

It is claimed that what works well for the presented cases covered by this study will also be supportive for other companies during modularisation transition. Therefore, it is claimed to be helpful that the research was designed, for instance by triangulation, in a way to produce generalisable findings.

In sum, this research thesis presents new understanding about the modular system life cycle for industry. For the first time in research, industry gets a compressed overview about issues and support throughout the whole modular system life cycle. Moreover, a set of means of support for evolution and change phase of the modular system is provided, a phase with only scarce pre-existing support for industry.

10.5 Limitations of the study and implications for further research

While the findings of this study are certainly valid for “bigger” companies that develop products in a cross-market, cross-brand or cross-country environment, they might not be fully applicable to companies with a small and narrow product portfolio. For instance, in cases where engineering actors of very small companies have full knowledge about the market and the architecture of all products in scope, the measures proposed in this work could seem like taking a sledgehammer to crack a nut. While there are so many different application areas where modularisation can be applied, this work focused on companies that want to embrace commonality across a wide range of products without compromising product variety.

Because this research thesis evolved from a set of case studies, more research in other companies and industries is needed in order to come to findings which are *broadly generalisable* from a scientific point of view. This includes that the overall perspective on modularisation transition has to be further studied. It would be interesting to see how other companies made the transition and how the real outcomes of their undertakings are.

The study has shown that it is an important factor to gauge overall goals of the company with the goals related to single development projects, products or product families. To be able to prioritise appropriately, this is especially true for *financial considerations*. Therefore, further studies, that investigate new means how the effects of modularisation can be measured at an early stage and that bring up financial results that directly show up in accounting books, would be very supportive for the transition process. In addition to financial aspects, further elements which motivate people to embrace common elements of the modular system have to be studied.

The thesis has demonstrated the importance of IT as enabler for modularisation. Going one step further in this area would open totally new research opportunities. First, *IT facilitates* modular design operations. It would be beneficial if modularisation evaluation generated extended real-time information about the benefits of the modular strategy, including financial aspects. Moreover, such evaluations could be applied to directly evaluate the impact of different product architecture alternatives and to determine the optimum degree of modularity based on real data, instead of relying on impact estimations. In addition, further efforts should be spent on researching how to efficiently handle interfaces of modular systems within IT-systems of large enterprises.

The presented modularisation IT-integration approach could also be extended to cover the complete architecture-relevant life cycle and to further contribute to *automated modular design*. This could already start during the pre-study phase with an automatic generator of modularisation transition potential. Such a feature would research the possibility of automatically analysing product structures and hierarchies within IT-systems. This could be used as starting point to establish “easy” to “difficult” modularisation transitioning routes. Another feature during more mature design phases could be an automated design generator for handling fixed and scalable parts within the modular reference architecture. It is also thinkable to develop an automated company-wide alert or assistance mechanism for cases where central architecture specifications are violated.

11 List of author's related publications

Research publications are usually used to disseminate research findings in academia, to refine research work and to get a broad validation perspective. The corresponding research publications (each at least double-blind peer-reviewed) which are directly contributing to the research objectives of this thesis are as follows:

Paper contributing to research objective 1 (RO1) – elements of transitioning

Heilemann, M., Schlueter, M., Culley, S. J. and Haase, H.-J. (2012) Methodologies toward product architecture improvement in theory and practice, In Proceedings of 12th International Design Conference - DESIGN 2012, Cavtat, Dubrovnik, Croatia, pp. 919–928.

Paper contributing to research objective 2 (RO2) – modularisation assessment framework

Heilemann, M. and Culley, S. J. (2015) Capability audit for modular system development - Assessing important factors for establishing and maintaining common modular system architectures, In Proceedings of 1st IEEE International Symposium on Systems Engineering, Rome, Italy.

Papers contributing to research objective (RO3) – modularisation metrics

Heilemann, M., Culley, S. J., Schlueter, M. and Haase, H.-J. (2013) Examination of modularization metrics in industry, In Proceedings of the 19th International Conference on Engineering Design (ICED13), Seoul, Korea, pp. 427–436.

Heilemann, M., Culley, S. J., Schlueter, M. and Lindemer, V. (2015) Assessing modularisation transition with metrics, In Proceedings of the 20th International Conference on Engineering Design (ICED15), Milan, Italy.
(recognised as Reviewer's Favourite)

Paper contributing to research objective 4 (RO4) – modularisation information provision

Heilemann, M., Culley, S. J., Schlueter, M. and Lindemer, V. (2014) Method to integrate modular product architecture information into standard IT-systems, In Proceedings of 13th International Design Conference - DESIGN 2014, Cavtat, Dubrovnik, Croatia, pp. 863–872.

The research objectives and how they relate to the chapters of this thesis are further defined in Sections 1.7, 1.9, 4.1, 4.6 and Figure 5.

12 References

Adolph, N. (2005) Probleme mit Straßenbahn - Tram-Desaster belastet Siemens, *Der Tagesspiegel*, [online] Available from: http://www.tagesspiegel.de/wirtschaft/probleme-mit-strassenbahn-tram-desaster-belastet-siemens/517354.html (Accessed 29 April 2014).

Airbus (2014) Material, logistics and suppliers services, *Material sales*, [online] Available from: http://www.airbus.com/support/material-logistics/material-sales/ (Accessed 30 September 2014).

Alblas, A. A. and Wortmann, J. C. (2008) Maintaining product platforms in industrial machinery, In *Proceedings of the 10th International Design Conference, DESIGN 2008*, Dubrovnik, Croatia, The Design Society, pp. 261–270.

Alders, K. (2006) Komplexitäts- und Variantenmanagement der AUDI AG, In *Individualisierte Produkte : Komplexität beherrschen in Entwicklung und Produktion*, Lindemann, U., Zäh, M. F., and Reichwald, R. (eds.), Berlin, Springer.

Alizon, F., Khadke, K., Thevenot, H. J., Gershenson, J. K., Marion, T. J., Shooter, S. B. and Simpson, T. W. (2007) Frameworks for Product Family Design and Development, *Concurrent Engineering*, 15(2), pp. 187–199.

Alizon, F., Marion, T. J., Shooter, S. B. and Simpson, T. W. (2008) Product family design: Strategic principles to choose between product-driven and platform-driven processes, In *ASME 2008 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference - IDETC/CIE2008*, American Society of Mechanical Engineers, pp. 1019–1029.

Alizon, F., Shooter, S. B. and Simpson, T. W. (2009) Assessing and improving commonality and diversity within a product family, *Research in Engineering Design*, 20(4), pp. 241–253.

Alizon, F., Shooter, S. B. and Simpson, T. W. (2009) Henry Ford and the Model T: lessons for product platforming and mass customization, *Design Studies*, 30(5), pp. 588–605.

Alizon, F., Shooter, S. B. and Thevenot, H. J. (2006) Design structure matrix flow for improving identification and specification of modules, In *Proceedings of 2006 ASME International Design Engineering Technical Conferences and Computers and Information In Engineering Conference DETC2006*, Philadelphia, PA, United States, American Society of Mechanical Engineers.

Allen, K. and Carlson-Skalak, S. (1998) Defining product architecture during conceptual design, In *Proceedings of ASME Design Engineering Technical Conference, Design Theory and Methodological Conference, DETC1998, September 13-16*, Atlanta, GA, USA.

Andreasen, M. M., McAloone, T. and Mortensen, N. H. (2001) *Multi-Product Development - platforms and modularisation*, *The P* Programme's series of Insight Reports*, Lyngby, Technical University of Denmark.

Ansoff, H. I. (1957) Strategies for Diversification, *Harvard Business Review*, 37(5), pp. 113–124.

Arnoscht, J. (2011) Beherrschung von Komplexität bei der Gestaltung von Baukastensystemen, Thesis (PhD), Aachen, University Aachen.

Arthur, C. (2014) How to keep on selling smartphones when we've nearly all got one already, *The Guardian*, 27th April, [online] Available from: http://www.theguardian.com/technology/2014/apr/27/smartphone-market-saturation-apple-samsung (Accessed 26 August 2014).

Arthur, C. (2012) PC market shrinks year-on-year: can Windows 8 save it now?, *The Guardian*, 11th October, [online] Available from: http://www.theguardian.com/technology/2012/oct/11/pc-market-windows-8-dip (Accessed 26 August 2014).

Autobild.de (2013) VW Baukasten-Familie im Überblick: Die Kürzel der Zukunft, [online] Available from: http://www.autobild.de/artikel/vw-baukasten-familie-im-ueberblick-3902458.html (Accessed 30 December 2014).

Automobil Produktion (2014) Auf BMW-Mann Diess warten in Wolfsburg mächtige Aufgaben, [online] Available from: http://www.automobil-produktion.de/2014/12/auf-bmw-mann-diess-warten-in-wolfsburg-maechtige-aufgaben/ (Accessed 30 December 2014).

Aydin, A. O. and Güngör, A. (2005) Effective relational database approach to represent bills-of-materials, *International Journal of Production Research*, 43(6), pp. 1143–1170.

Bahns, T., Beckmann, G., Gebhardt, N. and Krause, D. (2015) Sustainability of modular product families, In *Proceedings of the 20th International Conference on Engineering Design (ICED15)*, Milan, Italy.

Bahns, T., Beckmann, G., Gebhardt, N. and Krause, D. (2015) Sustainability of modular product families, In *Proceedings of the 20th International Conference on Engineering Design (ICED15)*, Milan, Italy.

Bahns, T. and Krause, D. (2013) Pflege modularer Produktfamilien nach Markteintritt durch die Produktentwicklung, In *Design for X, Beiträge zum 24. DfX-Symposium*, Hamburg, pp. 209–220.

Baldwin, C. Y. and Clark, K. B. (2000) *Design Rules, Vol. 1: The Power of Modularity*, The MIT Press.

Baldwin, C. Y. and Woodard, J. C. (2008) The Architecture of Platforms - A Unified View, *Harvard Business School Working Paper*, No. 09-034.

Barton, R. A. and Thomas, A. J. (2009) Using the quickscan audit methodology as a precursor to a successful lean sigma implementation, In *Proceedings of International Conference on Computers and Industrial Engineering CIE 2009*, Troyes, France, IEEE, pp. 1314–1319.

Bäßler, R. (1987) Integration der montagegerechten Produktgestaltung in den Konstruktionsprozess, Thesis (PhD), University of Stuttgart.

Bathelt, J., Joensson, A., Meier, M. and Bacs, C. (2003) *Modularisierung SPS-gesteuerter mechatronischer Systeme*, *1. Paderborner Workshop Intelligente Mechatronische Systeme*, Paderborn, Heinz Nixdorf Institut.

Bayou, M. E. and de Korvin, A. (2008) Measuring the leanness of manufacturing systems—A case study of Ford Motor Company and General Motors, *Journal of Engineering and Technology Management*, 25(4), pp. 287–304.

Beckmann, G., Gebhardt, N. and Krause, D. (2014) Transfer of methods for developing modular product families into practice - an interview study, In *Proceedings of 13th International Design Conference - Design 2014*, Dubrovnik, Croatia, pp. 965–974.

Beisheim, N. and Stotz, F. (2013) Key Performance Indicators for Design and Engineering, In *Concurrent Engineering Approaches for Sustainable Product Development in a Multi-Disciplinary Environment*, Stjepandic, J., Rock, G., and Cees, B. (eds.), London, Springer, pp. 341--351.

Billies, M., Francisco, V., Krueger, P. and Linville, D. (2010) Participatory Action Research: Our Methodological Roots, *International Review of Qualitative Research*, 3(3), pp. 277–286.

Blackenfelt, M. (2001) Managing complexity by product modularisation - Balancing the aspects of technology and business during the design process, Thesis (PhD), Stockholm, Royal Institute of Technology.

Blackenfelt, M. (2000) Modularisation by relational matrices - a method for the consideration of strategic and functional aspects, In *Proceedings of the 5th WDK Workshop on Product Structuring*, Tampere.

Blackenfelt, M. and Stake, R. B. (1999) Classification of module drivers to support product modularisation by relational reasoning, In *Integration of process knowledge into design support systems - Proceedings of the 1999 CIRP International Design Seminar*, Kals, H. and Houten, F. van (eds.), Dordrecht, Netherlands, Kluwer Academic Publishers, pp. 449–458.

Blecker, T. and Abdelkafi, N. (2007) The Development of a Component Commonality Metric for Mass Customization, *IEEE Transactions on Engineering Management*, 54(1), pp. 70–85.

Blees, C., Henry, J. and Krause, D. (2008) Entwurf von modularen Produktarchitekturen unter Betrachtung unterschiedlicher Unternehmenssichten, In *Beiträge zum 19. Symposium 'Design for X'*, Neukirchen, pp. 149–158.

Blees, C., Henry, J. and Krause, D. (2009) Perspective-based development of modular product architectures, In *Proceedings of the 17th International Conference on Engineering Design (ICED'09)*, Palo Alto, pp. 95–106.

Blees, C., Kipp, T. and Krause, D. (2010) Development of Modular Product Families: Integration of Design for Variety and Modularization, In *Proceedings of the 8th International NordDesign Conference 2010*, Gothenburg, pp. 159–170.

Blees, C. and Krause, D. (2008) On the development of modular product structures: A differentiated approach, In *Proceedings of the DESIGN 2008*, Dubrovnik, pp. 301–308.

Blessing, L. T. M. and Chakrabarti, A. (2009) *DRM, a Design Research Methodology*, Dordrecht, Springer.

Boas, R. C. (2008) Commonality in complex product families: implications of divergence and lifecycle offsets, Thesis (PhD), Massachusetts Institute of Technology.

Boehm, B. (1995) Spiral model of software development and enhancement, *IEEE Engineering Management Review*, 23(4), pp. 69–81.

Boeing (2014) Boeing 777 Facts, *777 Family*, [online] Available from: http://www.boeing.com/boeing/commercial/777family/pf/pf_facts.page (Accessed 30 September 2014).

Boeriu, H. (2015) BMW CFO confirms CLAR - the new RWD platform and future hybrids, *BMWBLOG - News*, [online] Available from: http://www.bmwblog.com/2015/03/19/bmw-cfo-confirms-clar-the-new-rwd-platform-and-future-hybrids/ (Accessed 27 March 2015).

Borjesson, F. (2009) Improved output in modular function deployment using heuristics, In *International Conference on Engineering Design, ICED'09*, Stanford.

Bosch Group (2014) History - Bosch KWK Systeme, *Homepage Bosch Germany*, [online] Available from: http://www.bosch-kwk.de/en/about-us/history.html (Accessed 5 October 2014).

Bremmer, R. (1999) Cutting edge platforms, *Fincancial Times Automotive World*, June 1999, pp. 30–38.

Brewer, J. and Hunter, A. (2006) *Foundations of Multimethod Research*, Thousand Oaks, California, United States, SAGE Publications, Inc.

Bruun, H. P. L. and Mortensen, N. H. (2012) Visual product architecture modelling for structuring data in a PLM system, In *IFIP AICT - Advances in Information and Communication technology*, Montreal, Canada, Springer, pp. 598–611.

Bruun, H. P. L., Mortensen, N. H. and Harlou, U. (2013) PLM support for development of modular product families, In *Proceedings of the 19th International Conference on Engineering Design (ICED13)*, Seoul, Korea, pp. 379–388.

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

BusinessWeek (1997) CAN HONDA BUILD A WORLD CAR?, [online] Available from: http://www.businessweek.com/1997/36/b3543001.htm (Accessed 30 April 2012).

Caesar, C. (1991) *Kostenorientierte Gestaltungsmethodik für variantenreiche Serienprodukte: variant mode and effects analysis (VMEA)*, *Fortschrittberichte VDI : Reihe 2, Fertigungstechnik ; 218*, Düsseldorf, VDI-Verlag.

Camuffo, A. (2002) *Rolling Out a 'World Car': Globalization, Outsourcing and Modularity in the Auto Industry*, *International Motor Vehicle Program*, Working Paper, Cambridge, Massachusetts, Massachusetts Institute of Technology.

CARSCOOPS (2014) BMW to Limit Platform Architectures to Just Two, Mercedes to Four, [online] Available from: http://www.carscoops.com/2014/03/bmw-to-limit-platform-architectures-to.html (Accessed 27 March 2015).

Chandrasekaran, B., Stone, R. B. and McAdams, D. A. (2004) Developing design templates for product platform focused design., *Journal of Engineering Design*, 15(3), pp. 209–227.

Chao, L. P. and Ishii, K. (2004) Product platform design and the gate model: Lessons from industry case studies, In *Proceedings of 2004 ASME International Mechanical Engineering Congress and Exposition, IMECE*, Anaheim, CA, United States, ASME, pp. 13–22.

Chynoweth, C. (2011) How to avoid a tangled web, *The Sunday Times - Features*, [online] Available from: http://www.thesundaytimes.co.uk/sto/public/Appointments/article575337.ece (Accessed 5 October 2014).

Clark, K. B. and Fujimoto, T. (1989) Lead time in automobile product development. Explaining the Japanese advantage, *Journal of Engineering and Technology Management - JET-M*, 6(1), p. 25.

Clarkson, J. and Eckert, C. (eds.) (2005) *Design process improvement - A review of current practice*, London, England, Springer.

Collier, D. A. (1982) Aggregate Safety Stock Levels and Component Part Commonality, *Management Science*, 28(11), pp. 1296–1303.

Collier, D. A. (1981) The measurement and operating benefits of component part commonality, *Decision Sciences*, 12, pp. 85–96.

Connors, S. R. (1996) Informing decision makers and identifying niche opportunities for windpower, *Energy Policy*, 24(2), pp. 165–176.

Cooper, R. G. (2014) What's Next? After Stage-Gate, *Research Technology Management*, 57(1), pp. 20–31.

Coulter, S., L., McIntosh, M., W., Bras, B. and Rosen, D. W. (1998) Identification of limiting factors for improving design modularity, In *Proceedings of 1998 ASME Design Engineering Technical Conference*, Atlanta, Georgia.

Crawley, E., De Weck, O. L., Eppinger, S. D., Magee, C., Moses, J., Seering, W., Schindall, J., Wallace, D. and Whitney, D. E. (2004) The influence of architecture in engineering systems, In *Paper presented at the MIT Engineering Systems Symposium 2004 March 29-31*, Cambridge, Massachusetts.

Cusumano, M. A. and Nobeoka, K. (1998) *Thinking Beyond Lean: How Multi Project Management is Transforming Product Development at Toyota and Other Companies*, New York, USA, Free Press.

Cutrone, C. (2013) Cutting Down On Choice Is The Best Way To Make Better Decisions, *Business Insider*, [online] Available from: http://www.businessinsider.com/too-many-choices-are-bad-for-business-2012-12?op=1 (Accessed 1 May 2014).

Dahl, B. (1990) Entwicklung eines Konstruktionssystems zur Unterstützung der montagegerechten Produktgestaltung, Thesis (PhD), University of Stuttgart.

Dahmus, J. B., Gonzalez-Zugasti, J. P. and Otto, K. N. (2001) Modular product architecture, *Design Studies*.

Daimler (2014) Mercedes-Benz investiert mehr als drei Milliarden Euro in Deutschland: Mercedes-Benz richtet Pkw-Produktionsorganisation neu aus, *Unternehmen - Nachrichten*, [online] Available from: http://www.daimler.com/dccom/0-5-7153-49-1737801-1-0-0-0-0-0-9293-0-0-0-0-0-0-0-0.html (Accessed 11 January 2015).

Daniilidis, C., Enßlin, V., Eben, K. and Lindemann, U. (2011) A classification framework for product modularization methods, In *International Conference on Engineering Design, ICED11*, Copenhagen.

Day, R., Stone, R. B. and Lough, K. G. (2010) Validating module heuristics on large scale products, In *Proceedings of the ASME International Design Engineering Technical Conferences and Computers and Information in Engineering Conference 2009, DETC2009*, San Diego, CA, United states, American Society of Mechanical Engineers, pp. 1079–1087.

DeAngelis, T. (2004) Too many choices?, *Monitor on Psychology - American Psychology Association*, p. 56.

De Weck, O. L., Suh, E. S. and Chang, D. (2003) Product Family and Platform Portfolio Optimization, In *Proceedings of 2003 ASME Design Engineering Technical Conferences - Design Automation Conference*, Chicago, IL, United states.

De Weck, O. L., Suh, E. S. and Chang, D. (2004) *Product family strategy and platform design optimization*, MIT Working Paper, Cambridge, Massachusetts Institute of Technology.

Department of Trade and Industry (2010) *Energy consumption in the United Kingdom*, London, DTI.

Der Teckbote (2012) Mehr Automodelle, *Der Teckbote*, Kirchheim (Teck), 14th May, p. 7.

Deutsches Ärzteblatt (2013) Endoprothe-senregister Deutschland umfasst mehr als 35.000 Produkte, *Vermischtes*, [online] Available from: http://www.aerzteblatt.de/nachrichten/56209/Endoprothesenregister-Deutschland-umfasst-mehr-als-35-000-Produkte (Accessed 4 November 2014).

Dieter, G. E. and Schmidt, L. C. (2009) *Engineering Design*, 4th ed, New York, McGraw-Hill.

DIN Deutsches Institut für Normung e.V. (2009) *Leiten und Lenken für den nachhaltigen Erfolg einer Organisation – Ein Qualitätsmanagementansatz (ISO 9004:2009); Dreisprachige Fassung EN ISO 9004:2009 (Managing for the sustained success of an organization – A quality management approach (ISO 9004:2009); Trilingual version EN ISO 9004:2009)*, Berlin, Germany, Beuth Verlag GmbH.

DIN Deutsches Institut für Normung e.V. (2008) *Qualitätsmanagementsysteme – Anforderungen (ISO 9001:2008); Dreisprachige Fassung EN ISO 9001:2008 (Quality management systems – Requirements (ISO 9001:2008); Trilingual version EN ISO 9001:2008)*, Berlin, Germany, Beuth Verlag GmbH.

Döpke, S., Kress, S. and Kühl, A. (2009) Performance Measurement zur Unterstützung des Variantenmanagements - Konzeption und Implementierung eines kennzahlenorientierten Informationssystems auf Basis von SAP BI bei der KSB AG., *ZWF Zeitschrift für wirtschaftlichen Fabrikbetrieb*, pp. 965–970.

Du, X., Jiao, J. and Tseng, M. M. (2001) Architecture of Product Family: Fundamentals and Methodology, *Concurrent Engineering*, 9(4), pp. 309–325.

Du, X., Jiao, J. and Tseng, M. M. (2005) Understanding customer satisfaction in product customization, *The International Journal of Advanced Manufacturing Technology*, 31(3–4), pp. 396–406.

Eckert, C. M., Stacey, M. K. and Clarkson, P. J. (2003) The spiral of applied research: A methodological view on integrated design research, In *Proceedings of the 14th International Conference on Engineering Design ICED'03*, Stockholm, Sweden.

Eeles, P. and Cripps, P. (2010) *The Process of Software Architecting*, Boston, USA, Addison-Wesley.

Ehrlenspiel, K. (2003) *Integrierte Produktentwicklung : Denkabläufe, Methodeneinsatz, Zusammenarbeit*, 2nd ed, Munich, Hanser.

Ehrlenspiel, K. (2007) *Integrierte Produktentwicklung : Methoden für Prozessintegration, Produkterstellung und Konstruktion*, 3rd ed, Munich, Hanser.

Ehrlenspiel, K., Kiewert, A. and Lindemann, U. (2007) *Kostengünstig Entwickeln und Konstruieren: Kostenmanagement bei der integrierten Produktentwicklung*, 6th ed, Berlin, Springer-Verlag.

Eilmus, S., Ripperda, S. and Krause, D. (2013) Towards the development of commonal product programs, In *Proceedings of 19th International Conference on Engineering Design (ICED13)*, Seoul, Korea.

Eisenhardt, K. M. (1989) Building Theories from Case Study Research, *Academy of Management Review*, 14(4), pp. 532–550.

Emmatty, F. J. and Sarmah, S. P. (2012) Modular product development through platform-based design and DFMA, *Journal of Engineering Design*, 23(9), pp. 696–714.

Eppinger, S. D. and Browning, T. R. (2012) Design structure matrix methods and applications, In *Engineering systems*, Cambridge, MA, USA, MIT Press.

Erens, F. and Verhulst, K. (1997) Architectures for product families, *Computers in Industry*, 33, pp. 165–178.

Ericsson, A. and Erixon, G. (1999) *Controlling Design Variants: Modular Product Platforms*, Dearborn, Society of Manufacturing Engineers.

Erixon, G. (1998) A method for product modularisation, Thesis (PhD), Stockholm, Royal Institute of Technology.

Erixon, G. (1996) Modular Function Deployment (MFD): Support for good product structure creation, In *Proceedings of the 2nd WDK Workshop on product structuring*, Delft University of Technology, Netherlands, pp. 181–196.

Erixon, G., von Yxkull, A. and Arnström, A. (1996) Modularity – the Basis for Product and Factory Reengineering, *CIRP Annals - Manufacturing Technology*, 45(1), pp. 1–6.

Ernst, R. and Kamrad, B. (2000) Evaluation of supply chain structures through modularization and postponement, *European Journal of Operational Research*, 124(3), pp. 495–510.

Farrell, R. S. and Simpson, T. W. (2010) Improving cost effectiveness in an existing product line using component product platforms, *International Journal of Production Research*, 48(11), pp. 3299–3317.

Feitzinger, E. and Lee, H. L. (1997) Mass Customization at Hewlett-Packard: The Power of Postponement, *Harvard Business Review*, pp. 116–121.

Finger, S. and Dixon, J. R. (1989) A review of research in mechanical engineering design. Part I: Descriptive, prescriptive, and computer-based models of design processes, *Research in Engineering Design*, 1(1), pp. 51–67.

Fixson, S. K. (2004) Assessing product architecture costing - product life cycles, allocation rules, and cost models, In *Proceedings of ASME 2004 Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, Salt Lake City, Utah.

Fixson, S. K. (2002) Linking modularity and cost: A methodology to assess cost implications of product architecture differencdes to support product design, Thesis (PhD), Massachusetts Institute of Technology.

Fixson, S. K. (2007) Modularity and Communality Research: Past Developments and Future Opportunities, *Concurrent Engineering: Research and Applications*, 15(2), pp. 85–111.

Fixson, S. K. (2005) Product architecture assessment: a tool to link product, process, and supply chain design decisions, *Journal of Operations Management*, 23(3–4), pp. 345–369.

Fixson, S. K. and Blanchard, P. J. (2001) Economic Analysis of Two Different Door Architectures, In *Proceedings of the 2001 SAE International Body Engineering Conference*, Detroit, Michigan.

Fixson, S. K. and Clark, J. P. (2002) On the link between modularity and cost - a methodology to assess cost implications of product architecture differences, In *Proceedings of the International Engineering Management Conference IEMC'02*, Cambridge.

Fleming, B. (2011) Automotive Electronics - Microcontroller Units in Automobiles, *IEEE Vehicular Technology Magazine*, September, pp. 4–8.

Flyvbjerg, B. (2011) Case Study, In *The Sage Handbook of Qualitative Research*, Denzin, N. K. and Lincoln, Y. S. (eds.), Thousand Oaks, California, Sage Publications, pp. 301–316.

Focus Online (2013) Renault-Nissan wird flexibler: Neuer Baukasten für 14 Modelle, *News*, [online] Available from: http://www.focus.de/auto/news/renault-nissan-wird-flexibler-neuer-baukasten-fuer-14-modelle_aid_1021019.html (Accessed 30 December 2014).

Ford, H. (1926) *Today and Tomorrow*, New York, USA, Doubleday and Company.

Ford, H. and Crowther, S. (2005) My Life and Work, *The Project Gutenberg EBook of My Life and Work by Henry Ford*, [online] Available from: http://www.gutenberg.org/dirs/etext05/hnfrd10.txt (Accessed 4 October 2014).

Franke, H.-J., Hesselbach, J., Firchau, N. L. and Huch, B. (2002) *Variantenmanagement in der Einzel- und Kleinserienfertigung*, München, Hanser.

Freitag, M. (2014) Produktstrategiechef verliert Aufgabe: Das Wechselspiel bei Volkswagen geht weiter, *Manager Magazin Online*, [online] Available from: http://www.manager-magazin.de/unternehmen/autoindustrie/volkswagen-produktstrategiechef-wolpert-kuemmert-sich-kuenftig-um-lkw-a-1008974-druck.html (Accessed 30 December 2014).

Friedman, T. L. (2007) *The World Is Flat - The Globalized World in the Twenty-first Century, Release 2.0, Updated & Expanded 2006 Edition*, London, England, Penguin Books Ltd.

Fujita, K. (2002) Product variety optimization under modular architecture, *Computer-Aided Design*, 34(12), pp. 953–965.

Fujita, K., Sakaguchi, H. and Akagi, S. (1999) Product variety deployment and its optimization under modular architecture and module commonalization, In *Proceedings of the 1999 ASME Design Engineering Technical Conferences*, Las Vegas, Nevada.

Fujita, K. and Yoshida, H. (2004) Product Variety Optimization Simultaneously Designing Module Combination and Module Attributes, *Concurrent Engineering*, 12(2), pp. 105–118.

Gebhardt, N., Bahns, T. and Krause, D. (2014) An example of visually supported design of modular product families, In *Proceedings of 24rd CIRP Design Conference*, Milano.

Gebhardt, N. and Krause, D. (2015) On the development of visualisation concepts as tools in product design, In *Proceedings of the 20th International Conference on Engineering Design (ICED15)*, Milan, Italy.

Gerring, J. (2007) *Case Study Research - Principles and Practices*, New York, USA, Cambridge University Press.

Gershenson, J. K., Prasad, G. J. and Allamneni, S. (1999) Modular product design: a life-cycle view, *Transactions of the Society for Design and Process Science*, 3(4), pp. 13–26.

Gershenson, J. K., Prasad, G. J. and Zhang, Y. (2004) Product modularity: measures and design methods, *Journal of Engineering Design*, 15(1), pp. 33–51.

Gibbs, G. R. (2007) *Analyzing Qualitative Data*, London, England, SAGE Publications Ltd.

Gillham, B. (2000a) *Developing a Questionnaire*, London, Continuum International Publishing Group.

Gillham, B. (2000b) *The Research Interview*, London, Continuum International Publishing Group.

Goepfert, J. (1998) *Modulare Produktentwicklung. Zur gemeinsamen Gestaltung von Technik und Organisation*, Wiesbaden, Gabler Verlag.

Goepfert, J. and Steinbrecher, M. (2000) Modulare Produktentwicklung leistet mehr - Warum Produktarchitektur und Projektorganisation gemeinsam gestaltet werden müssen, *Harvard Business Manager*.

Gomez, P. and Probst, G. (1997) *Die Praxis des ganzheitlichen Problemlösens*, 2nd ed, Bern, Paul Haupt Verlag.

Gomoll, W. (2013) Strategie: Volvo schaut sich bei VW das Baukastenprinzip ab, *Zeit Online*, [online] Available from: http://www.zeit.de/auto/2013-02/volvo-strategie-zukunft (Accessed 19 February 2013).

Gonzalez-Zugasti, J. P., Otto, K. N. and Baker, J. D. (2000) A Method for Architecting Product Platforms, *Research in Engineering Design*, 12(2), pp. 61–72.

Gonzalez-Zugasti, J. P., Otto, K. N. and Baker, J. D. (2001) Assessing value in platformed product family design, *Research in Engineering Design - Theory, Applications, and Concurrent Engineering*, 13(1), pp. 30–41.

Götzfried, M. (2013) Managing Complexity Induced by Product Variety in Manufacturing Companies - Complexity Evaluation and Integration in Decision-Making, Thesis (PhD), St. Gallen, University of St. Gallen.

Gu, P., Hashemian, S. and Sosale, S. (1997) An integrated modular design methodology for life-cycle engineering, *Annals of the CIRP*, 46(1), pp. 71–74.

Gu, P. and Sosale, S. (1999) Product modularization for life cycle engineering, *Robotics and Computer-Integrated Manufacturing*, 15(5), pp. 387–401.

Gubbi, J., Buyya, R., Marusic, S. and Palaniswami, M. (2013) Internet of Things (IoT): A vision, architectural elements, and future directions, *Future Generation Computer Systems*, 29, pp. 1645–1660.

Gubrium, J. F. and Holstein, J. A. (2001) *Handbook of Interview Research*, Thousand Oaks, California, United States, SAGE Publications, Inc.

Gudmundsson, A., Boer, H. and Corso, M. (2004) The implementation process of standardisation, *Journal of Manufacturing Technology Management*, 15(4), pp. 335–342.

Guo, F. and Gershenson, J. K. (2004) A Comparison of Modular Product Design Methods Based on Improvement and Iteration, *ASME Conference Proceedings*, 2004(46962a), pp. 261–269.

Hackl, J., Gumpinger, T. and Krause, D. (2014) Generating a meta-model for modularization methods, In *Proceedings of International Design Conference - DESIGN 2014*, Dubrovnik, Croatia, pp. 821–830.

Hales, C. (1987) Analysis of the engineering design process in an industrial context, Thesis (PhD), Cambridge, University of Cambridge.

Halman, J. I., Hofer, A. P. and van Vuuren, W. (2003) Platform-Driven Development of Product Families: Linking Theory with Practice, *Journal of Product Innovation Management*, 20(2), pp. 149–162.

Handelsblatt (2012a) Autos nach dem Prinzip Lego - Die neuen Produktions-Baukästen, *Test + Technik*, [online] Available from: http://www.handelsblatt.com/auto/test-technik/die-neuen-produktions-baukaesten-autos-nach-dem-prinzip-lego/7176548.html (Accessed 11 January 2015).

Handelsblatt (2012b) VWs Lkw-Baukasten ist gescheitert, *Industrie - Unternehmen*, [online] Available from: http://www.handelsblatt.com/unternehmen/industrie/medienbericht-vws-lkw-baukasten-ist-gescheitert/7155304.html (Accessed 12 February 2013).

Handelsblatt (2014) Wachstumspläne - Baukastenspiele bei Volkswagen, *Handelsblatt*, [online] Available from: http://www.handelsblatt.com/unternehmen/industrie/wachstumsplaene-baukastenspiele-bei-volkswagen/9610816.html (Accessed 15 March 2014).

Hansen, C. L., Mortensen, N. H., Hvam, L. and Harlou, U. (2012) Calculation of Complexity Costs – An Approach for Rationalizing a Product Program, In *Proceedings of Nord-Design Conference 2012*, Aalborg, Denmark.

Hansen, P. K. and Sun, H. (2010) An incremental approach to support realization of modularization benefits, In *Proceedings of 2010 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM)*, Macao, pp. 173–177.

Harlou, U. (2006) Developing product families based on architectures - Contribution to a theory of product families, Thesis (PhD), Lyngby, Technical University of Denmark.

Haskins, C., Forsberg, K., Krueger, M., Walden, D. and Hamelin, D. R. (eds.) (2011) *INCOSE Systems Engineering Handbook - A guide for system life cycle processes and activities*, ed. 3.2.2, San Diego, CA, United States, SE Handbook Working Group, INCOSE (International Council on Systems Engineering).

Haug, A., Hvam, L. and Mortensen, N. H. (2013) Reducing variety in product solution spaces of engineer-to-order companies: The case of Novenco A/S, *International Journal of Product Development*, 18(6), pp. 531–547.

Hegge, H. M. H. and Wortmann, J. C. (1991) Generic bill-of-material: a new product model, *International Journal of Production Economics*, 23(1–3), pp. 117–128.

Heilemann, M. and Culley, S. J. (2015) Capability audit for modular system development - Assessing important factors for establishing and maintaining common modular system architectures, In *Proceedings of 1st IEEE International Symposium on Systems Engineering*, Rome, Italy.

Heilemann, M., Culley, S. J., Schlueter, M. and Haase, H.-J. (2013) Examination of modularization metrics in industry, In *Proceedings of the 19th International Conference on Engineering Design (ICED13)*, Seoul, Korea, pp. 427–436.

Heilemann, M., Culley, S. J., Schlueter, M. and Lindemer, V. (2015) Assessing modularisation transition with metrics, In *Proceedings of the 20th International Conference on Engineering Design (ICED15)*, Milan, Italy.

Heilemann, M., Culley, S. J., Schlueter, M. and Lindemer, V. (2014) Method to integrate modular product architecture information into standard IT-systems, In *Proceedings of 13th International Design Conference - DESIGN 2014*, Cavtat, Dubrovnik, Croatia, pp. 863–872.

Heilemann, M., Schlueter, M., Culley, S. J. and Haase, H.-J. (2012) Methodologies toward product architecture improvement in theory and practice, In *Proceedings of 12th International Design Conference - DESIGN 2012*, Cavtat, Dubrovnik, Croatia, pp. 919–928.

Helmer, R., Yassine, A. and Meier, C. (2008) Systematic module and interface definition using component design structure matrix, *Journal of Engineering Design*, 21(6), pp. 647–675.

Hino, S. (2006) *Inside the Mind of Toyota: Management Principles for Enduring Growth*, New York, USA, Productivity Press.

Hips For You (2009) Smith & Nephew Total Hip Replacement Systems, *Hips for you - guide to total hip replacement*, [online] Available from: http://www.hipsforyou.com/s&ntotalhipdevices.php (Accessed 4 November 2014).

Hofer, A. P. and Gruenenfelder, M. (2001) Product Family Management Based on Platform Concepts, In *Proceedings of 2001 International Conference on Engineering Design, ICED'01*, Glasgow, pp. 491–498.

Holtta, K. and Otto, K. (2005) Incorporating design effort complexity measures in product architectural Design and Assessment, *Design Studies*, 26(5), pp. 445–564.

Holtta, K. and Otto, K. N. (2003) Incorporating design complexity measures in architectural assessment, In *Proceedings of DETC'03*, Chicago.

Holtta, K. and Salonen, M. (2003) Comparing three modularity methods, In *Proceedings of DETC'03*, Chicago.

Holtta, K., Tang, V. and Seering, W. (2003) Modularizing product architectures using dendrograms, In *Proceedings of ICED'03*, Stockholm.

Holtta-Otto, K., Chiriac, N. A., Lysy, D. and Suh, E. S. (2012) Comparative analysis of coupling modularity metrics, *Journal of Engineering Design*, 23(10–11), pp. 787–803.

Holtta-Otto, K. and De Weck, O. L. (2007) Degree of Modularity in Engineering Systems and Products with Technical and Business Constraints, *Concurrent Engineering*, 15(2), pp. 113–126.

Holzner, S. (2006) *How Dell Does It*, New York, McGraw-Hill.

Horvath, I. (2001) A contemporary survey of scientific research into engineering design, In *Proceedings of International Conference on Engineering Design ICED'01*, Glasgow, pp. 13–20.

Hrachowy, F. (2011) Quer gedacht. Volkswagen fuehrt den modularen Querbaukasten ein, *Krafthand*, pp. 64–66.

Huang, C. C. and Kusiak, A. (1998) Modularity in Design of Products and Systems, *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, 28(1), pp. 66–77.

Hubka, V. (1982) *Principles of Engineering Design*, London, Butterworth.

Hüner, K. M., Ofner, M. and Otto, B. (2009) Towards a maturity model for corporate data quality management, In *Proceedings of the 2009 ACM Symposium on Applied Computing*, *SAC '09*, New York, NY, USA, ACM, pp. 231–238.

ID-Consult (2015) METUS Software, [online] Available from: http://id-consult.com/metus/metus-software (Accessed 8 January 2016).

Institute of Electrical and Electronics Engineers (IEEE) (2010) Systems and software engineering – Vocabulary, In *ISO/IEC/IEEE 24765:2010(E)*, New York, USA, pp. 1–418.

ISO/IEC/IEEE (2015) International Standard ISO/IEC/IEEE 15288:2015-05 Systems and software engineering - System life cycle processes,.

Jacobs, M., Droge, C., Vickery, S. K. and Calantone, R. (2011) Product and process modularity's effects on manufacturing agility and firm growth performance, *Journal of Product Innovation Management*, 28(1), pp. 123–137.

Jasmine, K. S. and Vasantha, R. (2010) A New Capability Maturity Model For Reuse Based Software Development process, *IACSIT International Journal of Engineering and Technology*, 2(1), pp. 112–116.

Ji, Y., Jiao, R. J., Chen, L. and Wu, C. (2013) Green modular design for material efficiency: a leader–follower joint optimization model, *Journal of Cleaner Production*, 41, pp. 187–201.

Jiao, J., Simpson, T. W. and Siddique, Z. (2007) Product family design and platform-based product development: a state-of-the-art review, *Journal of Intelligent Manufacturing*, 18(1), pp. 5–29.

Jiao, J. and Tseng, M. M. (1999) A methodology of developing product family architecture for mass customization, *Journal of Intelligent Manufacturing*, 10, pp. 3–20.

Jiao, J. and Tseng, M. M. (2000) Understanding product family for mass customization by developing commonality indices, *Journal of Engineering Design*, 11(3), pp. 225–243.

Johnson, M. D. and Kirchain, R. (2010) Developing and Assessing Commonality Metrics for Product Families: A Process-Based Cost-Modeling Approach, *IEEE Transactions on Engineering Management*, 57(4), pp. 634–648.

Jonas, H., Gebhardt, N. and Krause, D. (2012) Towards a strategic development of modular product programs, In *Proceedings of 12th International Design Conference - DESIGN 2012*, Cavtat, Dubrovnik, Croatia, pp. 959–968.

Jordan, W. C. and Graves, S. C. (1995) Principles on the Benefits of Manufacturing Process Flexibility, *Management Science*, 41(4), pp. 577–594.

Jose, A. and Tollenaere, M. (2005) Modular and platform methods for product family design: literature analysis, *Journal of Intelligent Manufacturing*, 16(3), pp. 371–390.

Junge, M. (2005) *Controlling modularer Produktfamilien in der Automobilindustrie*, Wiesbaden, Deutscher Universitaetsverlag.

Kamrad, B., Schmidt, G. M. and Ulku, S. (2013) Analyzing Product Architecture Under Technological Change: Modular Upgradeability Tradeoffs, *IEEE Transactions on Engineering Management*, 60(2), pp. 289–300.

Kamrani, A. K. and Salhieh, S. M. (2002) *Product Design for Modularity*, 2nd ed, Boston, Springer.

Karandikar, H. and Nidamarthi, S. (2007) Implementing a platform strategy for a systems business via standardization, *Journal of Manufacturing Technology Management*, 18(3), pp. 267–280.

Karius, A. (2011) Fabrik des Jahres: Abschied von der Überkomplexität, *Produktion.de - Technik und Wirtschaft für die deutsche Industrie*, [online] Available from: http://www.produktion.de/produktivitaet/fabrik-des-jahres-schwerer-abschied-von-der-ueberkomplexitaet/ (Accessed 11 January 2015).

Kazi, S. (2015) Toshiba's Reference designs for Ara Module Developers, In *DevCon2 2015*.

Keese, D. A., Takawale, N. P., Seepersad, C. C. and Wood, K. L. (2006) An enhanced change modes and effects analysis (CMEA) tool for measuring product flexibility with applications to consumer products., In *Proceedings of ASME 2006 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference*, Philadelphia, Pennsylvania, USA.

Kehl, S., Stiefel, P. and Müller, J. P. (2015) Changes on changes: Towards an agent-based approach for managing complexiy in decentralized product development, In *Proceedings of the 20th International Conference on Engineering Design (ICED15)*, Milan, Italy.

Keller, D. (2015) Zetsche bleibt am Steuer - Vertrag des Daimler-Chefs soll bis Ende 2019 verlängert werden, *Der Teckbote, 2nd April 2015*, p. 5.

Khedekar, N. (2016) LG G5, Project Ara and others: The modular smartphone revolution may come sooner than expected, *Tech 2*, [online] Available from: http://tech.firstpost.com/news-analysis/lg-g5-project-ara-and-others-the-modular-smartphone-revolution-may-come-sooner-than-expected-303928.html (Accessed 29 March 2016).

Kidd, S. L. (1998) A systematic method for valuing a product platform strategy, Thesis (MSc), Massachusetts Institute of Technology.

Kim, K. and Chhajed, D. (2001) An experimental investigation of valuation change due to commonality in vertical product line extension, *Journal of Product Innovation Management*, 18(4), pp. 219–230.

Kimberley, W. (1999) Back to the Future, *Automotive Engineer*, pp. 62–64.

Kipp, T., Blees, C. and Krause, D. (2010) Anwendung einer integrierten Methode zur Entwicklung modularer Produktfamilien, In *Design for X - Beiträge zu 21. DfX-Symposium*, Hamburg, pp. 157–168.

Kipp, T. and Krause, D. (2008) Design for Variety - Efficient support for Design Engineers, In *Proceedings of the DESIGN 2008*, Dubrovnik, pp. 425–432.

Kissel, M., Bradford, N., Kreimeyer, M. and Lindemann, U. (2012) Product structure management as the backbone of engineering design: exploration of a reference model, In *Proceedings of International Design Conference - DESIGN 2012*, Cavtat, Dubrovnik, Croatia, pp. 1709–1718.

Kissel, M. P. (2014) Mustererkennung in komplexen Produktportfolios, Thesis (PhD), Munich, Technische Universitaet Muenchen.

Koehler, N., Naumann, T. and Vajna, S. (2014) Supporting the modelling of traceability information, In *Proceedings of 13th International Design Conference - DESIGN 2014*, Cavtat, Dubrovnik, Croatia, pp. 1811–1820.

Koeppen, B. (2008) Modularisierung komplexer Produkte anhand technischer und betriebswirtschaftlicher Komponentenkopplung, Thesis (PhD), Technische Universitaet Hamburg-Harburg.

Kohlhase, N. (1996) Strukturieren und Beurteilen von Baukastensystemen: Strategien, Methoden, Instrumente, Thesis (PhD), Duesseldorf, Technische Hochschule Darmstadt.

Kohlhase, N. and Birkofer, H. (1996) Development of modular structures: The prerequisite for successful modular products, *Journal of Engineering Design*, 7(3), pp. 279–291.

Koppenhagen, F. (2004) Systematische Ableitung modularer Produktarchitekturen : Komplexitaetsreduzierung in der Konzeptphase, Thesis (PhD), Technische Universitaet Hamburg-Harburg.

Kota, S., Sethuraman, K. and Miller, R. (2000) A Metric for Evaluating Design Communality in Product Families, *Journal of Mechanical Design*, pp. 403–410.

Koziolek, H., Domis, D., Goldschmidt, T. and Vorst, P. (2013) Measuring Architecture Sustainability, *IEEE Software*, 30(6), pp. 54–62.

KPMG International Cooperative (2013) *Global automotive retail market - From selling cars on the spot to centrally managing the retail grid*, [online] Available from: http://www.kpmg.com/Global/en/IssuesAndInsights/ArticlesPublications/Documents/global-automotive-retail-market-study-part1.pdf (Accessed 26 August 2014).

Kraus, P. K. (2005) Plattformstrategien - Realisierung einer varianz- und kostenoptimierten Wertschöpfung, Thesis (PhD), Erlangen-Nürnberg, University Erlangen-Nürnberg.

Krause, D., Beckmann, G., Eilmus, S., Gebhardt, N., Jonas, H. and Rettberg, R. (2014) Integrated Development of Modular Product Families: A Methods Toolkit, In *Advances in Product Family and Product Platform Design*, Simpson, T. W., Jiao, J. R., Siddique, Z., and Hölttä-Otto, K. (eds.), Springer New York, pp. 245–269, [online] Available from: http://link.springer.com/chapter/10.1007/978-1-4614-7937-6_10 (Accessed 26 December 2014).

Kreimeyer, M. (2012) A product model to support PLM-based variant planning and management, In *Proceedings of International Design Conference - DESIGN 2012*, Cavtat, Dubrovnik, Croatia, pp. 1741–1752.

Kreimeyer, M. (2014) Implementing architecture in industry, In *Proceedings of International Design Conference - DESIGN 2014*, Dubrovnik, Croatia, pp. 1967–1978.

Krepper, D. (2011) Der Schluessel zum Erfolg liegt im Baukasten, *Stuttgarter Zeitung*, Stuttgart, 12th February.

Krishnan, V. and Ulrich, K. (2001) Product Development Decisions: A Review of the Literature, *Management Science*, 47(1), pp. 1–21.

Kristjansson, A. H. (2005) Platform Assessment Matrix (PAMatrix): A Method to Support Decision Making on Product Platforms, Thesis (PhD), Norwegian University of Science and Technology, Faculty of Engineering Science and Technology.

Kristjansson, A. H. and Hildre, H. P. (2004a) A framework for evaluating platforms in product developing organizations, In *Proceedings of 7th Workshop on Product Structuring - Product Platform Development*, Gothenburg, Sweden.

Kristjansson, A. H. and Hildre, H. P. (2004b) PAMatrix: A method to assess platforms in product developing companies, In *Proceedings of NordDesign 2004 - Product Design in Changing Environment*, Tampere, Finland.

Kristjansson, A. H. and Hildre, H. P. (2004c) Platform strategy: a study of influencing factors, In *Proceedigns of NordDesign 2004*, Tampere, Finland.

Kristjansson, A. H., Jensen, T. and Hildre, H. P. (2004) The term platform in the context of product developing company, In *Proceedings of the 8th International Design Conference DESIGN 2004*, Dubrovnik, Croatia, pp. 325–330.

Kruse, M., Ripperda, S. and Krause, D. (2015) Platform concept development within the integrated PKT-Approach, In *Proceedings of the 20th International Conference on Engineering Design (ICED15)*, Milan, Italy.

Kuhlemann, M. (2006) Moderne Modularisierungstechniken und ihre Bedeutung fuer qualitativ hochwertige Software, Thesis (Diplom), Otto-von-Guericke-Universitaet Magdeburg.

Kurtadikar, R. M., Stone, R. B., Van Wie, M. and McAdams, D. A. (2004) A customer needs motivated conceptual design methodology for product portfolios, In *Proceedings of ASME 2004 International Design Engineering Technical Conferences, DETC'04*, Salt Lake City, UT.

Kurylko, D. T. (2014) BMW 7 series to launch new platform - 35up architecture to underpin rwd cars, *Automotive News*, [online] Available from: http://www.autonews.com/article/20140804/OEM04/308049972?template=printart (Accessed 27 March 2015).

Kurylko, D. T. (2003) Model T had many shades; black dried fastest, *Automotive News*, [online] Available from: http://www.autonews.com/article/20030616/SUB/306160713/model-t-had-many-shades;-black-dried-fastest (Accessed 11 May 2016).

Kusiak, A. (1995) *Engineering Design - Products, Processes, and Systems*, San Diego, Academic Press.

Kusiak, A. and Chow, W. S. (1987) An Efficient Cluster Identification Algorithm, *IEEE Transactions on Systems, Man and Cybernetics*, 17(4), pp. 696–699.

Kusiak, A. and Huang, C. C. (1996) Development of modular products, *IEEE Transactions on components, packaging and manufacturing technology*, 19(4), pp. 523–538.

Kvist, M. (2010) Product Family Assessment, Thesis (PhD), Lyngby, Technical University of Denmark.

LaLande, M. (2013) *Overcoming Barriers to a Successful Vehicle Modularity Strategy*, *SAE Technical Paper*, Warrendale, SAE International.

Lange, M. W. (1998) CONCAD Bridging: Managing the integration of technical solutions in a modularised product concept, In *2nd International Conference on Integrated Design and Manufacturing in Mechanical Engineering*, Compiegne, pp. 521–528.

Lanner, P. and Malmqvist, J. (1996) An approach towards considering technical and economic aspects in product architecture design, In *Proceedings of 2nd WDK Workshop on Product Structuring*, Delft University of Technology, Netherlands.

Lau, A. K. W. (2011) Critical success factors in managing modular production design: Six company case studies in Hong Kong, China, and Singapore, *Journal of Engineering and Technology Management*, 28(3), pp. 168–183.

Lau, A. K. W. (2009) Managing Modular Product Design: Critical Factors and a Managerial Guide, In *PICMET 2009 Proceedings*, Portland, Oregon, USA, pp. 2045–2057.

Lehnerd, A. P. (1987) Revitalizing the Manufacture and Design of Mature Global Products, In *Technology and Global Industry: Companies and Nations in the World Economy*, Guile, B. R. and Brooks, H. (eds.), Washington D.C., National Academy Press, pp. 49–64.

Lehtonen, T. (2007) Designing modular product architecture in the new product development, Thesis (PhD), Tampere University of Technology.

Lindemann, U. (2009) *Methodische Entwicklung technischer Produkte : Methoden flexibel und situationsgerecht anwenden*, 3rd ed, Heidelberg, Springer.

Lindemann, U. (2006) *Methodische Entwicklung technischer Produkte: Methoden flexibel und situationsgerecht anwenden*, 2nd ed, Berlin, Springer.

Lindemann, U., Maurer, M. and Braun, T. (2009) *Structural Complexity Management: An Approach for the Field of Product Design*, Berlin, Springer.

Liu, J.-G. (2006) Clustering algorithm of product architecture modularization based on design structure matrix, 华南理工大学学报 (自然科学版), 34(11), pp. 45–48.

Liu, J.-G. (2008) Clustering and evaluating of product architecture based on physical DSM and row-column transform, 系统工程与电子技术, 30(10), pp. 1904–1908.

Liu, Z., Wong, Y. S. and Lee, K. S. (2010) Modularity analysis and commonality design: A framework for the top-down platform and product family design, *International Journal of Production Research*, 48(12), pp. 3657–3680.

Lutz, R. A. (1998) *Guts: The Seven Laws of Business That Made Chrysler the World's Hottest Car Company*, New York, USA, John Wiley & Sons.

Mack, N., Woodsong, C., MacQueen, K. M., Guest, G. and Namey, E. (2005) *Qualitative Research Methods - A Data Collector's Field Guide*, Research Triangle Park, North Carolina, USA, Family Health International.

Mael, F. (2014) Why fewer options mean more sales, *Smart CEO*, [online] Available from: http://www.smartceo.com/fewer-options-mean-sales/ (Accessed 1 May 2014).

Malik, F. (2003) *Strategie des Managements komplexer Systeme*, Bern, Paul Haupt Verlag.

Marshall, R. and Leaney, P. G. (2002) Holonic Product Design: A process for modular product realization, *Journal of Engineering Design*, 13(4), pp. 293–303.

Marti, M. (2007) Complexity Management: Optimizing Product Architecture of Industrial Products, Thesis (PhD), St. Gallen, University of St. Gallen.

Martin, M. V. (1999) Design for Variety: A methodology for developing product platform architectures, Thesis (PhD), Stanford University.

Martin, M. V. and Ishii, K. (1996) Design for variety: A methodology for understanding the costs of product proliferation, In *Proceedings of the 1996 ASME Design Engineering Technical Conferences and Computers in Engineering Conference*, Irvine, California.

Martin, M. V. and Ishii, K. (2002) Design for variety: developing standardized and modularized product platform architectures, *Research in Engineering Design*, pp. 213–235.

Martin, M. V. and Ishii, K. (1997) Design for Variety: development of complexity indices and design charts, In *Proceedings of DETC '97*, Sacramento.

Mattson, C. A. and Magleby, S. P. (2001) The influence of product modularity during concept selection of consumer products, In *Proceedings of 13th International Conference on Design Theory and Methodology*, American Society of Mechanical Engineers, pp. 299–308.

Mayer de Ávila, R. and Borsato, M. (2014) Modularity Adoption in Product Development - A Case Study in the Brazilian Agricultural Machinery Industry, *SAE Int. J. Mater. Manf.*, pp. 122–128.

McAdams, D. A., Stone, R. B. and Wood, K. L. (1999) Functional Interdependence and Product Similarity Based on Customer Needs, *Research in Engineering Design*, 11(1), pp. 1–19.

McGrath, M. E. (1995) *Product Strategy for High-Technology Companies*, New York, USA, Irwin Professional Publishing.

McIntyre, A. (2008) *Participatory Action Research*, Thousand Oaks, California, United States, SAGE Publications, Inc.

Meehan, J. S., Duffy, A. H. B. and Whitfield, R. I. (2007) Supporting `Design for Re-use' with Modular Design, *Concurrent Engineering*, 15(2), pp. 141–155.

Menard, S. (2002) *Longitudinal Research*, Thousand Oaks, California, SAGE Publications Inc.

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

Mikkola, J. H. (2006) Capturing the Degree of Modularity Embedded in Product Architectures, *Journal of Product Innovation Management*, 23(2), pp. 128–146.

Mikkola, J. H. and Gassmann, O. (2003) Managing modularity of product architectures: toward an integrated theory, *IEEE Transactions on Engineering Management*, 50(2), pp. 204–218.

Miller, T. (2000) Modular Engineering - An approach to structuring business with coherent modular architectures of artefacts, activities, and knowledge, Thesis (PhD), Technical University of Denmark.

MIT (2011) MIT Open Course Ware - Chapter 9: Product architecture, Cambridge, Massachusetts.

Mitteldeutsche Zeitung (2014) Daimler und Renault-Nissan sparen durch Zusammenarbeit, *Mitteldeutsche Zeitung Web - Wirtschaft*, [online] Available from: http://www.mz-web.de/wirtschaft/daimler-und-renault-nissan-sparen-durch-zusammenarbeit,20642182,28638214.html (Accessed 5 October 2014).

Möller, K., Hülle, J. and Kahle, S. (2011) Kennzahlencockpits zur Steuerung und zum Monitoring der Standardisierung, *ZWF Zeitschrift für wirtschaftlichen Fabrikbetrieb*, pp. 741–745.

Montano, R. P. (2011) Platform project management: optimizing product development by actively managing commonality, Thesis (MSc), Massachusetts Institute of Technology.

Moon, S. K., Sim, J., Shu, J. and Simpson, T. W. (2007) Strategic Module Sharing for Customized Service Family Design using a Bayesian Game, In *Proceedings of IEEE International Conference on Service Operations, Logistics and Informatics SOLI 2007*, Philadelphia, pp. 1–6.

Mortensen, N. H., Harlou, U. and Haug, A. (2008) Improving decision making in the early phases of configuration projects, *International Journal of Industrial Engineering*, 15(2), pp. 185–194.

Muffatto, M. (1996) Evolution of product development in Japanese automobile firms, *International Journal of Vehicle Design*, 17(4), pp. 343–359.

Muffatto, M. (1999) Introducing a platform strategy in product development, *International Journal of Production Economics*, 60–61, pp. 145–153.

Muffatto, M. and Roveda, M. (2000) Developing product platforms: analysis of the development process, *Technovation*, 20(11), pp. 617–630.

Mühlbradt, T. and Mirwald, D. (1992) Mit Komplexitätsmanagement handlungsfähig bleiben, *IO Management Zeitschrift*, pp. 41–42.

Munk, L. (2011) Product Platform Performance: Achieving internal effects, Lyngby, Technical University of Denmark.

Myrodia, A. and Hvam, L. (2015) Identification of complexity cost factors in manufacturing companies, In *Proceedings of the 22nd EurOMA Conference*.

National Aeronautics and Space Administration (NASA) (2007) *Systems Engineering Handbook*, Washington D.C., USA, NASA Headquarters.

Nepal, B., Monplaisir, L., Singh, N. and Yaprak, A. (2008) Product modularization considering cost and manufacturability of modules, *International Journal of Industrial Engineering*, 15(2), pp. 132–142.

Newcomb, P. J., Bras, B. and Rosen, D. W. (1996) Implications of modularity on product design for the life cycle, In *Proceedings of the 1996 ASME Design Engineering Technical Conferences and Computers in Engineering Conference*, Irvine, California.

Nielsen, O. F. (2010) Continuous Platform Development - Synchronizing Platform and Product Development, Thesis (PhD), Lyngby, Technical University of Denmark.

Nilsson, P. (2010) Conceptual product development in small corporations, Thesis (PhD), Stockholm, Royal Institute of Technology.

Nilsson, P. and Erixon, G. (1998) The chart of Modular Function Deployment, In *Proceedings of the 4th WDK Workshop on Product Structuring*, Delft, pp. 147–153.

Nobelius, D. and Sundgren, N. (2002) Managerial issues in parts sharing among product development projects: a case study, *Journal of Engineering and Technology Management*, 19(1), pp. 59–73.

Oosterman, B. (2001) Improving Product Development Projects by Matching Product Architecture and Organization, Thesis (PhD), Rijksuniversiteit Groningen.

Orfi, N., Terpenny, J. and Sahin-Sariisik, A. (2011) Harnessing Product Complexity: Step 1- Establishing Product Complexity Dimensions and Indicators, *Engineering Economist*, 56(1), pp. 59–79.

Orfi, N., Terpenny, J. and Sahin-Sariisik, A. (2012) Harnessing Product Complexity: Step 2 - Measuring and Evaluating Complexity Levels, *Engineering Economist*, 57(3), pp. 178–191.

Otto, K. N. and Holtta, K. (2004) A multi-criteria framework for screening preliminary product platform concepts, In *Proceedings of DETC'04*, Salt Lake City.

Oxford Dictionaries Online (2011) Definition for heuristic, [online] Available from: http://oxforddictionaries.com/definition/heuristic?q=heuristic (Accessed 9 January 2012).

Oxford Dictionaries Online (2016) Stable - definition of stable in English from the Oxford Online Dictionary, [online] Available from: http://www.oxforddictionaries.com/definition/english/stable?searchDictCode=all (Accessed 29 February 2016).

Oxford Dictionaries Online (2015) Variant - definition of variant in English from the Oxford dictionary, [online] Available from: http://www.oxforddictionaries.com/definition/english/variant?searchDictCode=all (Accessed 10 February 2015).

Pahl, G., Beitz, W., Feldhusen, J. and Grote, K.-H. (2007) *Engineering design : a systematic approach*, 3., London, Springer.

Pakkanen, J., Juuti, T. and Lehtonen, T. (2015) Brownfield process for the rationalisation of existing product variety towards a modular product family, In *Proceedings of the 20th International Conference on Engineering Design (ICED15)*, Milan, Italy.

Pander, J. (2012) Neues Konstruktionssystem bei VW: Gleich ist geil, *Spiegel Online*, [online] Available from: http://www.spiegel.de/auto/aktuell/0,1518,814246,00.html (Accessed 23 April 2012).

Pander, J. (2014a) Qualitätsprobleme - Neuer Rekord bei Rückrufen, *Spiegel Online*, [online] Available from: http://www.spiegel.de/auto/aktuell/rueckrufe-autobauer-in-

2014-mit-massiven-qualitaetsproblemen-a-965963-druck.html (Accessed 1 May 2014).

Pander, J. (2014b) Rekordjahr für Autohersteller Skoda: Die Millionen-Marke, *Spiegel Online*, [online] Available from: http://www.spiegel.de/auto/aktuell/skoda-mit-rekordjahr-dilemma-im-vw-konzern-a-1008541-druck.html (Accessed 30 December 2014).

Pandremenos, J., Paralikas, J., Salonitis, K. and Chryssolouris, G. (2009) Modularity concepts for the automotive industry - A critical review, *CIRP Journal of Manufacturing Science and Technology*, *Design Synthesis*, 1(3), pp. 148–152.

Park, J. and Simpson, T. W. (2005) Development of a production cost estimation framework to support product family design, *International Journal of Production Research*, 43(4), pp. 731–772.

Park, J. and Simpson, T. W. (2008) Toward an activity-based costing system for product families and product platforms in the early stages of development., *International Journal of Production Research*, 46(1), pp. 99–130.

Park, R. E., Goethert, W. B. and Florac, W. A. (1996) *Goal-Driven Software Measurement - A Guidebook*, Pittsburgh, Pennsylvania, Software Engineering Institute, Carnegie Mellon University.

Parslov, J. F. and Mortensen, N. H. (2015) Interface definitions in literature: A reality check, *Concurrent Engineering*, 23(3), pp. 183–198.

Paul Byrd Law Firm (2013) Risks - Stryker Hip Recall, [online] Available from: http://byrdlaw.net/stryker-hip-recall/stryker-hip-cases/risks-stryker-hip-recall/ (Accessed 4 November 2014).

Pedersen, R. (2010) Product Platform Modeling: Contributions to the discipline of visual product platform modelling, Lyngby, Technical University of Denmark.

Persson, M. and Åhlström, P. (2006) Managerial issues in modularising complex products, *Technovation*, 26(11), pp. 1201–1209.

Persson, M. and Ahlstrom, P. (2013) Product modularisation and organisational coordination, *International Journal of Automotive Technology and Management*, 13(1), pp. 55–74.

Pettigrew, A. M. (1990) Longitudinal Field Research on Change - Theory and Practice, *Organization Science*, 1(3), pp. 267–292.

Phonebloks (2016) Phonebloks – a project by dave hakkens, [online] Available from: https://phonebloks.com/ (Accessed 29 March 2016).

Piller, F. (2001) *Mass Customization - Ein wettbewerbsstrategisches Konzept im Informationszeitalter*, 2nd ed, Wiesbaden, Gabler Verlag.

Piller, F. and Waringer, D. (1999) *Modularisierung in der Automobilindustrie - neue Formen und Prinzipien*, Aachen, Shaker.

Pimmler, T. U. (1994) A development methodology for product decomposition and integration, Thesis (M.S.), Massachusetts Institute of Technology.

Pimmler, T. U. and Eppinger, S. D. (1994) Integration analysis of product decompositions, In *ASME Design Theory and Methodology Conference*, Minneapolis.

Pine, B. J. (1992) *Mass Customization: The New Frontier in Business Competition*, 1st ed, Harvard Business Press.

Pirmoradi, Z. and Wang, G. G. (2011) Recent advancements in product family design and platform-based product development: A literature review, In *Proceedings of the ASME Desing Engineering Technical Conferences - DETC2011*, Washington D.C., ASME, New York.

Plaikner, S., Kissel, M., Kreimeyer, M. and Lindemann, U. (2012) An Architecture Framework for Multi-Product Portfolio Management in the Commercial Vehicle Industry, In *Proceedings of the 9th NordDesign Conference 2012*, Aarlborg University, Denmark.

Ponn, J. (2015) Portfolio management for electric drives in powertools at Hilti: Challenges and solution approaches, In *Proceedings of the 20th International Conference on Engineering Design (ICED15)*, Milan, Italy.

Ponn, J. and Lindemann, U. (2008) *Konzeptentwicklung und Gestaltung technischer Produkte: Optimierte Produkte - systematisch von Anforderungen zu Konzepten*, Berlin, Springer.

Porter, M. E. (1992) *Wettbewerbsstrategie: Methoden zur Analyse von Branchen und Konkurrenten*, 7th ed, Frankfurt a. M., Campus-Verl.

Pötsch, H. D. (2011) Volkswagen - On the Road of Success, Frankfurt a. M., [online] Available from: http://www.volkswagenag.com/content/vwcorp/info_center/de/talks_and_presentations/2011/03/Presentation_Frankfurt_Poetsch.bin.html/binarystorageitem/file/Presentation+Frankfurt+14-03-2011+Top+Copy.pdf (Accessed 26 March 2015).

PSA Peugeot Citroen (2013) EMP2 - the new Efficient Modular Platform by PSA Peugeot Citroën, *PSA PEUGEOT CITROËN*, [online] Available from: http://www.psa-peugeot-citroen.com/en/content/emp2-new-efficient-modular-platform-psa-peugeot-citroen-0 (Accessed 4 September 2013).

Rajan, P. K. P., Wie, M. J. V., Campbell, M. I., Otto, K. N. and Wood, K. L. (2003) Design for Flexibility - Measures and Guidelines, In *Proceedings of International Conference on Engineering Design, ICED2003*, Stockholm, Sweden.

Rajan, P. K. P., Wie, M. J. V., Campbell, M. I., Wood, K. L. and Otto, K. N. (2005) An Empirical Foundation for Product Flexibility, *Design Studies*, 26(4), pp. 405–438.

Rajan, P. K. P., Wie, M. J. V., Wood, K. L., Otto, K. N. and Campbell, M. I. (2004) Empirical Study on Product Flexibility, In *Proceedings of the ASME Design Engineering Technical Conference, DETC2004*, Salt Lake City, Utah, pp. 109–124.

Rapp, T. (1999) *Produktstrukturierung : Komplexitaetsmanagement durch modulare Produktstrukturen und -plattformen*, Wiesbaden, Deutscher Universitaetsverlag.

Rathnow, P. I. (1993) *Integriertes Variantenmanagement: Bestimmung, Realisierung, und Sicherung der optimalen Produktvielfalt*, Goettingen, Vandenhoeck & Ruprecht.

Rauchenberger, J. (2010) Reifegradmodelle als Ordnungsrahmen zur systematischen Prozessverbesserung für mechatronische Entwicklungsprozesse, Aachen, Apprimus-Verl.

Rechtin, E. and Maier, M. (2000) *The Art of Systems Architecting*, 2nd ed, CRC Press Inc.

Reddi, K. R. and Moon, Y. B. (2013) Modelling engineering change management in a new product development supply chain, *International Journal of Production Research*, 51(17), pp. 5271–5291.

Rehesaar, H. (2011) Capability Assessment for Introducing Component Reuse, In *Top Productivity through Software Reuse*, *Lecture Notes in Computer Science*, Schmid, K. (ed.), Springer Berlin Heidelberg, pp. 87–101.

Rennekamp, M. (2013) *Methode zur Bewertung des Komplexitätsgrades von Unternehmen*, *Ergebnisse aus der Produktionstechnik*, Aachen, Apprimus-Verlag.

Renner, I. (2007) Methodische Unterstützung funktionsorientierter Baukastenentwicklung am Beispiel Automobil, Thesis (PhD), Munich, Technische Universitaet Muenchen.

Riepe, B. (2003) *Integrierte Produktstrukturmodellierung in den frühen Phasen der Produktentstehung: Eine Methode zur Modularisierung variantenreicher mechatronischer Produkte.*, Books on Demand Gmbh.

Ripperda, S. and Krause, D. (2015) Cost prognosis of modular product structure concepts, In *Proceedings of the 20th International Conference on Engineering Design (ICED15)*, Milan, Italy.

Ripperda, S. and Krause, D. (2013) Komplexitätskosteneffekte modularer Produktfamilien, In *Design for X - Beiträge zum 24. DfX-Symposium*, Hamburg, pp. 13–24.

Ripperda, S. and Krause, D. (2014) Towards Complexity Cost Management within Approaches for Developing Modular Product Families, In *Proceedings of the International Conference on Advanced Design Research and Education - ICADRE14*, Singapore, pp. 15–19.

Robertson, D. and Ulrich, K. (1998) Planning for Product Platforms, *Sloan Management Review*, 39(4), pp. 19–31.

Robson, C. (1993) *Real World Research - A Resource for Social Scientists and Practicioner-Researchers*, Oxford, UK, Blackwell Publishers.

Roland Berger Strategy Consultants (2012) Modular Products - How to leverage modular product kits for growth and globalization, Study - Long Version, Study Presentation, Stuttgart/Munich, [online] Available from: http://www.rolandberger.com/media/publications/2012-04-10-rbsc-pub-Modular_Products.html (Accessed 2 June 2014).

Romanowski, C. J. and Nagi, R. (2005) On comparing bills of materials: A similarity/distance measure for unordered trees, In *IEEE Transactions on Systems, Man, and Cybernetics, Part A: Systems and Humans*, pp. 249–260.

Roozenburg, N. F. M. and Eekels, J. (1995) *Product Design: Fundamentals and Methods*, New York, Wiley.

Rothwell, R. and Gardiner, P. (1983) The role of design in product and process change, *Design Studies*, 4(3), pp. 161–169.

Saldana, J. (2013) *The coding manual for qualitative researchers*, 2nd ed, London, England, SAGE Publications Ltd.

Salhieh, S. M. and Kamrani, A. K. (1999) Macro level product development using design for modularity, *Robotics and Computer-Integrated Manufacturing*, 15(4), pp. 319–329.

Salvador, F. (2007) Toward a Product System Modularity Construct: Literature Review and Reconceptualization, *IEEE Transactions on Engineering Management*, 54(2), pp. 219–240.

Sanchez, R. and Mahoney, J. T. (1996) Modularity, flexibility, and knowledge management in product and organization design, *Strategic Management Journal*, 17(Winter Special Issue), pp. 63–76.

Sand, J., Gu, P. and Watson, G. (2002) HOME: House Of Modular Enhancement - A Tool for Modular Product Redesign, *Concurrent Engineering*, 10(2), pp. 153–164.

Sand, J., Gu, P. and Watson, G. (2001) Home: House of modular enhancement for product redesign for modularization, In *Proceedings of the ASME Design Engineering Technical Conference and Computers and Information in Engineering Conference*, *Proceedings of the ASME Design Engineering Technical Conference*, Pittsburgh, PA, United states, American Society of Mechanical Engineers, pp. 515–526.

Sanderson, S. and Uzumeri, M. (1995) Managing product families: The case of the Sony Walkman, *Research Policy*, 24(5), pp. 761–782.

Scania (2000) Prestigious engineering award to Scania, *Scania*, [online] Available from: http://www.scania.com/media/pressreleases/toolman3/2000/q2/20000413120000en64420.aspx (Accessed 25 May 2011).

Scania (2009) Success through modularisation, *Scania Value*, pp. 6–7.

Scania (2008) Urban transport solutions for maximum uptime, [online] Available from: http://www.scania.com/Images/P08405EN%20Urban%20transport%20solutions_tcm10-201849_75145.pdf (Accessed 25 May 2011).

Scania Group (2011) Easier customisation with Scania's modular system, *Scania Group*, [online] Available from: http://www.scania.com/media/feature-stories/general/module-system.aspx (Accessed 25 May 2011).

Schaeppi, B., Andreasen, M. M., Kirchgeorg, M. and Radermacher, F.-J. (2005) *Handbuch Produktentwicklung*, München, Wien, Carl Hanser Verlag.

Schifferes, S. (2007) Globalising the car industry, *BBC News - Business*, [online] Available from: http://news.bbc.co.uk/2/hi/business/6346325.stm (Accessed 26 August 2014).

Schuh, G. (1988) Gestaltung und Bewertung von Produktvarianten : ein Beitrag zur systematischen Planung von Serienprodukten, Thesis (PhD), Technische Hoschschule Aachen.

Schuh, G. (2005) *Produktkomplexitaet managen : Strategien, Methoden, Tools.*, 2nd ed, Munich, Hanser.

Schuh, G., Aleksic, S. and Arnoscht, J. (2013) Module based release planning for technical changes, In *2013 Proceedings of PICMET'13: Technology Management for Emerging Technologies*, pp. 1604–1617.

Schuh, G., Aleksic, S. and Rudolf, S. (2015) Module-based release management for technical changes, *Progress in Systems Engineering - Advances in Intelligent Systems and Computing Volume*, 330, pp. 293–298.

Schuh, G., Arnoscht, J. and Nussbaum, C. (2007) Produktarchitekturen richtig gestalten - Ein Weg zum variantenoptimierten Produktprogramm, *Industrie Management*, pp. 29–32.

Schuh, G. and Jonas, I. (1997) Variantenreduzierung im Verbund - Praktikable Methode zum Variantenmanagement, In *Ein Leitfaden zur Beherrschung der Variantenvielfalt*, Düsseldorf, ViA Verbundinitiative Automobil Nordrhein-Westfalen.

Schuh, G., Lenders, M. and Bender, D. (2009) *Szenariorobuste Produktarchitekturen*, *HNI-Verlagsschriftenreihe 265*, Paderborn, Heinz Nixdorf Institut, pp. 99–119.

Schuh, G., Rudolf, S. and Vogels, T. (2014) Performance measurement of modular product platforms, In *Proceedings of the 47th CIRP Conference on Manufacturing Systems - Variety Management in Manufacturing*, pp. 266–271.

Schuh, G., Schultze, W., Schiffer, M., Rieger, A., Rudolf, S. and Lehbrink, H. (2014) Scenario-based determination of product feature uncertainties for robust product architectures, *Production Engineering*, 8(3), pp. 383–395.

Schuh, G. and Schwenk, U. (2001) *Produktkomplexität managen - Strategien, Methoden, Tools*, München, Carl Hanser Verlag.

Schuh, G., Sommer, M. and Rudolf, S. (2015) Organizational implementation of product architecture development, In *2015 Portland International Conference on Management of Engineering and Technology (PICMET)*, Portland, Oregon, USA, pp. 1769–1776.

Schuh & Co. (2015) Complexity Manager, *Complexity Manager*, [online] Available from: http://www.complexitymanager.de/en/ (Accessed 8 January 2016).

Schwartz, B. (2005) *The Paradox of Choice: Why More Is Less*, New York, NY, USA, Harper-Collins.

Scoltock, J. (2014) Editor's Comment, *Automotive Engineer*.

Shamsuzzoha, A. H. M. (2011) Modular product architecture for productivity enhancement, *Business Process Management Journal*, 17(1), pp. 21–41.

Shan, Q. and Chen, Y. (2009) Research on Numerical Module Identification Approaches, In *Eighth IEEE/ACIS International Conference on Computer and Information Science, 2009. ICIS 2009.*, Shanghai, pp. 672–676.

Sharafi, A., Wolfenstetter, T., Wolf, P. and Krcmar, H. (2010) Comparing product development models to identify process coverage and current gaps: A literature review, In *Proceedings of 2010 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM)*, Macao, pp. 1732–1736.

Shibata, T. and Kodama, M. (2013) Crafting a product platform strategy: Change in strategy of Mabuchi Motors, a leading DC motor manufacturer, In *2013 Proceedings of PICMET'13: Technology Management for Emerging Technologies*, pp. 1625–1633.

Siddique, Z. and Repphun, B. (2001) Estimating Cost Savings when Implementing a Product Platform Approach, *Concurrent Engineering*, 9(4), pp. 285–294.

Siddique, Z. and Rosen, D. W. (1999) Product platform design: A graph grammar approach, In *Proceedings of 1999 ASME Design Engineering Technical Conferences*, Las Vegas, Nevada.

Siddique, Z., Rosen, D. W. and Wang, N. (1998) On the applicability of product variety design concepts to automotive platform commonality, In *Proceedings of 1998 ASME Design Engineering Technical Conferences*, Atlanta, Georgia.

Simpson, T. W. (2004) Product platform design and customization: Status and promise, *Artificial Intelligence for Engineering Design*, 18(1), pp. 3–20.

Simpson, T. W., Bobuk, A., Slingerland, L. A., Brennan, S., Logan, D. and Reichard, K. (2012) From user requirements to commonality specifications: an integrated approach to product family design, *Research in Engineering Design*, 23(2), pp. 141–153.

Simpson, T. W., Jiao, J., Siddique, Z. and Holtta-Otto, K. (eds.) (2014) *Advances in Product Family and Product Platform Design - Methods & Applications*, New York, Springer.

Simpson, T. W., Maier, J. and Mistree, F. (2001) Product platform design: method and application, *Research in Engineering Design*, 13(1), pp. 2–22.

Simpson, T. W., Siddique, Z. and Jiao, J. (2005a) Platform-Based Product Family Development - An Introduction and Overview, In *Product Platform and Product Family Design - Methods and Application*, Simpson, T. W., Siddique, Z., and Jiao, J. (eds.), New York, USA, Springer US, pp. 1–15.

Simpson, T. W., Siddique, Z. and Jiao, J. (eds.) (2005b) *Product Platform and Product Family Design: Methods and Applications*, New York, USA, Springer.

Simpson, T. W., Tucker, M., De Weck, O. L., Holtta-Otto, K., Kokkolaras, M. and Shooter, S. B. (2006) Platform-based design and development: current trends and needs in industry, In *Proceedings of IDETC/CIE 2006*, Philadelphia.

Sinigalias, P. C. and Dentsoras, A. (2015) Index-based metrics for the evaluation of effects of custom parts on the standardization of mechanical systems, In *Proceedings of the 20th International Conference on Engineering Design (ICED15)*, Milan, Italy.

Skodaportal (2014) Der neue Fabia nutzt teilweise die MQB-Plattform, *Fabia*, [online] Available from: http://www.skoda-portal.de/index.php/2014/08/der-neue-fabia-baut-teilweise-auf-mqb/ (Accessed 28 October 2014).

Skold, M. and Karlsson, C. (2007) Multibranded platform development: A corporate strategy with multimanagerial challenges, *Journal of Product Innovation Management*, 24(6), pp. 554–566.

Smith, P. G. and Reinertsen, D. (1991) *Developing products in half the time*, New York, Van Nostrand Reinhold.

Software Engineering Institute (SEI) (2010) *CMMI® for Development, Version 1.3 (CMMI-DEV, V1.3) - Improving processes for developing better products and services*, Technical Report, Carnegie Mellon University.

Sörensen, D. (2006) The automotive development process: a real options analysis, Thesis (PhD), Wiesbaden, Deutscher Universitäts-Verlag.

Sosa, M., E., Agrawal, A., Eppinger, S. D. and Rowles, C. M. (2005) A network approach to define modularity of product components, In *Proceedings of IDETC/CIE 2005, ASME 2005 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference*, Long Beach, California, USA, p. 2005 September 24-28.

Sosa, M., E., Eppinger, S. D. and Rowles, C. M. (2003) Identifying Modular and Integrative Systems and Their Impact on Design Team Interactions, *Journal of Mechanical Design*, 125(2), p. 240.

Spiegel Online (2015) Aufholjagd auf dem Weltmarkt - Fünf Macken bei Volkswagen, *Spiegel Online - Wirtschaft*, [online] Available from: http://www.spiegel.de/wirtschaft/unternehmen/volkswagen-wird-der-konzern-2015-zum-weltgroessten-autobauer-a-1020524.html (Accessed 11 April 2015).

Stake, R. B. (2000) On conceptual development of modular products, Thesis (PhD), Stockholm, Royal Institute of Technology.

Steinbeis (2011) Kundenorientierung trotz Standardisierung, [online] Available from: http://www.stw.de/publikationen/transfermagazin/transfer-012011/kundenorientierung-trotz-standardisierung.html (Accessed 16 August 2012).

Steva, E. D., Rice, E. N., Marion, T. J., Simpson, T. W. and Stone, R. B. (2006) Two Methodologies for Identifying Product Platform Elements Within an Existing Set of Products, In *Proceedings of the ASME 2006 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference - DETC2006*, Philadelphia, Pennsylvania, USA, pp. 811–821.

Steward, D. (1981) Design structure system: A method for managing the design of complex systems, *IEEE transactions on engineering management*, 28(3), pp. 71–74.

Stiftung Warentest (2015) Studie zur Obsoleszenz - Schnellerer Austausch, *Test*, p. 33.

Stockmar, J. (2004) Variantenmanagement: Varianten vermeiden, beherrschen, reduzieren, finanzieren. Ergebnisse einer Umfrage 2004, In *6. Internationales Automobil-Forum*, Graz, Austria, Moderne Industrie (Landsberg).

Stone, R. B. (1997) Towards a theory of modular design, Thesis (PhD), Austin, University of Texas.

Stone, R. B., Wood, K. L. and Crawford, R. H. (2000) A heuristic method for identifying modules for product architectures, *Design Studies*, 21(1), pp. 5–31.

Stone, R. B., Wood, K. L. and Crawford, R. H. (1999) Product architecture development with quantitative functional models, In *Proceedings of ASME Design Engineering Technical Conferences, DETC99*, Las Vegas, Nevada.

Suh, N. P. (2001) *Axiomatic Design - Advances and Applications*, New York, Oxford University Press.

Suh, N. P. (1990) *The Principles of Design*, New York, Oxford University Press.

Sundgren, N. (1999) Introducing interface management in new product family development, *Journal of Product Innovation Management*, 16(1), pp. 40–51.

Sustainable Energy Authority Ireland (2013) *Efficient Home Heating - Your Options*, Wilton Place, Dublin, Ireland.

The Economic Times (2013) Code X4: Tata to launch hatchbacks, sedans and MPVs to regain lost glory, *The Economic Times - Automobiles*, [online] Available from: http://articles.economictimes.indiatimes.com/2013-09-02/news/41688847_1_tata-nano-tata-motors-nano-diesel (Accessed 4 September 2013).

The Huffington Post (2013) Hydrogen Fuel Cell Cars To Be Researched By Ford, Daimler And Nissan Partnership, *Huff Post Green*, [online] Available from: http://www.huffingtonpost.com/2013/01/28/hydrogen-fuel-cell-cars-vehicles_n_2566285.html (Accessed 7 October 2014).

Thevenot, H. J. and Simpson, T. W. (2007) A comprehensive metric for evaluating component commonality in a product family, *Journal of Engineering Design*, 18(6), pp. 577–598.

Thevenot, H. J. and Simpson, T. W. (2006) Commonality indices for product family design: a detailed comparison, *Journal of Engineering Design*, 17(2), pp. 99–119.

Thomke, S. (1997) The role of flexibility in the development of new products: An empirical study, *Research Policy*, 26(1), pp. 105–119.

Thomke, S. and Reinertsen, D. (1998) Agile Product Development: managing development flexibility in uncertain environments, *California Management Review*, 41(1), pp. 8–30.

Thonemann, U. W. and Brandeau, M. L. (2000) Optimal Commonality in Component Design, *Operations Research*, 48(1), pp. 1–19.

Thumm, B. R. and Göhlich, D. (2015) Product architecture design methodology for developing standardized modules, In *Proceedings of International Conference on Engineering Design 2015, ICED 2015*, Politecnico di Milano, Italy.

Thyssen, J., Israelsen, P. and Jørgensen, B. (2006) Activity-based costing as a method for assessing the economics of modularization - A case study and beyond, *International Journal of Production Economics*, 103(1), pp. 252–270.

Tiwana, A., Konsynski, B. and Bush, A. A. (2010) Research Commentary - Platform Evolution: Coevolution of Platform Architecture, Governance, and Environmental Dynamics, *Information Systems Research*, 21(4), pp. 675–687.

Toyota Motor Company (2014) How many parts is each car made of?, *Children's Question Room*, [online] Available from: http://www.toyota.co.jp/en/kids/faq/d/01/04/ (Accessed 30 September 2014).

Tripathi, A. K. and Ratneshwer (2009) Some Observations on a Maturity Model for CBSE, In *14th IEEE International Conference on Engineering of Complex Computer Systems*, pp. 273–281.

Tseng, M. M. and Jiao, J. (1997) A module identification approach to the electrical design of electronic products by clustering analysis of the design matrix, *Computers & Industrial Engineering*, 33(1–2), pp. 229–233.

Ullman, D. G. (2003) *The Mechanical Design Process*, 3rd ed, New York, McGraw-Hill.

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

Ulrich, K. and Eppinger, S. D. (2012) *Product design and development*, 5th ed, New York, USA, McGraw-Hill.

Ulrich, K. and Eppinger, S. D. (2004) *Product design and development*, 3rd ed, Boston, McGraw-Hill.

Ulrich, K. and Eppinger, S. D. (2008) *Product design and development*, 4th ed, Boston, McGraw-Hill.

Ulrich, K. and Tung, K. (1991) Fundamentals of product modularity, In *Winter Annual Meeting of the American Society of Mechanical Engineers, Issues in Design/Manufacture Integration - 1991*, Atlanta, GA, USA, ASME, pp. 73–79.

Van den Linden, F. (2005) Family Evaluation Framework, overview & introduction, In *Eureka 2023 Programme, ITEA project ip02009*.

Van Eikema Hommes, Q. D. (2008) Comparison and application of metrics that define the components modularity in complex products, In *Proceedings of 2008 ASME International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, Brooklyn, New York, USA, ASME, pp. 287–296.

Van Wie, M. J., Rajan, P., Campbell, M. I., Stone, R. B. and Wood, K. L. (2003) Representing product architecture, In *2003 ASME Design Engineering Technical Conference and Computers and Information in Engineering Conference, September 2, 2003 - September 6, 2003*, Chicago, IL, United states, American Society of Mechanical Engineers, pp. 731–746.

Verband der Automobilindustrie (VDA) (2010) *Qualitätsmanagement in der Automobilindustrie Band 6 - Prozessaudit (Teil 3) - Produktentstehungsprozess / Serienproduktion, Dienstleistungsentstehungsprozess / Erbringung der Dienstleistung*, 2nd ed, Frankfurt a. M., Heinrich Druck + Medien GmbH.

Verband der Automobilindustrie (VDA) (2003) *Qualitätsmanagement in der Automobilindustrie Band 18 - Vom Qualitätsmanagement der deutschen Automobilindustrie zu Automotive Excellence (AE)*, 2nd ed, Frankfurt a. M., Heinrich Druck + Medien GmbH.

Verein Deutscher Ingenieure (2004a) *VDI-Richtlinie 2206: Design Methodology for mechatronic systems*, Berlin, Beuth.

Verein Deutscher Ingenieure (1993) *VDI-Richtlinie 2221: Methodik zum Entwickeln und Konstruieren technischer Systeme und Produkte*, Berlin, Beuth.

Verein Deutscher Ingenieure (2004b) *VDI-Richtlinie 2223: Methodisches Entwerfen technischer Produkte*, Berlin, Beuth.

Vietor, T. and Hoffmann, C.-A. (2014) Es liegt im Auge des Betrachters… - Organisatorische Anforderungen an die Steuerung von modularen Baukästen in der Automobilindustrie, *Mitteilungen der WiGeP - Wissenschaftliche Gesellschaft für Produktentwicklung*, pp. 4–6.

Vijayenthiran, V. (2015) BMW To Reduce Number Of Platforms, *Motor Authority*, [online] Available from: http://www.motorauthority.com/news/1090151_bmw-to-reduce-number-of-platforms (Accessed 27 March 2015).

Volkswagen (2012) Der Baukasten für die Zukunft, *Autogramm - Die Zeitung für Mitarbeiterinnen und Mitarbeiter der Marke Volkswagen, Ausgabe 1-2/2012*, [online] Available from: http://autogramm.volkswagen.de/01-02_12/standorte/standorte_01.html (Accessed 11 January 2015).

Volkswagen (2014) Volkswagen Group - The Group, [online] Available from: http://www.volkswagenag.com/content/vwcorp/content/en/the_group.html (Accessed 27 March 2015).

Volkswagen Group (2012) Flexible Fabrik - Die neuen Produktionsstandards, *VIAVISION - Volkswagen Group - Nachrichten aus de mobilen Zukunft*, p. 3.

Volkswagen Group (2015) Modular Tookit Strategy, [online] Available from: http://www.volkswagenag.com/content/vwcorp/content/de/investor_relations/Warum_Volkswagen/MQB.html (Accessed 11 January 2015).

Wacker, J. G. and Treleven, M. (1986) Component part standardization: An analysis of commonality sources and indices, *Journal of Operations Management*, 6(2), pp. 219–244.

Whitfield, R. I., Smith, J. S. and Duffy, A. H. (2002) Identifying component modules, In *Proceedings of Seventh International Conference on Artificial Intelligence in Design (AID'02), July 15-17*, Cambridge, UK, pp. 571–592.

Whitney, D. E. (1993) Nippondenso Co. Ltd: A case study of strategic product design, *Research in Engineering Design*, 5(1), pp. 1–20.

Whitney, D. E. (2004) Product Architecture, [online] Available from: http://www.bb.ustc.edu.cn/ocw/NR/rdonlyres/Mechanical-Engineering/2-875Mechanical-Assembly-and-Its-Role-in-Product-DevelopmentFall2002/5E46ECDB-1315-4798-8BED-F56820807F19/0/Class15ProductArchitecture.pdf (Accessed 6 November 2011).

Whitney, D. E., Dong, Q., Judson, J. and Mascoli, G. (1999) Introducing knowledge-based engineering into an interconnected product development process, In *ASME 1999 International Design Engineering Technical Conference, iDETC1999, September 12-15*, Las Vegas, Nevada.

Wijnstra, J. G. (2004) Variation Mechanisms and Multi-view Architecting in Platform-based Product Family Development, Thesis (PhD), Groningen, Rijksuniversiteit Groningen.

Wildemann, H. (2007) *Produktordnungssysteme: Leitfaden zur Standardisierung und Individualisierung des Produktprogramms durch intelligente Plattformstrategien*, 6th ed, München, TCW, Transfer-Centrum-Verl.

Wildemann, H. (2005) *Variantenmanagement*, 14th ed, Munich, TCW.

Wildemann, H. (1991) Zeit als Wettbewerbsfaktor durch Motivation und Qualifikation, In *Wettbewerbsfaktor Zeit in Produktionsunternehmen*, *Referate des Münchner Kolloquiums '91, Institut für Werkzeugmaschinen und Betriebswirtschaften, Technische Universität München*, Milberg, J. (ed.), Berlin, Heidelberg, Springer, pp. 339–355.

WORLDCARFANS (2014) New Skoda Fabia half based on MQB platform – report, [online] Available from: http://www.worldcarfans.com/114082079875/new-skoda-fabia-half-based-on-mqb-platform--report (Accessed 28 October 2014).

Xu, X., Li, C., Yan, J. and Chen, Y. (2006) An analytical method based on design structure matrix for modular identification, In *7th International Conference on Computer-Aided Industrial Design and Conceptual Design, CAIDC*, Hangzhou, China, Inst. of Elec. and Elec. Eng. Computer Society.

Yeo, K. T. and Ren, Y. (2009) Risk management capability maturity model for complex product systems (CoPS) projects, *Systems Engineering*, 12(4), pp. 275–294.

Yigit, A. S., Galip, A. and Allahverdi, A. (2002) Optimizing modular product design for reconfigurable manufacturing, *Journal of intelligent manufacturing*, 13(4), pp. 309–316.

Yin, R. K. (1994) *Case study research: design and methods*, California, Sage Publications.

Yu, S., Yang, Q., Tao, J., Tian, X. and Yin, F. (2011) Product modular design incorporating life cycle issues - Group Genetic Algorithm (GGA) based method, *Journal of Cleaner Production*, 19(9–10), pp. 1016–1032.

Yu, T.-L., Yassine, A. A. and Goldberg, D. E. (2003) A Genetic Algorithm for Developing Modular Product Architectures, *ASME Conference Proceedings*, 2003(37017b), pp. 515–524.

Yu, T.-L., Yassine, A. A. and Goldberg, D. E. (2007) An information theoretic method for developing modular architectures using genetic algorithms, *Research in Engineering Design*, 18(2), pp. 91–109.

Yu, T.-L., Yassine, A. and Goldberg, D. E. (2005) An information theoretic method for developing modular architectures using genetic algorithms, In *Department of General Engineering, Illinois Genetic Algorithms laboratory IlliGAL*, University of Illinois, Urbana-Champlain.

Zacharias, N. A. and Yassine, A. A. (2008) Optimal platform investment for product family design, *Journal of Intelligent Manufacturing*, 19(2), pp. 131–148.

Zamirowski, E. J. and Otto, K. N. (1999) Identifying product portfolio architecture modularity using function and variety heuristics, In *Proceedings of the 11th International Conference on Design Theory and Methodology*, Las Vegas, Nevada.

ZF Corporate Communications (2010) *Excellent Production Support*, Press Release, Friedrichshafen, Germany, ZF Friedrichshafen AG.

ZF Friedrichshafen AG (2012) Ein variabler Baukasten für zukünftige Technologien, [online] Available from: http://www.zf.com/corporate/de/products/innovations/8hp_automatic_transmissions/variable_construction/variable_construction.html (Accessed 16 August 2012).

Zoltan, G. (2014) UX-Myths - Myth #12: More choices and features result in higher satisfaction, *UX Myths - Build your website based on evidence, not false belief*, [online] Available from: http://uxmyths.com/post/712569752/myth-more-choices-and-features-result-in-higher-satisfac (Accessed 1 May 2014).

Zwicky, F. (1971) *Entdecken, Erfinden, Forschen im morphologischen Weltbild*, Munich, Droemer Knaur.

Zwicky, F. (1966) *Entdecken, Erfinden, Forschen im morphologischen Weltbild*, Munich, Droemer Knaur.

Appendix A: Overview of modularisation methods

Appendix A shows an overview of the modularisation methods considered for this work. The methods are ordered into following categories:

- Methods using abstract factors as main input factors
- Methods using strategic and/or physical factors as main input factors
- Methods using integrating various factors as main input
- Methods considering degree of modularity or uncertainty

The overview is given in a table which lists main input factors that are used to establish the product architecture. It also lists the phase in the product development process in which the product architecture is established.

If methods use supporting side input factors to establish the product architecture, they are presented in the table as well, together with the phase in the development process to which the side input factor is assigned to. This phase can either be before the actual modularisation step or afterwards.

The table also contains the mode of product architecture representation. In most of the cases, the product architecture representation is the starting point for product architecture improvement. There are three different possibilities how the product architecture is represented within modularisation methods: a) graph-based, b) matrix-based or c) via mathematical models.

Table A - I: Methods using different factors as input factors

Methods using abstract factors as main input factor								
Method	**Category**	**Main Input Factor**	**Phase**	**Side Input Factor (before modularisation phase)**	**Phase**	**Side Input Factor (after modularisation phase)**	**Phase**	**Architecture representation**
Shamsuzzoha (2011)	physical interactions between elements	strength of interaction between elements	concept	-		-		graph-based
Pimmler and Ep-pinger (1994)	physical interactions between elements	strength of interaction between elements: energy, spatial, material, information	concept	-		-		matrix-based
Helmer (2008)	physical interactions between elements	strength of interaction between elements: spatial, material, informa-tion, energy, structural	concept	-		correction through constraints and feasibility	various	matrix-based
Alizon (2006)	physical interactions between elements	interactions between elements (e.g. components, modules) and flow interactions	concept	-		-		matrix-based
Shan and Chen (2009)	physical interactions between elements	functional correlation, physical correlation, geometrical correlation	concept	-		-		matrix-based, numerical optimisation
Xu et al. (2006)	physical interactions between elements	interaction between elements	concept	-		coordination cost between modules	concept	matrix-based, mathematical optimisa-tion
Huang and Kusiak (1998)	physical interactions between elements	- interaction between elements - suitability of elements for grouping	concept	-		-		matrix-based
Kusiak and Huang (1996)	physical interactions between elements	trade-off between cost and perform-ance	various	interaction between elements	concept	-		graph-based fuzzy logic
Siddique and Rosen (1999)	functional structures	common and variety functional elements	concept	-		-		graph-based

Pahl and Beitz (2007)	functional structures	function structure: basic, special, auxiliary and adaptive functions	concept	- requirements for product variants - market data and economic analysis	requirements	- working principles and concept alternatives - economical factors - technical/qualitative factors - manufacturing factors	concept, design	graph-based
McAdams (1999)	functional structures	functional similarity, common function chains, causal links between flows	concept	- weighted customer needs - functional importance	requirements, concept	-		mathematical models, matrix-based
Holtta (2003)	functional structures	similarity between in- and outputs of functions	concept	-		-		graph-based, mathematical
Stone (1997, 1999, 2000), Day et al. (2010), Kurtadikar et al. (2004)	functional structures	- dominant flows - branching flow - conversion-transmission flow	concept	customer needs and their relation to functions	requirements	- concepts and geometric layouts - standardizing solutions vs. focusing on unique solutions - feasibility of modular concept - continuous evaluation of agreed concept	concept, design	graph-based
Chandrasekaran et al. (2004)	functional structures	re-occurrence of modules in several products	concept	module dependencies	concept			matrix-based
Zamirowski and Otto (1999)	functional structures	- dominant flows - branching flow - conversion-transmission flow - shared functions - unique functions	concept	-		-		graph-based
Dahmus et al. (2001)	functional structures	- dominant flows - branching flow - conversion-transmission flow - shared functions - unique functions - similar target values of functions	concept	external and internal requirements and their relation to modules	requirements	- concept layouts - separation into existing and innovative solutions - continuous evaluation of modular concepts	concept	graph-based
Stake (2000)	functional-physical relations	relationship between functions and technical solutions of products	concept	-		-		matrix-based

Jiao and Tseng (1999), Du et al. (2001, 2005)	functional-physical relations	relationship between functional requirements and design properties	concept	- market data: competitors, technologies, market trends, market segments, requirements, goal values for market segments, sales volume for product attributes - function tree	requirements, concept	- technological feasibility - manufacturability, costs, volume, time schedule - evaluating trade-off: performance vs. cost, commonality vs. Variety - configuration structure with a generic BOM structure	concept, design	matrix-based, graph-based
Goepfert (1998), Goepfert and Steinbrecher (2000)	functional-physical relations	relationship between product functions, components and modules	concept	-		- alignment with organisational issues	various	graph-based
Methods using strategic and/or physical factors as main input factors								
Method	**Category**	**Main Input Factor**	**Phase**	**Side Input Factor (before modularisation phase)**	**Phase**	**Side Input Factor (after modularisation phase)**	**Phase**	**Architecture representation**
Erixon (1996; 1998), Erixon et al. (1996), Nilsson and Erixon (1998), Ericsson and Erixon (1999), Nilsson (2010), Stake (2000), Blackenfelt & Stake (1999)	strategic reasons	- module drivers reflecting needs for modularisation from R&D, product management, assembly, quality, purchasing, service	concept	- market data: competitors, technologies, market trends, market segments, requirements, goal values for market segments, sales volume for product attributes - evaluation and selection of technical solution	requirements, concept	- evaluation and impact of interfaces, lead time, system cost, production cost, quality, development, sales and after-sales service - improvement of modular system with DFM and DFA - module and interface specifications	concept	matrix-based
Borjesson (2009)	strategic reasons	- module drivers reflecting needs for modularisation from R&D, product management, assembly, quality, purchasing, service	concept	- functional flows and heuristics - module driver compatibility	concept, various	- technical constraints for physical solutions - geometrical properties reflecting engineering knowledge	concept	matrix-based

Yu et al. (2011)	strategic reasons	PLC aspects as modular driving forces: functionality, structural aspects, component lifetime, material com-patibility, recyclability	concept	-		-		mathematical optimisa-tion
Ji et al. (2013)	strategic reasons	10 factors from following categories: - functional similarity - structural similarity - material reuse similarity	concept	-		-		mathematical optimisa-tion
Newcomb et al. (1996)	strategic reasons	PLC aspects: material recycling, service, post-life-intend	concept	-		- interaction within modules vs. interactions of system - congruence of the three life cycle viewpoints	concept	matrix-based
Gu et al. (1997), Gu and Sosale (1999)	strategic reasons	PLC and value stream needs: design, manufacturing, operation, service, recycling	concept	- alignment of general objectives with modulari-sation objectives - functional similarity - interactions between components	concept	-		matrix-based
Coulter et al. (1996)	strategic reasons	PLC aspects: material recycling, service, post-life-intend	concept	- functional similarity - interactions between components	concept	- redesign of components to better suit the product archi-tecture strategy	design	matrix-based
Meehan et al. (2007)	holistic: functional and physical factors	- functional factors - working principles - dependencies between physical components	various	-		-		matrix-based
Lange (1998)	holistic: physical and strategic aspects	- technical dependencies between elements (DSM) - strategic factors in terms of module drivers (MIM)	concept	-		-		matrix-based
Lanner and Malm-qvist (1996)	holistic: physical and strate-gic aspects	- technical dependencies between elements (DSM) - strategic factors in terms of module drivers (MIM)	concept	-		-		matrix-based

Blackenfelt (2000, 2001)	holistic: physical and strategic aspects	- strategic factors in terms of module drivers (strategic DSM) - physical dependencies between elements (technical DSM)	concept	-		-		matrix-based
Blees and Krause (2008), Blees et al. (2008), Blees et al. (2009)	holistic: physical and strategic aspects	- physical module interaction graph - module driver analysis (perspective-based module drivers)	concept	- functions of components/ modules - evaluation of module driver fulfilment	concept	-		graph-based, matrix-based
Salhieh and Kamrani (1999), Kamrani and Salhieh (2002)	holistic: functional and physical factors	similarity between components concerning physical specifications and functional requirements	concept	- market needs	requirements	- DFA and DFM methodologies	concept	matrix-based

Methods integrating diverse multiple factors as main input

Method	Category	Main Input Factor	Phase	Side Input Factor (before modularisation phase)	Phase	Side Input Factor (after modularisation phase)	Phase	Architecture representation
Gonzalez-Zugast et al. (2000)	holistic: platform-based	trade-off between individually designed products and platform products in terms of cost, revenue and performance	concept	mathematical models to relate design parameters with requirements, performance, cost, revenue and competition models	requirements, concept, design	-		tables describing the platform performance
Meyer and Lehnerd (1997)	holistic: platform-based	- detailed description of different market segments - platform strategy - core competency analysis - price and performance targets	concept	- implicit and explicit needs - compelling product features	requirements	- interdisciplinary team to introduce and manage the platform	various	graph-based, matrix-based
Simpson et al. (2001)	holistic: platform-based	- scenarios of metamodels relating scaling/design variables with platform specification/performance	various	- market understanding - market segmentation	requirements	-		mathematical representation

DeWeck (2004)	holistic: platform-based	- market segmentation with target prices and design variables related to target specifications - sales volume equation for each market segment - profit function aggregating the platform choices	various	-		-		mathematical represen-tation
Dahl (1990), Bäßler (1987)	holistic: production and logistic processes	- number of assembly steps and assembly order - sub-assembly optimum - integrated assembly parts	concept	- functional hierarchy - assembly-relevant relationships between functional elements	concept	- assemblability of the designed product architecture	design	graph-based
Schuh (1988)	holistic: production and logistic processes	visual impact on process complexity by: - standardization - integration - differentiation - segmentation - vertical range of manufacture - subassemblies	concept	- right product architec-ture alternative	concept	-		graph-based
Caesar (1991)	holistic: production and logistic processes	impact of variant optimization meas-ures on process complexity: - metrics derived from variant tree - costs related to process complexity cost	concept	-		-		graph-based
Emmatty and Sarmah (2012)	holistic: production processes	reference to other methods concern-ing design of product architecture	various	- product requirements - product functions - design knowledge database and common platform parts	requirements, concept, design	- Design for Assembly (DFA) - Design for Manufacturing (DFM)	design	-
Schuh and Jonas (1997), Schuh and Schwenk (2001)	holistic: variant management and modularisation	-optimised scenario between offered features and internal complexity in terms of process complexity - trade-off between complexity cost and direct cost	concept	-		-		graph-based

Kipp and Krause (2008), Blees et al. (2010), Kipp et al. (2010)	holistic: variant management and modularisation	- relation between features, functions, working principles and components - perspective-based module drivers	concept	-		-		graph-based, matrix-based
Ulrich and Eppinger (2012)	holistic: PLC/strategic, functional and physical aspects	- interaction/interfaces between elements - function sharing - certain set of module drivers	various	-		- delayed differentiation in production - trade-off between commonality and variety	various	graph-based
Sand et al. (2001), Sand et al. (2002)	holistic: PLC/strategic, functional and physical aspects	- life cycle characteristics - relation between functions and components - functional and physical relationship between components	concept	-		-		matrix-based
Jonas et al. (2012)	holistic: strategic planning and product structuring	- product structuring with multiple input factors	concept	- strategic product programme planning and analysis	requirements	-		graph-based
Koppenhagen (2004)	holistic: PLC/strategic, functional and physical aspects	- strategic reasons (MIM) - technical dependencies between components (DSM) - customer requirements (QFD)	concept	-		-		matrix-based
Krause et al. (2014), Kruse et al. (2015)	holistic: various factors from different fields	- Design for optimised variety of modules and products - Modularity for different product life cycle phases - Development of platform-based or modular product programs	concept	- product program planning as input for modularisation	requirements	-		graph-based
Koeppen (2008)	holistic: PLC/strategic, functional and physical aspects	mathematical formulation of: - strategic factors - PLC factors - functional factors - technical factors	concept	-		-		matrix-based, mathematical models

Simpson et al. (2012), Thumm and Göhlich (2015), Pakkanen et al. (2015)	holistic: integrating various factors	- factors from product planning - factors to generate variety - factors to generate commonality	concept	- requirement analysis	requirements	-		matrix-based, graph-based, optimisation-based
Kristjansson et al. (2004), Kristjansson & Hildre (2004a), Kristjansson & Hildre (2004b), Kristjansson & Hildre (2004c), Kristjansson (2005)	holistic: various factors from different fields	multiple factors	various	-		-		metric-based

Methods considering degree of modularity or uncertainty

Method	Category	Main Input Factor	Phase	Side Input Factor (before modularisation phase)	Phase	Side Input Factor (after modularisation phase)	Phase	Architecture representation
Marshall and Leaney (2002)	holistic: optimum degree of modularity	- optimum degree of modularity - interactions between elements - relation of functions to elements - PLC and strategic factors	concept	- requirements analysis - criteria for modularisation aligned with overall criteria - functional structure	requirements, concept	- process integration of modularisation - modules are tested against criteria	various	-

Schuh et al. (2007)	holistic: optimum degree of modularity	relation of functions to components: different scenarios and their evaluation regarding practicability and cost	concept	- market data: analysis of customers, competitors, markets, legal issues, technological trends and country-specific needs - internal and external success factors - internal and external requirements - product functions	requirements	- interface description - relation of components/modules to the product portfolio	concept	matrix-based
Fujita et al. (1999), Fujita (2002), Fujita and Yoshida (2004)	holistic: trade-off between merits and demerits of modularity	optimizing the trade-off in different scenarios between: - variety and commonality - performance - cost - price potential - constraints of modular architecture	various	-		-		mathematical models
Yigit (2002)	holistic: trade-off between merits and demerits of modularity	optimizing the trade-off in different scenarios between: - quality and performance loss - cost for reconfigurable manufacturing	various	-		-		mathematical models
Nepal et al. (2008)	holistic: uncertainties	- cost - qualitative manufacturability - degree of interaction between components - impact of change to considered variables in future	various	-		-		mathematical models
Schuh et al. (2009), Schuh et al. (2014)	holistic: uncertainties	link of product architecture alternatives to product features and uncertainty factors	concept	- link between uncertainty factors and features - product development strategy		-		graph-based
Moon et al. (2007)	holistic: uncertainties	mathematical profit model which is optimized for different scenarios	concept	-		-		mathematical models

Kidd (1998)	holistic: uncertainties	estimated development of high-level company performance metrics based different product architecture scenarios and other varying input factors	concept	-		-		mathematical models

Appendix B: Characterisation of case study organisations

Following table gives an overview of all primary and secondary cases that were analysed in order to retrieve data for the qualitative study. The table gives following information for each case:

- Identifier (ID): The identifier gives information about whether a case is from the primary case company or from a secondary case. Moreover, this column gives a unique identifier for each case.
- Cohort: This column gives information about the cohort to which a case belongs to. Each case can either be from the primary case company in the central department, from a development project within the primary case company, or from a secondary case where it may belong to a mature cohort, a mixed cohort or a young cohort.
- Characterisation: This column characterises the case.
- Research methods: This column gives information about the research methods applied for each case.
- Aspect considered: This column explains the support aspects that were considered by the respective case organisation. For instance, it describes whether the case organisation just considered a single aspect for modularisation (e.g. a method or an organisational change) or an integrated set of aspects. The respective aspect applied by the specific case organisation is indicated by an "X" which follows the respective aspect.

Table B – I: Overview and characterisation of all considered case study organisations

ID	Cohort	Characterisation	Research methods	Aspect considered
Primary Case Company - Project A	primary case - central department	- Main product portfolio of entire company transitioning toward modular system development - HVAC industry - Integrated approach for transitioning toward modular system development	o Participation o Observation o Project work o Discussions o Interviews o Document analysis	Process/Method X IT-Integration X Organisation X Implementation X Evaluation X

ID	Cohort	Characterisation	Research methods	Aspect considered
Primary Case Company - Project B	primary case - development projects	- Development project aiming at developing products based on a modular reference architecture - HVAC-industry - Modularisation method (MFD) - Other support elements for modularisation	o Participation o Observation o Project work o Discussions o Interviews o Document analysis	Process/Method X IT-Integration X Organisation X Implementation X Evaluation X
Primary Case Company - Project C	primary case - development projects	- Development project bringing several product families onto the same modular platform - HVAC-industry - Modularisation method (Schuh(2007)-approach) - Other support elements for modularisation	o Participation o Observation o Project work o Discussions o Interviews o Document analysis	Process/Method X IT-Integration X Organisation X Implementation Evaluation X
Secondary Case 1	secondary case - mature cohort	- Development project with aim to modularise product family - Product from HVAC industry (air conditioning appliances) - Modularisation method - Variant management	o Interview o Document analysis o Observation	Process/Method X IT-Integration Organisation Implementation Evaluation
Secondary Case 2	secondary case - mature cohort	- Consultancy specialised in modularisation - < 50 modularisation consultants - Link to university research institute - Modularisation method (MFD) - Other support elements for modularisation	o Discussions o Presentations o Trainings o Project work o Interviews	Process/Method X IT-Integration X Organisation X Implementation X Evaluation X
Secondary Case 3	secondary case - mature cohort	- Company transition toward modular systems across different product lines - Automation machinery industry (drives) - Modularisation method (MFD) - Other support elements for modularisation	o Interview o Document analysis	Process/Method X IT-Integration X Organisation X Implementation X Evaluation X

ID	Cohort	Characterisation	Research methods	Aspect considered
Secondary Case 4	secondary case - mature cohort	- Consultancy specialised in modularisation - < 50 modularisation consultants - link to university research institute - modularisation method (METUS) - other support elements for modularisation	o Interview o Document analysis	Process/Method X IT-Integration X Organisation X Implementation X Evaluation X
Secondary Case 5	secondary case - mature cohort	- Development project aiming at developing products based on modular reference architecture - System control units - Modularisation method (METUS) - Other support elements for modularisation	o Interview o Document analysis	Process/Method X IT-Integration Organisation Implementation Evaluation X
Secondary Case 6	secondary case - mature cohort	- All development projects of company using modularisation method in order to modularise their product architecture - Home appliances (dryers) - Modularisation method (METUS) - Other support elements for modularisation	o Interview o Document analysis	Process/Method X IT-Integration X Organisation Implementation Evaluation X
Secondary Case 7	secondary case - mature cohort	- Strong focus of all development projects on complexity reduction - Modularisation not primary goal of company - Home appliances (dishwashers) - Complexity management approach	o Interview o Document analysis o Site visit	Process/Method IT-Integration Organisation Implementation X Evaluation X
Secondary Case 8	secondary case - mature cohort	- New modular platform of company built around a modular organization - Home appliances (cookers) - Modularisation organisation	o Interview o Document analysis o Site visit	Process/Method IT-Integration Organisation X Implementation X Evaluation

ID	Cohort	Characterisation	Research methods	Aspect considered
Secondary Case 9	secondary case - mature cohort	- Consultancy with focus on structural complexity management of products - < 10 consultants - Structural complexity management	o Interview o Document analy-sis	Process/Method X IT-Integration Organisation Implementation Evaluation
Secondary Case 10	secondary case - mature cohort	- Development project modularising its products in order to reduce com-plexity - High pressure pumps from automotive industry - structural complexity management	o Interview o Document analy-sis	Process/Method X IT-Integration Organisation Implementation Evaluation
Secondary Case 11	secondary case - mature cohort	- Development project from automotive industry implementing a classical Product Line Approach across different projects - Airbag control units - Product line approach	o Interview o Document analy-sis	Process/Method X IT-Integration Organisation X Implementation Evaluation
Secondary Case 12	secondary case - mature cohort	- Company-internal con-sultancy from automotive industry with a focus on variant management - Broad product scope - Variant management approach	o Interview o Document analy-sis	Process/Method X IT-Integration Organisation X Implementation Evaluation X
Secondary Case 13	secondary case - mature cohort	- Company restructures its entire range of products in order to make them con-figurable - Similar to modular ap-proach - Boilers for industry - Configuration manage-ment approach	o Interview o Document analy-sis o Participant-observation	Process/Method IT-Integration X Organisation Implementation Evaluation
Secondary Case 14	secondary case - mature cohort	- Consultancy specialised in modularisation and process efficiency - < 50 consultants - Modularisation method	o Interview o Document analy-sis	Process/Method X IT-Integration Organisation Implementation Evaluation

ID	Cohort	Characterisation	Research methods	Aspect considered
Secondary Case 15	secondary case - mature cohort	- Gathering of experts at large automotive supplier in order to share expertise on how to structure products for configuration - Similar to modularisation approach - Mainly experts from automotive industry, but also participants from automation technology - Configuration management approach	o Interview o Document analysis	Process/Method IT-Integration X Organisation X Implementation Evaluation
Secondary Case 16	secondary case - mature cohort	- Platform development projects with strong control of variety of derivative product development projects - Automotive supplier industry - Organisation: organisational separation between platform development and customer projects	o Interview o Document analysis	Process/Method IT-Integration Organisation X Implementation Evaluation
Secondary Case 17	secondary case - mixed cohort	- Development project aiming at developing products based on a modular reference architecture - HVAC-industry - Approximately 20 engineers - Modularisation method (MFD) - Other support elements for modularisation	o Project work o Discussions o Interviews o Document analysis	Process/Method X IT-Integration X Organisation X Implementation X Evaluation X
Secondary Case 18	secondary case - mixed cohort	- Consultancy specialised in modularisation and variant management - > 50 modularisation consultants - Link to university research institute - Modularisation method (Schuh(2007)-approach) - Other support elements for modularisation	o Interview o Document analysis o Joint project work	Process/Method X IT-Integration Organisation X Implementation X Evaluation X

ID	Cohort	Characterisation	Research methods	Aspect considered
Secondary Case 19	secondary case - mixed cohort	- Full transition of whole company (supporting central department and development projects) towards modular system development - Home appliances (cookers) - Modularisation method (Schuh(2007)-approach) - Other support elements for modularisation	o Extended and regular interviews o Document analysis	Process/Method X IT-Integration Organisation X Implementation X Evaluation X
Secondary Case 20	secondary case - mixed cohort	- All development projects of the company transitioning toward more modular product architectures - Support from a consultancy - Heavy industrial packaging machines - Modularisation method	o Interview o Document analysis	Process/Method X IT-Integration Organisation Implementation Evaluation
Secondary Case 21	secondary case - mixed cohort	- Central department of a large manufacturer preparing modularisation transition - Traditionally, company with a strong focus on complexity management - Large manufacturer for all kinds of home appliances - Central department support for modularisation and complexity management	o Interview o Document analysis	Process/Method IT-Integration Organisation Implementation X Evaluation X

ID	Cohort	Characterisation	Research methods	Aspect considered
Secondary Case 22	secondary case - mixed cohort	- Central Department gathering experts from its different business units in order to share experience on modularisation and to derive a guideline from that expertise - Mainly automotive supplier industry, but also infrequent participation of other business units like power tools - Central department support for modularisation and variant management	o Interviews o Document analysis o Participant-observation	Process/Method X IT-Integration Organisation X Implementation X Evaluation X
Secondary Case 23	secondary case - younger cohort	- Central engineering support department aiming at introducing functional modularisation methods - Automation technology - Modularisation method	o Interview o Document analysis	Process/Method X IT-Integration Organisation Implementation Evaluation
Secondary Case 24	secondary case - younger cohort	- Development department trying to cut down complexity by increasing control and transparency - Automotive supplier industry - Central department support for modularisation and complexity management	o Interview o Document analysis	Process/Method IT-Integration X Organisation X Implementation Evaluation
Secondary Case 25	secondary case - younger cohort	- Large electronic and software systems manufacturer on the verge toward modularisation - Safety technology - Company transitioning toward modular system development	o Interview o Document analysis o Participant-observation	Process/Method X IT-Integration Organisation X Implementation Evaluation

ID	Cohort	Characterisation	Research methods	Aspect considered
Secondary Case 26	secondary case - younger cohort	- Transition toward modular system development for main product portfolio of company - Truck industry - Company transitioning toward modular system development	o Interview o Document analysis	Process/Method IT-Integration X Organisation X Implementation X Evaluation
Secondary Case 27	secondary case - younger cohort	- Transition toward modular system development for main product portfolio of company - Power tool industry - Company transitioning toward modular system development	o Interview o Document analysis	Process/Method X IT-Integration Organisation X Implementation Evaluation

For confidentiality reasons, it is not possible to name the collaborating industrial partners. Moreover, for the same reason, details that could be used for identification were requested to be removed.

Several cases are distinct entities of the same group. For instance, different business units or divisions with a different product portfolio or industrial setting of the same group/company have been considered as a distinct case.

The directly involved development sites are located in eight different countries: Sweden, England, USA, Germany, Turkey, Netherlands, Portugal and France.

The time spent for each case varied between several hours and several years.

Appendix C: Questionnaire for benchmark analysis

Several secondary cases that were mature and experienced in modularisation were interviewed and visited before modularisation transition was started at the primary case company. It was the purpose of collaborating with these more "mature cohorts" to learn and to find out what went wrong and what worked out well during their modularisation transition. Consequently, a site visit with these secondary cases from industry was arranged. On the agenda of each site visit was at least a semi-structured interview with a senior modularisation expert or with a senior engineering manager. In order to facilitate preparation for the semi-structured interview, a questionnaire was sent out to each modularisation expert. The questionnaire contained following questions:

Questions for more mature secondary cases in industry

Design of modular product architectures

- How do you design your modular product architecture?
- How do you design module variants?
- Which rules, processes, and methods do you use for that?
- What has to be considered during modularisation, based on your experience?

Technical implementation of modules and interfaces

- How and where do you specify modules and interfaces?
- How do you represent the modular system in your IT-Systems (e.g. CAD, PDM, ERP)?
- Do you separate modules from interfaces?
- How do you configure your products?

Administration of modules and interfaces

- How do you administrate modules?
- How do you administrate interfaces?
- How do you ensure that modules and interfaces are kept stable over time?
- How do you prevent changes to modules and interfaces?

- How do you prevent the introduction of new parts and product variants?
- How do you define if it makes sense to introduce a new product variant?

Implementation Strategy

- How did you implement modularisation within your / an organisation?
- How did / do you build-up know-how for modularisation?
- How did you find out which means of support is best suitable for you?
- How did you find out which effort is best suitable to invest for modularisation?
- Which roles did you implement for product architecture related processes in your organisation?
- How did you integrate product architecture processes into existing engineering design processes?

Benefits from modularisation

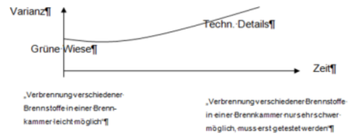
- How much cost could you save after introducing modularisation?
- How many part numbers could you reduce when introducing new product platforms? (for consultants: based on your experience)
- Could you measure if there is less increase in part numbers during the product life cycle?
- How much practical experience do you have with modularisation?
- What were the costs and benefits for you? (for industry only)
- What are your metrics for modularisation and how do they develop?
- How do you evaluate the product architecture?

Appendix D: Export of coding database

The following table shows an export of the coding database with a special focus on issues during modularisation transition.

Table D – I: Export of coded issues

Id	Disguised Source	Issue	Product Design / Life cycle Phase	Aspect Classification	Sorting Classification	Company Function	Scope Classification	Modular System Design / Life cycle Phase	Important Factor	Important Support	Contradictory to ...
1	Secondary case	Product development project cannot manage effort of extended requirements engineering	PLC - Product Development: Requirements Phase	Process	n.a.	Marketing / Product Management	Product / Project Scope	n.a.	1	1	
2	Primary case C	In single projects that shall accommodate modularisation, there is not enough time to test newly conceived modules on fulfilment of customer demand (see comment)	PLC - Product Development: Requirements Phase	Process	n.a.	Marketing / Product Management	Product / Project Scope	n.a.	6	8	
3	Secondary case	A lot of customer demands come afterwards, there are always new customer requirements, very hard task for PRM to fix attributes for modules	PLC - Product Development: Requirements Phase	Process	n.a.	Marketing / Product Management	n.a.	n.a.			
4	Secondary case	Modularisation processes from consultancies want to start fixing requirements at the very beginning but companies want to stay flexible	PLC - Product Development: Requirements Phase	n.a.	n.a.	Marketing / Product Management	n.a.	n.a.	31	37	
5	Primary case C	If we are too broad at the beginning, we might encounter the risk of being fed up with too many tasks => we will not be able to finish	PLC - Product Development: Requirements Phase	Process	n.a.	Marketing / Product Management	Modular System Scope	Developing Modular System			

Id	Disguised Source	Issue	Product Design / Life cycle Phase	Aspect Classifi-cation	Sorting Classifica-tion	Company Function	Scope Classifica-tion	Modular System De-sign / Life cycle Phase	Impor-tant Factor	Impor-tant Support	Contra-dictory to ...
6	Primary case C	During requirements collection and analysis, engineers were very disappointed because there were too many "tbd"	PLC - Product Development: Requirements Phase	Process	n.a.	Marketing / Product Management	Modular System Scope	Developing Modular System			
7	Secondary case	If we want to fulfil every customer requirement or if we want to be very fast, we should not modularise our products (see Mr. XY, secondary case from packaging machines) -> actually exactly that is associated with modularisation	PLC - Product Development: Requirements Phase	Introduc-tion	n.a.	Marketing / Product Management	n.a.	n.a.			
8	Secondary case	Requirements: Market/operational requirements are not clear at DEV project start, but that has to be known before starting with development	PLC - Product Development: Requirements Phase	Process	n.a.	Marketing / Product Management	Modular System Scope	Planning Modular System			
9	Primary case C	 ENG: Viele Anforderungen, auch solche die eigentlich ins Lastenheft gehören würden, kommen erst auf, wenn entwickelt wird / ist.¶ Die Bereitschaft zur Variantenreduktion nimmt im zeitlichen Verlauf ab.¶ ENG: Problem Festlegung der Anforderungen 2 Jahre vor Produktionsstart. Z.B. das Design (dynamische Anforderung) wird sich bis SOP wieder verändern. Die Lösung besteht darin diese dynamischen Anforderungen kurzfristig veränderbar zu machen, dabei aber keine an-deren Module zu betreffen.¶	PLC - Product Development: Requirements Phase	Process	n.a.	Marketing / Product Management	Modular System Scope	Planning Modular System			
10	Primary case C	Primary case C pilot project took more than one year. After that year, we had nothing but a .ppt concept that was needed for management. We spent most of the time on requirements and feature analysis. The real technical concept was still very vague.	PLC - Product Development: Requirements Phase	Process	n.a.	Marketing / Product Management	Modular System Scope	Planning Modular System			

Id	Disguised Source	Issue	Product Design / Life cycle Phase	Aspect Classification	Sorting Classification	Company Function	Scope Classification	Modular System Design / Life cycle Phase	Important Factor	Important Support	Contradictory to ...
11	Secondary case	No mechanisms how module variant commitments are sustainably achieved => planning of low value	PLC - Product Development: Concept Phase	Process	Standardisation	R & D	Product / Project Scope	n.a.	2	2, 3	
12	Secondary case	In single projects there is not enough time and expertise to develop newly introduced technical solutions for the modular concept (see comments)	PLC - Product Development: Concept Phase	Process	n.a.	R & D	Product / Project Scope	n.a.	7	9	
13	Primary Case B	We did not identify an algorithm or method that has really good suggestions for modules => manual engineering work was required in all cases	PLC - Product Development: Concept Phase	Process	n.a.	R & D	n.a.	n.a.			
14	Secondary case	In Benchmark Studies, single products are compared to highly integrated products (cheapest in class) without looking at the overall picture => Modular products will always loose.	PLC - Product Development: Concept Phase	Organisation	Financial	Diverse	Product / Project Scope	n.a.	34	40	
15	Secondary case	Modularisation process is very hard for design engineers as they have to wait much longer until they can actually start drawing	PLC - Product Development: Concept Phase	Process	n.a.	Diverse	n.a.	n.a.			
16	Secondary case	Functional module clustering not compliant with necessary module clustering for manufacturing	PLC - Product Development: Concept Phase	Process	n.a.	Diverse	n.a.	n.a.			
17	Primary case A	In the end, direct cost rule the product => especially faced strong competition from Asia who have low-cost single runners	PLC - Product Development: Concept Phase	n.a.	Financial	Diverse	n.a.	n.a.			
18	Secondary case	Methods always come up with different architectures / modules (see Primary Case B, Study of Holtta, Primary Case C, different architectural views through module drivers)	PLC - Product Development: Concept Phase	Process	n.a.	R & D	n.a.	n.a.	39	55	
19	Secondary case	Each architectural strategy has distinctive requirements, pros and cons	PLC - Product Development: Concept Phase	Process	n.a.	R & D	n.a.	n.a.	40	56	

Id	Disguised Source	Issue	Product Design / Life cycle Phase	Aspect Classification	Sorting Classification	Company Function	Scope Classification	Modular System Design / Life cycle Phase	Important Factor	Important Support	Contradictory to ...
20	Primary case C	Trade-off: if platform is too broad, there will be too many compromises and it will take too much time; if platform is too narrow, we don't have enough synergies.	PLC - Product Development: Concept Phase	n.a.	n.a.	Diverse	Modular System Scope	Developing Modular System			
21	Primary case C	Methods are not idiot-proof => sometimes those who did not attend workshops found mistakes in the modular concept	PLC - Product Development: Concept Phase	Process	n.a.	R & D	n.a.	n.a.			
22	Primary case C	Module driver analysis did not bring in sufficient new insights for development: a) not enough time, b) modules were already clear, c) other company functions did not have enough details to fix their view on modules during that process phase	PLC - Product Development: Concept Phase	Process	n.a.	Diverse	n.a.	n.a.			
23	Primary case C	Module driver analysis: there is no method that delivers sufficient results vs. Why do we need that, engineer has to rework it later on anyway	PLC - Product Development: Concept Phase	Process	n.a.	R & D	n.a.	n.a.			
24	Primary case C	Module driver analysis did not bring the expected advantages by talking about patterns and sums in Excel spreadsheets => holistic overall view gets lost (by the way this is also valid for other parts of the methods)	PLC - Product Development: Concept Phase	Process	n.a.	Diverse	n.a.	n.a.	45	78	
25	Primary case C	For every "template" that you provide to engineers, they form resistance: a) it is in our heads, b) we don't know it yet, c) time	PLC - Product Development: Concept Phase	Process	n.a.	Diverse	n.a.	n.a.			
26	Primary case C	When applying modularisation method: in most cases there are existing technical solutions for product functions => it is not necessary to decompose functions into elementary functions like physical, chemical or biological flows/dependencies => functional heuristics are not appropriate => where is the use of a method that comes only improperly close to find out already existing modules??	PLC - Product Development: Concept Phase	Process	System/Product Architecting	R & D	n.a.	n.a.			

Id	Disguised Source	Issue	Product Design / Life cycle Phase	Aspect Classification	Sorting Classification	Company Function	Scope Classification	Modular System Design / Life cycle Phase	Important Factor	Important Support	Contradictory to ...
27	Primary case C	Method: The good thing when coming from the functional side is that we are solution-neutral. However, modularisation is not meant for making experiments, coming to new solution is necessary but also stretches the modularisation process and makes it resource-intensive. Coming to new ideas is not the purpose of modularisation.	PLC - Product Development: Concept Phase	Process	n.a.	R & D	n.a.	n.a.			
28	Primary case A	Problem with module clustering with graph-based approach (method ABCDE) was that the complexity of the system control could not be handled with the tool anymore	PLC - Product Development: Concept Phase	Process	System/Product Architecting	Diverse	Modular System Scope	n.a.			
29	Primary case C	Modules are always seen from an organisational perspective, module driver clustering does not bring anything but compromises that do not lead anywhere. Moreover, module driver clustering did not even reveal following issue: e.g. chamotte inside or outside burning chamber? ENG: inside, MFG: outside, Service: outside, Installer: outside => this discussion was revealed during IT-Integration and IT-integration was better than method for this! => method was filling Excel file without broad view, filling huge templates instead of change in mindset due to lack of time and capacity	PLC - Product Development: Concept Phase	Process	n.a.	R & D	Modular System Scope	n.a.			
30	Primary case A	After modularisation method in Onion-peel model: We got good modules over different layers (different parts of it on different layers; see primary case C where we used average calculations but this is pseudo-correct) => do we have to cut modules in a new way?? Actually no as this all is a big compromise; we also cannot do the onion peel model on part level (like it is done at secondary case for home appliances) for our organisational concept	PLC - Product Development: Concept Phase	Process	n.a.	R & D	Modular System Scope	n.a.			

Id	Disguised Source	Issue	Product Design / Life cycle Phase	Aspect Classification	Sorting Classification	Company Function	Scope Classification	Modular System Design / Life cycle Phase	Important Factor	Important Support	Contradictory to ...
31	Primary case C	During Design phase during defining the physical product structure or during assigning attributes to modules or parts, one realises that the modular concept does not work in the way suggested by the method (e.g. XYZ) => see support	PLC - Product Development: Concept Phase	Process	n.a.	R & D	Modular System Scope	n.a.	56	124	
32	Primary case C	It is easier to deviate from the product architecture or to create new variants than using artefacts that are already available	PLC - Product Development: Design Phase	n.a.	n.a.	R & D	Product / Project Scope	n.a.	2,3,4, 8	2,3,4,5,7	
33	Primary Case B	Once the concept was established, management decided to cooperate with another company => whole modular concept was spoilt => constant changes are acid for modularisation	PLC - Product Development: Design Phase	Process	n.a.	R & D	Product / Project Scope	n.a.			
34	Primary Case B	On the one hand we need early decisions, on the other hand there will always be changes	PLC - Product Development: Design Phase	Process	n.a.	R & D	Product / Project Scope	n.a.	23	24	
35	Secondary case	Strong local responsible try to spoil module variance targets and interfaces	PLC - Product Development: Design Phase	Process	n.a.	R & D	n.a.	n.a.	29	29	
36	Secondary case	Modularisation compromises freedom of developer who does not like that.	PLC - Product Development: Design Phase	n.a.	n.a.	R & D	n.a.	n.a.			
37	Secondary case	No direct benefit for designer through modularisation.	PLC - Product Development: Design Phase	n.a.	n.a.	R & D	n.a.	n.a.			
38	Primary case C	e.g. at the beginning we should freeze the architecture, but we did not know all market requirements or whether we can derive XYZ or XYZ from the same modular system	PLC - Product Development: Design Phase	n.a.	n.a.	R & D	Modular System Scope	Developing Modular System			
39	Primary case C	If we start modularisation too early, and if we don't plan enough resources, we are not confident enough to make safe design decisions	PLC - Product Development: Design Phase	Process	n.a.	R & D	Modular System Scope	Developing Modular System			

Id	Disguised Source	Issue	Product Design / Life cycle Phase	Aspect Classifi-cation	Sorting Classifica-tion	Company Function	Scope Classifica-tion	Modular System De-sign / Life cycle Phase	Impor-tant Factor	Impor-tant Support	Contra-dictory to ...
40	Primary case C	After architecting, as soon as the first product is detailed it gets difficult to design multi-purpose modules instead of designing individualised modules (see remark)	PLC - Product Development: Design Phase	Process	n.a.	R & D	n.a.	n.a.			
41	Primary case C	During designing first modules, different other technical solutions that have to be considered are not yet clear. => difficulties to fully focus on one product & on whole future modular system => High drive of project team to focus on only one current product with known technical solutions	PLC - Product Development: Design Phase	Process	n.a.	R & D	n.a.	n.a.			
42	Primary case C	You cannot specify interfaces before the products are developed	PLC - Product Development: Design Phase	Process	n.a.	R & D	Modular System Scope	n.a.			
43	Primary case C	At the end of the workshop series with consultants , we realised that even our features were everything but fix. Moreover, we haven't decided which features to cover by the modular system and which to cover in parallel outside the modular system.	PLC - Product Development: Design Phase	Process	n.a.	Diverse	Modular System Scope	n.a.			
44	Primary case C	ENG: Actually central platform development needed => developing modular system with first product variant without knowing modular system in detail has enormous risks and is more than a gamble	PLC - Product Development: Design Phase	Organisa-tion	n.a.	R & D	Modular System Scope	n.a.			
45	Primary case C	Projects just do not get the scope toward considering other areas and products	PLC - Product Development: Design Phase	n.a.	n.a.	R & D	Modular System Scope	n.a.			
46	Primary case B	After 1.5 years of development, primary case B felt back to the situation where they said that it is questionable if they can share the modules for small and mid range for the large range as well, this is not clear but it is considered => three different chassis-sizes, module envelopes and interfaces, different interface positions	PLC - Product Development: Design Phase	Process	n.a.	R & D	Modular System Scope	n.a.			

Id	Disguised Source	Issue	Product Design / Life cycle Phase	Aspect Classification	Sorting Classification	Company Function	Scope Classification	Modular System Design / Life cycle Phase	Important Factor	Important Support	Contradictory to ...
47	Primary case C	Very likely that product for primary case C falls out of modular system between A- and B-Sample as engineers are allergic to modularisation. As soon as the attention stopped, they totally neglected the modular system (rearranging solutions, discussing about features, not maintaining transfer documents)	PLC - Product Development: Design Phase	n.a.	n.a.	R & D	Modular System Scope	Developing Modular System			
48	Primary case C	Transfer documents as glue for modular system are too weak.	PLC - Product Development: Design Phase	Process	n.a.	Diverse	Modular System Scope	n.a.			
49	Primary case A	As long as we have different structures within a company (TC, SAP), we cannot compare by structures - precondition that we have different BOMs within a company (ENG, PROD, Service) - We have to compare modules by attributes and not by BOM-structure - no direct/automated transfer of part-module attributes between systems?? - interesting that we have different modules between different functions, link to flaw in modularisation methodologies	PLC - Product Development: Design Phase	IT	n.a.	R & D	Modular System Scope	n.a.			
50	Primary case C	Consultation with Mr. XY: Assessment of 1st Product of Primary Case C (the "modular" one) did not contain any modules, but single parts on first level => - same as before - no modularisation - parts on first level - business as usual - in CAD, SAP, TC no difference to before recognizable => transition from concept to design failed totally! => such situations cannot happen with IT-Integration and evaluation	PLC - Product Development: Design Phase	n.a.	n.a.	R & D	Modular System Scope	n.a.			

Id	Disguised Source	Issue	Product Design / Life cycle Phase	Aspect Classification	Sorting Classification	Company Function	Scope Classification	Modular System Design / Life cycle Phase	Important Factor	Important Support	Contradictory to ...
51	Primary case C	During IT-Integration, it got evident that it is sooo difficult to make the transition from modular concept to actual design phase (it was so easy during potential analysis and during method) => the issues are pushed through and get bigger like a snowball - it is very difficult to achieve consistent namings and classification for modules or parts that have not even been modules before - almost impossible to assign always the same part to the same module => primary case C design engineers still work on the product instead of on the modules	PLC - Product Development: Design Phase	n.a.	n.a.	R & D	Modular System Scope	n.a.			
52	Primary case C	During IT-Integration, it got evident that the whole concept lost ist stability already: transfer documents contradicted each other and were not coherent (variant tree versus morphological box versus onion peel model) => the once neat concept got much too complex for the team already quite early!	PLC - Product Development: Design Phase	Process	n.a.	R & D	Modular System Scope	n.a.			
53	Secondary case	We always use new interfaces, instead of using them from predecessor: - interfaces have never been developed for several projects - no designated standard interfaces or reuse process => missing knowledge about that, especially what is used in other countries => laziness => project-own interest => no consequences in case of no reuse	PLC - Product Development: Design Phase	Diverse	Standardisation	R & D	n.a.	n.a.			
54	Secondary case	A product is actually developed for multi-purpose markets, but due to knowledge, costs and time it is only developed for one single project; moreover, due to delivery problems projects became more weight compared to line organisation	PLC - Product Development: Design Phase	Process	n.a.	Diverse	Modular System Scope	Developing Modular System			

Id	Disguised Source	Issue	Product Design / Life cycle Phase	Aspect Classifi-cation	Sorting Classifica-tion	Company Function	Scope Classifica-tion	Modular System De-sign / Life cycle Phase	Impor-tant Factor	Impor-tant Support	Contra-dictory to ...
55	Secondary case	Engineers are afraid to lose freedom and creativity with modularisation: designers never like to do what they are told, each time they want to start from scratch	PLC - Product Development: Design Phase	Diverse	n.a.	R & D	n.a.	n.a.			
56	Secondary case	In HW development, they say each time that they do not have the time to think one time properly instead of doing the same thing again and again in each project.	PLC - Product Development: Design Phase	Process	Change Manage-ment	R & D	Product / Project Scope	n.a.			
57	Primary Case C	In single projects there is not enough time and expertise to test newly introduced technical solutions for the modular concept (see comments)	PLC - Product Development: Testing	Process	n.a.	R & D	Product / Project Scope	n.a.			
58	Secondary case	For modularization, we need working concept at least until A-Sample (Mr. XY)	PLC - Product Development: Testing	Process	n.a.	R & D	n.a.	n.a.			
59	Primary case C	We don't know if the modular CONCEPT works and how/when the concept can be frozen (experience, simula-tion, test -> takes too long??)	PLC - Product Development: Testing	Process	n.a.	R & D	Modular System Scope	Developing Modular System			
60	Secondary case	Difference between design and manufacturing BOMS (even different BOMS at different manufacturing sites)	PLC - Product Development: Production Ramp-up	IT	n.a.	R & D	n.a.	n.a.			
61	Secondary case	Modularisation methods (like several secondary cases) do not consider evaluation of modular system after first prod-ucts.	Evolution of Products, Reuse, Design Modifica-tions, Engineer-ing Change	Process	n.a.	R & D	Product / Project Scope	n.a.			
62	Primary Case B	With current approaches it is not possible to manage inter-faces, modules and module variants separately	Evolution of Products, Reuse, Design Modifica-tions, Engineer-ing Change	Process	n.a.	R & D	Product / Project Scope	n.a.	24	25	

Id	Disguised Source	Issue	Product Design / Life cycle Phase	Aspect Classification	Sorting Classification	Company Function	Scope Classification	Modular System Design / Life cycle Phase	Important Factor	Important Support	Contradictory to ...
63	Secondary case	Platforms are drifting apart because of: local sourcing for products in different markets	Evolution of Products, Reuse, Design Modifications, Engineering Change	Process	n.a.	Diverse	Product / Project Scope	Maintenance of Modular System			
64	Secondary case	Platforms are drifting apart because of: different level of vertical integration	Evolution of Products, Reuse, Design Modifications, Engineering Change	Process	n.a.	Diverse	Product / Project Scope	Maintenance of Modular System			
65	Secondary case	Platforms are drifting apart because of: rationalisation projects	Evolution of Products, Reuse, Design Modifications, Engineering Change	Process	n.a.	Diverse	Product / Project Scope	Maintenance of Modular System			
66	Secondary case	Escalation procedure unclear if there is a conflict between module responsible and a project => avoid such conflicts through right process organisation	Evolution of Products, Reuse, Design Modifications, Engineering Change	Process	n.a.	Diverse	n.a.	n.a.			
67	Secondary case	Platforms are drifting apart because of: Central decisions or decisions from other local entities are not accepted by certain development sites => general resistance against such decisions	Evolution of Products, Reuse, Design Modifications, Engineering Change	Process	n.a.	Diverse	n.a.	n.a.			
68	Secondary case	Platforms are drifting apart because of: introduction of different technology (e.g. from LCD vs. Tubes in TVs)	Evolution of Products, Reuse, Design Modifications, Engineering Change	Process	n.a.	Diverse	n.a.	n.a.			

Id	Disguised Source	Issue	Product Design / Life cycle Phase	Aspect Classification	Sorting Classification	Company Function	Scope Classification	Modular System Design / Life cycle Phase	Important Factor	Important Support	Contradictory to ...
69	Primary case A	Platforms are drifting apart because of: time pressure => development only for target market	Evolution of Products, Reuse, Design Modifications, Engineering Change	Process	n.a.	Diverse	Product / Project Scope	Maintenance of Modular System			
70	Primary case A	Project or even management decides to take another module/feature, or requirement	Evolution of Products, Reuse, Design Modifications, Engineering Change	Process	n.a.	Marketing / Product Management	Modular System Scope	Maintenance of Modular System			
71	Primary case A	Decision to add another part or that the interface cannot be kept stable.	Evolution of Products, Reuse, Design Modifications, Engineering Change	Process	n.a.	R & D	Modular System Scope	Maintenance of Modular System			
72	Primary case A	Most changes come after actual fixation: more requirements come over time, change of requirements, product is designed for DACH market => later decision to go into US-Market, (technical) problems lead to changes, ratio projects lead to changes	Evolution of Products, Reuse, Design Modifications, Engineering Change	Process	n.a.	Diverse	Modular System Scope	Maintenance of Modular System			
73	Secondary case	No direct negative effects on designers when they create new variants or violate modular system specs. But adverse effects in all other areas.	Evolution of Products, Reuse, Design Modifications, Engineering Change	n.a.	Standardisation	R & D	n.a.	n.a.			
74	Secondary case	Sometimes violation against product architecture rules is just laziness or "not knowing it better"	Evolution of Products, Reuse, Design Modifications, Engineering Change	n.a.	n.a.	R & D	n.a.	n.a.			

Id	Disguised Source	Issue	Product Design / Life cycle Phase	Aspect Classification	Sorting Classification	Company Function	Scope Classification	Modular System Design / Life cycle Phase	Important Factor	Important Support	Contradictory to ...
75	Secondary case	Modular systems are breaking apart over and over again (see support)	Evolution of Products, Reuse, Design Modifications, Engineering Change	n.a.	n.a.	Diverse	Modular System Scope	Developing Modular System	38	55	
76	Primary case C	Even though, product architecture and interfaces are defined, module envelopes and interfaces will continue to be adopted	Evolution of Products, Reuse, Design Modifications, Engineering Change	Process	n.a.	R & D	n.a.	n.a.			
77	Primary case C	If top-down-support is not available, large-scale platforms always failed in the past as the focus stays on single products and different areas didn't work together.	Evolution of Products, Reuse, Design Modifications, Engineering Change	n.a.	n.a.	R & D	n.a.	n.a.			
78	Primary case C	Only a very little part of the modular system gets detailed at a time, the rest will be done in future => real problems arise not before other parts of the modular system get detailed => this really endangers the so far "theoretical" modular system => this entails gradual changes on the overall architecture	Evolution of Products, Reuse, Design Modifications, Engineering Change	Process	n.a.	R & D	Modular System Scope	Maintenance of Modular System			
79	Primary case A	Modular systems are often neglected or forgotten after some time => processes, IT, financial pressure	Evolution of Products, Reuse, Design Modifications, Engineering Change	Process	n.a.	R & D	Modular System Scope	Maintenance of Modular System			
80	Primary case B	Drawback of primary case B after 1.5 years: "the focus of this platform is not only on modularisation, we have to develop new technologies for the best price, quality and performance	Evolution of Products, Reuse, Design Modifications, Engineering Change	Evaluation	n.a.	R & D	Product / Project Scope	n.a.			

Id	Disguised Source	Issue	Product Design / Life cycle Phase	Aspect Classification	Sorting Classification	Company Function	Scope Classification	Modular System Design / Life cycle Phase	Important Factor	Important Support	Contradictory to ...
81	Primary case A	The past has shown that disciplinary solution alone does not work, even responsible persons will not adhere to it (e.g. responsible engineers) => reason for IT-Integration, maybe it works with that	Evolution of Products, Reuse, Design Modifications, Engineering Change	Organisation	n.a.	R & D	Product / Project Scope	n.a.	IT-Integration	IT-Integration	
82	Secondary case	Assessing the product architecture during the product life cycle is a vital activity: - without defined and reviewed product architecture => no product - Development must be in accordance with product architecture - Is this according to what was defined in terms of product? BUT The real challenge is to overcome: - re-using modules between different products and development projects - "das haben wir schon immer so gemacht"	Evolution of Products, Reuse, Design Modifications, Engineering Change	Diverse	n.a.	R & D	n.a.	Diverse			
83	Secondary case	All changes at the modular system have to be agreed by top management. However, there is the danger that such decision a) slow down the process, b) decision makers just click but actually do not have the understanding or the time (see management system workflow)	Evolution of Products, Reuse, Design Modifications, Engineering Change	Process	n.a.	R & D	Modular System Scope	Maintenance of Modular System	n.a.	128	
84	Secondary case	The Sys EM Life cycle is very well described, however the interplay between several systems and projects is considered but only very poorly (only in several steps or via PLM)!	Evolution of Products, Reuse, Design Modifications, Engineering Change	Process	n.a.	Diverse	Product / Project Scope	n.a.			

Id	Disguised Source	Issue	Product Design / Life cycle Phase	Aspect Classification	Sorting Classification	Company Function	Scope Classification	Modular System Design / Life cycle Phase	Important Factor	Important Support	Contradictory to ...
85	Secondary case	To synchronize the development life cycle of modules with the development life cycle of products: product readiness versus module readiness	Evolution of Products, Reuse, Design Modifications, Engineering Change	Process	n.a.	Diverse	Modular System Scope	Developing Modular System			
86	Secondary case	It is very challenging to master maintenance/evolution of products and modules while keeping the architecture stable. E.g. though minimized coupling between modules => however there is much much more	Evolution of Products, Reuse, Design Modifications, Engineering Change	Diverse	n.a.	R & D	Modular System Scope	Maintenance of Modular System			
87	Secondary case	Secondary case product X was principally a generic product, but in the meantime every project builds its own solution based on the Finland secondary base product X => releases get branched wildly!	Evolution of Products, Reuse, Design Modifications, Engineering Change	Diverse	n.a.	R & D	Product / Project Scope	n.a.			
88	Primary case A	An important issue in transitioning is the short-term disadvantage in production and logistics for running a traditional and a modular product in parallel (it was not even possible to switch everything to Torx-Screws)	Value Stream - Production	n.a.	n.a.	n.a.	n.a.	n.a.			
89	Primary case C	Modularisation in single projects requires early and fixed decisions while always having open questions that cannot be solved before development or test => requires a lot of time and many iterations	Diverse	Process	n.a.	n.a.	Product / Project Scope	n.a.	9	10	
90	Secondary case	Building modules for a proper modular system is too complex and too time consuming for a normal product development project to handle	Diverse	Process	n.a.	n.a.	Product / Project Scope	n.a.	10	11	
91	Secondary case	Usually modularisation for single products is implemented during workshops which is not suitable for sustainable implementation of modularisation	Diverse	Introduction	n.a.	n.a.	Product / Project Scope	n.a.			

Id	Disguised Source	Issue	Product Design / Life cycle Phase	Aspect Classification	Sorting Classification	Company Function	Scope Classification	Modular System Design / Life cycle Phase	Important Factor	Important Support	Contradictory to ...
92	Secondary case	Not enough time and lengthy discussions about modularisation decisions causes frustrated team	Diverse	Process	n.a.	n.a.	Product / Project Scope	n.a.			
93	Primary Case B	PRM, Engineers and Managers found effort during discussions not acceptable.	Diverse	Process	n.a.	Diverse	Product / Project Scope	n.a.	14	13	
94	Primary Case B	People in projects always forget the big picture about modularisation (e.g. better life cycle characteristics, reduced overall cost through better reuse)	Diverse	Process	n.a.	Diverse	Product / Project Scope	n.a.			
95	Primary Case B	Compared to previous projects, in the extended market phase engineers miss samples, experiments and drawings by only discussing spreadsheets	Diverse	Process	n.a.	R & D	Product / Project Scope	n.a.			
96	Primary Case B	It is more in the nature of PRMs and engineers to avoid clear decisions as the future is unknown but to go for a concept and test it without high investment of modularisation	Diverse	Process	n.a.	Diverse	Product / Project Scope	n.a.			
97	Primary Case B	High effort and discussions about details during modularisation method causes high frustration over time.	Diverse	Process	n.a.	Diverse	Product / Project Scope	n.a.			
98	Primary Case B	Modularisation means front-loading to fuzzy front-end	Diverse	Process	n.a.	Diverse	Product / Project Scope	n.a.			
99	Primary Case B	It is not possible to run through a modularisation methodology subsequently => it has to be done in iterative cycles over time	Diverse	Process	n.a.	n.a.	n.a.	n.a.			
100	Primary Case B	Transitioning is a marathon, not a short-distance race	Diverse	Introduction	Change Management	Diverse	n.a.	n.a.	25		
101	Secondary case	Modularisation solely done by engineers does not work	Diverse	Process	n.a.	Diverse	n.a.	n.a.	27	27	
102	Secondary case	Looking back, companies have always created variance across projects without reason for more than 50 % of it.	Diverse	Process	n.a.	R & D	Product / Project Scope	n.a.	28	28	

Id	Disguised Source	Issue	Product Design / Life cycle Phase	Aspect Classification	Sorting Classification	Company Function	Scope Classification	Modular System Design / Life cycle Phase	Important Factor	Important Support	Contradictory to ...
103	Secondary case	Modularisation is not a self-runner with cause and investments today and benefits in future	Diverse	Evaluation	Financial	Diverse	n.a.	n.a.			
104	Secondary case	Modularisation is an enabler for product configuration: "My engineering know-how cannot be made configurable"	Diverse	Introduction	n.a.	R & D	n.a.	n.a.			
105	Secondary case	Engineers or business units always argue that they are doing this already.	Diverse	Introduction	Change Management	Diverse	n.a.	n.a.			
106	Secondary case	Organisational barriers between PRM and ENG.	Diverse	Organisation	n.a.	Diverse	n.a.	n.a.			
107	Secondary case	There is no lack of method in product development, to live the concept sustainably there is lack of time, information availability and motivation.	Diverse	n.a.	Change Management	Diverse	n.a.	n.a.			
108	Secondary case	Issue with methods: very time consuming and very complex, though, complete procedure	Diverse	Process	n.a.	Diverse	Modular System Scope	Developing Modular System			
109	Secondary case	People are not convinced about the importance for modularisation (unsexy, not enough time, capacity)	Diverse	n.a.	Change Management	Diverse	n.a.	n.a.	41	60	
110	Primary case C	Viewpoint of designer: there are other, much more important goals than modularisation and commonality: better to come up with a 70 % solution and to bring it onto the market quickly, modularisation takes too much time for it, we only can afford to look ahead for 1-2 years	Diverse	Process	n.a.	Diverse	Modular System Scope	Developing Modular System			
111	Primary case C	If there is not enough market and technical knowledge, it is the better option not to modularise the product immediately	Diverse	Introduction	n.a.	Diverse	n.a.	n.a.	71	68	

Id	Disguised Source	Issue	Product Design / Life cycle Phase	Aspect Classification	Sorting Classification	Company Function	Scope Classification	Modular System Design / Life cycle Phase	Important Factor	Important Support	Contradictory to …
112	Primary case C	Front-loading, broad scope, etc. is good, but we do not have the time for it if we have to fulfil all other steps of the time to market process and if we shall come up with more products every two years => modularisation is detrimental during transitioning	Diverse	Introduction	n.a.	Diverse	n.a.	n.a.			
113	Primary case C	Modularisation with our approach brings a lot of new documents => 80% of them are just filled because they have to be filled => researchers do always add something on top, instead they should seek out to reduce less important work packages for engineers!	Diverse	n.a.	n.a.	Diverse	n.a.	n.a.			
114	Primary case C	Approach of primary case C modularisation crashed: Wrong approach => you cannot start in a project with trouble from all sides with modularisation by introducing consultants and a few templates without any other changes	Diverse	n.a.	n.a.	Diverse	n.a.	n.a.			
115	Primary case C	We were always lost in a lot of detailed stuff that actually had nothing to do with modularisation (e.g. feature tree, QFD, … it took us 1000 y to come up with feature tree before it was spoilt again by product management). We were lost in the method instead of in working on the modules or coming to the modules (still knowing that this work is very important)	Diverse	Process	n.a.	Diverse	n.a.	n.a.			
116	Primary case C	Product architecture design has to be better transferred into organisation => design engineer heard 2 weeks before module driver analysis about the concept of product architecture design	Diverse	Introduction	Change Management	Diverse	n.a.	n.a.			
117	Primary case C	At primary case C after establishing product architecture, we actually needed 150% to design products for the next trade fair, but actually we also needed much more resources for modular system => we de not have that resources	Diverse	Introduction	n.a.	R & D	n.a.	n.a.			

Id	Disguised Source	Issue	Product Design / Life cycle Phase	Aspect Classifi-cation	Sorting Classifica-tion	Company Function	Scope Classifica-tion	Modular System De-sign / Life cycle Phase	Impor-tant Factor	Impor-tant Support	Contra-dictory to ...
118	Primary case C	In recent times, too much workload on engineers that does not contribute to direct design work. E.g. FMEA, DFMA, Engineering system, time to market process, product development processes. Modularisation adds to that while engineers are overwhelmed, resistant and try to make short-cuts wherever possible.	Diverse	Introduc-tion	n.a.	R & D	n.a.	n.a.			
119	Primary case C	Fear during project to have invested a lot in modularisation and afterwards modularisation doesn't work properly.	Diverse	n.a.	n.a.	Diverse	n.a.	n.a.			
120	Primary case C	Architecting is like looking in a crystal ball, that's why we didn't want to reduce too much and always came up with too many module variants that did not mean any part number reduction or increase in volume!	Diverse	n.a.	n.a.	Diverse	n.a.	n.a.			
121	Primary case C	Primary case C: Lack of resources, knowledge, motivation at project team	Diverse	Introduc-tion	n.a.	R & D	Modular System Scope	n.a.			
122	Primary case C	PRM: Much more time needed during modular system project compared to single product development project	Diverse	n.a.	n.a.	Diverse	Modular System Scope	n.a.			
123	Primary case A	Everything you measure with modularisation is only platform related and not directly cost-related! (compared to benchmarking, etc.)	Diverse	Evalua-tion	Financial	Diverse	Modular System Scope	n.a.			
124	Primary case A	Too high expectations very risky (e.g. too many different technologies in one modular system)	Diverse	n.a.	n.a.	Diverse	Modular System Scope	Planning Modular System			
125	Primary case A	Wanting too much too fast is very risky together with the situation that the organisation does not fit the modular structure	Diverse	Organisa-tion	n.a.	Diverse	Modular System Scope	Planning Modular System			
126	Primary case A	Applying methods is even dangerous, they suggerate that it is enough to apply the method, but other things are more important and applying a method is alone quite expensive	Diverse	Process	n.a.	Diverse	Modular System Scope	n.a.			

Id	Disguised Source	Issue	Product Design / Life cycle Phase	Aspect Classification	Sorting Classification	Company Function	Scope Classification	Modular System Design / Life cycle Phase	Important Factor	Important Support	Contradictory to ...
127	Primary case A	DEV doesn't want to have modular systems, so they are reluctant, modular systems are driven by sales, PRM and production => DEV has to be convinced/pulled	Diverse	n.a.	n.a.	R & D	Modular System Scope	n.a.			
128	Secondary case	Platforming is very expensive and it is a gamble against the future!	Diverse	n.a.	n.a.	Diverse	Modular System Scope	n.a.			
129	Primary case A	Decisions of PBs are only accepted or lived if they really come from PBs or sites, general rejection of ideas from other areas! => change of culture that goes beyond modularisation	Diverse	n.a.	Change Management	Diverse	Modular System Scope	n.a.			
130	Secondary case	Not clear when platform development is finished and when variant projects start	Diverse	Process	n.a.	R & D	Modular System Scope	Developing Modular System			
131	Primary case C	If product is placed over modular system, the whole concept does not work properly	Diverse	n.a.	n.a.	Diverse	Product / Project Scope	n.a.			
132	Secondary case	Prevailing goals of company are totally different than that with modular system development => sort this out!	Diverse	Evaluation	n.a.	Diverse	Product / Project Scope	n.a.			
133	Secondary case	Very difficult for ENG to get appropriate data for modularisation	Diverse	IT	n.a.	R & D	Modular System Scope	n.a.			
134	Primary case C	If one modularises like at primary case C, there are not enough information and too many uncertaintes => if one would modularise in an own process, such problems would not exist; the main issue when developing a platform is to have the right information at the right time (but that is hardly possible with primary case C approach)	Diverse	Process	n.a.	Diverse	Modular System Scope	n.a.			
135	Primary case A	Engineers said that they are already modular and that they are already very good	Diverse	n.a.	System/Product Architecting	R & D	n.a.	n.a.			

Id	Disguised Source	Issue	Product Design / Life cycle Phase	Aspect Classifi-cation	Sorting Classifica-tion	Company Function	Scope Classifica-tion	Modular System De-sign / Life cycle Phase	Impor-tant Factor	Impor-tant Support	Contra-dictory to ...
136	Primary case A	"Measurement is past-oriented" => measure both: seeding and harvesting!	Diverse	Evalua-tion	n.a.	Diverse	n.a.	n.a.			
137	Primary case A	Unless there are no radical organisational changes, engi-neers at each loation work on their own (e.g. high-tech for Germany, low-cost for Turkey, small ones for UK) and create own versions => very difficult to merge these concepts into the same modular system.	Diverse	Organisa-tion	n.a.	R & D	Product / Project Scope	n.a.			
138	Primary case A	Organisation does not understand that it is important to put much more effort into frontloading => prioritisation, maybe other projects have to be pruned for that	Diverse	n.a.	Change Manage-ment	Diverse	Modular System Scope	n.a.			
139	Primary case B	Modular system displayed in a morphological box is maybe a bit naive if people think that it works exactly that way.	Diverse	Introduc-tion	Change Manage-ment	Diverse	Modular System Scope	n.a.			
140	Primary case B	Where to establish modularisation roles? - if we establish them on a high level: more power, more overview - if they are on a lower level, they have a much more detailed knowledge - central department: ressources available, no overload, but no "doers" and no power on projects, no detailed knowledge & resolve the organisational issue between single product development and platform development	Diverse	Organisa-tion	n.a.	Diverse	Modular System Scope	n.a.			
141	Primary case A	Even if there is top management support for modularisation, engineers that have the detailed knowledge will not support modularisation.	Diverse	n.a.	n.a.	R & D	Modular System Scope	n.a.			
142	Primary case A	Applying KPIs is just too much effort	Diverse	Evalua-tion	n.a.	Diverse	Modular System Scope	n.a.			

Id	Disguised Source	Issue	Product Design / Life cycle Phase	Aspect Classifi-cation	Sorting Classifica-tion	Company Function	Scope Classifica-tion	Modular System De-sign / Life cycle Phase	Impor-tant Factor	Impor-tant Support	Contra-dictory to ...
143	Secondary case	Consultancies and institutes failed to recognize that users have to understand it, that systems in big companies are hard to change and that ordinary engineers have to under-stand it.	Diverse	Introduc-tion	Change Manage-ment	Diverse	Modular System Scope	n.a.			
144	Secondary case	They always talk about reuse, modularisation and platforms, but they never make a bigger action out of that (as example: automotive industry was mentioned.)	Diverse	Introduc-tion	n.a.	Top Manage-ment	n.a.	n.a.			
145	Secondary case	Software platform can be used across many different prod-ucts. BUT - in the past, the code for each product was very simple. Today it is bigger and more complex => for single projects, there is no use in developing and even using the platform as it generates more complexity for a single project => more rules have to be considered for development and usage, and more coordination and communication has to be considered. (see research note, research log book 19.12.2013)	Diverse	Diverse	Sys-tem/Produc t Archi-tecting	R & D	n.a.	n.a.			
146	Secondary case	Bringing products to the market is always more important than the overall modular system ("this is waste of my precious time", "give me a proof if this really pays off")	Diverse	Diverse	n.a.	Diverse	n.a.	n.a.			
147	Primary case A	What we found out at the primary case company is that we cannot reduce complexity => we can only slow down com-plexity growth => that causes lot of frustration in companies even tougher: eventually even more parts with modularisation (new modules to be developed in parallel, new virtual items, less integration); and the problem is with measurement that modularisation transition is costly without generating benefits at the transition phase!	Diverse	Diverse	n.a.	Diverse	Modular System Scope	Diverse			

Id	Disguised Source	Issue	Product Design / Life cycle Phase	Aspect Classification	Sorting Classification	Company Function	Scope Classification	Modular System Design / Life cycle Phase	Important Factor	Important Support	Contradictory to ...
148	Primary case A	Reasons for focus primary case shifting away from modularisation: - well known: time-pressure, cost, last-minute wishes of marketing, ... - primary case Cfting of top management's priority: New CEO who came into position and put focus rather on Innovation, passion for engineering, quality, lean processes, ... than on variant management/standardisation/modularisation => that is some kind of unpopular or unsexy, not spectacular or sexy enough.	Diverse	Diverse	n.a.	Top Management	Modular System Scope	Diverse			
149	Primary case C	In the approach we took, there are very limited ressources in manpower to set up and control the architectural design rules from the very beginnign until later stages and maintenance.	Diverse	Diverse	n.a.	R & D	Modular System Scope	Diverse			
150	Secondary case	Product Structure in PDM: - Every project creates its own product structure (it is started with Israel, Norway starts with its own structure afterwards). - Work products are identified over project-dependent SNRs, if they are reused for another project, if at all, they will be reused with another SNR => no central elements!	Diverse	IT	n.a.	R & D	Modular System Scope	Diverse	57	130	
151	Secondary case	Time: - Project should use generic module XY, but they don't have the time to wait until the generic modules are ready	Diverse	Process	n.a.	R & D	Modular System Scope	Diverse			
152	Secondary case	Architecture thinking only in products/ projects and not overarching!	Diverse	Organisation	n.a.	Diverse	Product / Project Scope	n.a.			
153	Primary case A	Project progress was slowed down due to extra-work due to data structure not fitting to analysis (e.g. feature list and BOM data)	Diverse	IT	n.a.	Diverse	Modular System Scope	Diverse			

Id	Disguised Source	Issue	Product Design / Life cycle Phase	Aspect Classification	Sorting Classification	Company Function	Scope Classification	Modular System Design / Life cycle Phase	Important Factor	Important Support	Contradictory to ...
154	Secondary case	The knowledge of system architecture is too often limited to a few of domain experts or even totally unknown => it must be the goal to make things explicit	Diverse	IT	n.a.	R & D	Modular System Scope	Diverse			
155	Primary case C	Arguments for reluctance of development team: - it is in our heads already, we don't need a huge method and templates - time - we don't know it yet - platforming on a broader scope always failed because it cannot be handled in parallel to a higher prioritized variant project - lack of collaboration between different units for broader platform	Diverse	Diverse	n.a.	Diverse	Modular System Scope	Diverse			
156	Primary case A	Traditionally products have been developed for the biggest market and on this basis individualisation has taken place.	Diverse	Diverse	n.a.	Diverse	Modular System Scope	Planning Modular System			
157	Secondary case	New functionality was actually to be developed for generic products. => Developers brought in too many bugs during designing as there was not sufficient knowledge about the new functionality => delivery date for the first customer approached "quicker than expected" => time pressure and even more bugs to be fixed => lack of time, resources and knowledge => highest priority of management to deliver as soon as possible => low motivation to pursue generic development approach => new functionality was only developed for first customer and application, not for the originally scope of products => dramatic failure of modular system development with high waste of resources	Diverse	Diverse	n.a.	R & D	Modular System Scope	Developing Modular System			
158	Primary Case B	Constant fear of missing cost targets for first single products endanger overall picture	Finance	Process	n.a.	R & D	Product / Project Scope	n.a.			

Id	Disguised Source	Issue	Product Design / Life cycle Phase	Aspect Classifi-cation	Sorting Classifica-tion	Company Function	Scope Classifica-tion	Modular System De-sign / Life cycle Phase	Impor-tant Factor	Impor-tant Support	Contra-dictory to ...
159	Primary Case B	Modularisation only contributes to single unit cost reduction through volume effect, e.g. in purchasing or production	Finance	Process	n.a.	Diverse	Product / Project Scope	n.a.			
160	Secondary case	Benefit through modularisation doesn't occor before third or fourth project (PLM: 3rd reuse).	Finance	n.a.	Financial	Diverse	n.a.	n.a.			
161	Primary case C	After running the modularisation project for 1-2 years, management wants to see first financial results, but at that time they are of course disastrous. Hence, they get impa-tient.	Finance	Evalua-tion	Financial	Diverse	Modular System Scope	n.a.			
162	Primary case A	Comparison to benchmark office: Benchmark analysis on product level contradicts with standardisation and modu-larisation; the definitive advantage of benchmarking is that the monetary potential can be directly related to a concrete product! (This is not the case with complexity cost applied with modularisation)	Finance	Evalua-tion	Financial	Diverse	Modular System Scope	n.a.			
163	Primary case B	Nobody wants to pay for additional effort for modularisa-tion. Project itself has no benefit and central department does not have any budget to support projects concerning modularisation. Will there be a budget for anywhere central -> only for implementation activities, but not for necessary project work. - central engineering department - business unit - development site - development project	Finance	n.a.	Financial	n.a.	Modular System Scope	n.a.			
164	Secondary case	Complexity costs are useless for us as they are only virtual. Engineers say, if you give me the complexity cost in cash, I will reduce complexity.	Finance	Evalua-tion	Financial	R & D	Modular System Scope	n.a.			

Id	Disguised Source	Issue	Product Design / Life cycle Phase	Aspect Classification	Sorting Classification	Company Function	Scope Classification	Modular System Design / Life cycle Phase	Important Factor	Important Support	Contradictory to ...
165	Secondary case	In most companies which we analysed, modularisation was not started from scratch but based on an existing product or platform (e.g. secondary case) => no real benchmark for our means of transition	n.a.	Introduction	n.a.	n.a.	n.a.	n.a.			
166	Secondary case	Engineers are reluctant to new platform concepts as they think that their previous work has not been good enough	n.a.	Introduction	Change Management	n.a.	n.a.	n.a.			

The following table shows an export of the coding database with a special focus on important factors for modularisation transition.

Table D – II: Export of coded important factors

Id	Detailed Source	Important Factor	Product Design / Life cycle Phase	Aspect Classification	Sorting Classification	Company Function	Scope Classification	Modular System Design / Life cycle Phase	Issue	Important Support	Contradictory to
1	diverse	There must be no constraints (e.g. capacity, knowledge) for extended requirement engineering	PLC - Product Development: Requirements Phase	Process	n.a.	Marketing / Product Management	Modular System Scope	Planning Modular System	1	1	
2	diverse	There must be enough resources to test customer appeal of new reuse modules/products during modularisation project	PLC - Product Development: Requirements Phase	Process	n.a.	Marketing / Product Management	Modular System Scope	Planning Modular System	4	8	
3	diverse	It must be told to the customer/sales what is feasible within the boundaries of the product architecture and what not => Deviation is only possible with additional costs	PLC - Product Development: Requirements Phase	Process	n.a.	Diverse	Modular System Scope	Diverse			
4	diverse	Modularisation process has to consider architecture goals (e.g. module drivers, variance vs. Standard, PLC issues)	PLC - Product Development: Concept Phase	Process	System/Product Architecting	Diverse	Modular System Scope	Planning Modular System			
5	diverse	Experienced engineers have to take part in module grouping.	PLC - Product Development: Concept Phase	Process	System/Product Architecting	R & D	Modular System Scope	Planning Modular System			
6	diverse	Right size of modules is important: If we have too many parts in a module, we will get too many module variants. If we have too small modules, we are not efficient enough.	PLC - Product Development: Concept Phase	Process	System/Product Architecting	R & D	Modular System Scope	Planning Modular System			

Id	Detailed Source	Important Factor	Product Design / Life cycle Phase	Aspect Classifi-cation	Sorting Classifica-tion	Company Function	Scope Classifica-tion	Modular System De-sign / Life cycle Phase	Issue	Impor-tant Support	Contra-dictory to
7	diverse	Artefacts have to be separated into modules, module vari-ants and interfaces.	PLC - Product Development: Concept Phase	Process	Sys-tem/Product Archi-tecting	R & D	Modular System Scope	Diverse	23	24	
8	diverse	Parts of the product that are linked to unstable customer requirements can be clustered into modules later on or remain outside the modular system	PLC - Product Development: Concept Phase	Process	Sys-tem/Product Archi-tecting	R & D	Modular System Scope	Planning Modular System			
9	diverse	Organisational barriers between PRM and ENG have to be broken through	PLC - Product Development: Concept Phase	Process	n.a.	Diverse	Modular System Scope	Planning Modular System			
10	diverse	Decision for degree of modularity with own strength and weaknesses	PLC - Product Development: Concept Phase	Process	Sys-tem/Product Archi-tecting	Diverse	Modular System Scope	Planning Modular System	61	56	
11	diverse	Even though there are many discussions at the very begin-ning, it is better to fix issues at the beginning instead of having them during design phase (but see issues during design phase, further investigate this factor)	PLC - Product Development: Concept Phase	Process	n.a.	Diverse	Modular System Scope	Diverse			
12	diverse	Product structuring during modularisation method has to be more seen as check than as means to setting up the struc-ture => in such a way it might be helpful	PLC - Product Development: Concept Phase	Process	Sys-tem/Product Archi-tecting	R & D	Modular System Scope	Planning Modular System	80	78	
13	diverse	The product architecture with its common, variant and optional elements has to be identified prior to the system design review.	PLC - Product Development: Concept Phase	Process	n.a.	Diverse	Modular System Scope	Diverse			
14	diverse	Interfaces between variant, common and optional configura-tion items have to be linked and to be common => Their description has to be attached or linked together with the functional description to all items they belong to	PLC - Product Development: Concept Phase	Diverse	n.a.	R & D	Modular System Scope	Diverse			

Id	Detailed Source	Important Factor	Product Design / Life cycle Phase	Aspect Classifi-cation	Sorting Classifica-tion	Company Function	Scope Classifica-tion	Modular System De-sign / Life cycle Phase	Issue	Impor-tant Support	Contra-dictory to
15	diverse	Issues between design and manufacturing BOM have to be resolved.	PLC - Product Development: Design Phase	IT	n.a.	R & D	Modular System Scope	Diverse			
16	diverse	Modular system strategy follows a descending approach that aims first at controlling (top-down) the building block's specifications	PLC - Product Development: Design Phase	Process	n.a.	n.a.	Modular System Scope	Diverse			
17	diverse	There must be enough time and resources to test whether newly created modules for the modular system contribute to fulfilment of customer needs as intended	PLC - Product Development: Testing	Process	n.a.	Marketing / Product Management	Modular System Scope	Planning Modular System		9	
18	diverse	There must be enough time and resources to test whether intended technical concepts for modules work as expected	PLC - Product Development: Testing	Process	n.a.	R & D	Modular System Scope	Diverse			
19	diverse	There must be enough room to develop the modular system gradually with customer approval and test of technical feasibility.	PLC - Product Development: Testing	Process	n.a.	Diverse	Modular System Scope	Diverse			
20	diverse	For modularisation, we first need several solutions (concep-tual study) to make decisions on that, a) that we do not overdimension modules and b) that we do not run into the danger than it won't work afterwards	PLC - Product Development: Testing	Process	n.a.	R & D	Modular System Scope	Developing Modular System			
21	diverse	Detailing the modular system (architecture, interfaces) not possible before first samples for all modules exist! (Either this is done centrally or gradually, while there is the prob-lem with gradually that the devil is in the detail and, thus, problems arise too late!	PLC - Product Development: Testing	Process	n.a.	R & D	Modular System Scope	Developing Modular System	91	87	
22	diverse	There must be mechanisms to control variance restrictions according to the plan	Evolution of Products, Reuse, Design Modifica-tions, Engineer-ing Change	Diverse	n.a.	Diverse	n.a.	Diverse	2	2, 3	

Id	Detailed Source	Important Factor	Product Design / Life cycle Phase	Aspect Classifi-cation	Sorting Classifica-tion	Company Function	Scope Classifica-tion	Modular System De-sign / Life cycle Phase	Issue	Impor-tant Support	Contra-dictory to
23	diverse	It must be evaluated whether the modular system is devel-oped according to the plans	Evolution of Products, Reuse, Design Modifica-tions, Engineer-ing Change	Diverse	n.a.	R & D	Modular System Scope	Diverse	2	4	
24	diverse	It must not be easier to create new module variants than to reuse existing module variants	Evolution of Products, Reuse, Design Modifica-tions, Engineer-ing Change	Process	n.a.	R & D	Modular System Scope	Diverse	3	7	
25	diverse	Separate process for interface and module management has to be established	Evolution of Products, Reuse, Design Modifica-tions, Engineer-ing Change	Process	n.a.	R & D	Modular System Scope	Diverse	24	25	
26	diverse	As architecture will be continuously changed, there has to be product architecture change process	Evolution of Products, Reuse, Design Modifica-tions, Engineer-ing Change	Process	n.a.	R & D	Modular System Scope	Maintenance of Modular System			
27	diverse	Complexity has to hurt at the point of creation	Evolution of Products, Reuse, Design Modifica-tions, Engineer-ing Change	Evalua-tion	n.a.	Diverse	n.a.	n.a.			
28	diverse	Modules have to be developed independently, free from product development projects under pressure.	Evolution of Products, Reuse, Design Modifica-tions, Engineer-ing Change	Process	n.a.	R & D	Modular System Scope	Developing Modular System			

Id	Detailed Source	Important Factor	Product Design / Life cycle Phase	Aspect Classification	Sorting Classification	Company Function	Scope Classification	Modular System Design / Life cycle Phase	Issue	Important Support	Contradictory to
29	diverse	Each new version to be created on the common architecture shall be subject to be reviewed for consistency with the architecture. If there is any deviation to plans, creation must be approved by modular system authorities, CCB and top-management.	Evolution of Products, Reuse, Design Modifications, Engineering Change	Process	n.a.	Diverse	Modular System Scope	Diverse			
30	diverse	When a change affects a shared item of the modular system, it shall be processed via an ECR by the product-line committee and communicated to all change committees in the projects working with the modular system => Always with "where used" and impact analysis	Evolution of Products, Reuse, Design Modifications, Engineering Change	Process	n.a.	R & D	Modular System Scope	Maintenance of Modular System			
31	diverse	Single projects must have an incentive to stick to the module roles	Diverse	Diverse	n.a.	Diverse	Modular System Scope	Diverse	2	5	
32	diverse	There must be a separate modular system development process "over" product development projects	Diverse	Process	n.a.	Diverse	Modular System Scope	Diverse	8,9,10		
33	diverse	Process how the modular system shall evolve must be available (e.g. see gradual evolvement of MQB)	Diverse	Process	n.a.	Diverse	Modular System Scope	Diverse	8,9,10		
34	diverse	There must be process-integration of modularisation relevant activities.	Diverse	Process	n.a.	Diverse	Modular System Scope	Diverse	8,9,10		
35	diverse	Modular system must be developed centrally from an organisational perspective	Diverse	Organisation	n.a.	Diverse	Modular System Scope	Diverse	8,9,10		
36	diverse	Modularisation must not generate additional workload to the existing organisation without compensation for it	Diverse	Diverse	n.a.	Diverse	Modular System Scope	Diverse	12	13	
37	diverse	Involved managers and engineers must have constantly transparency about the big picture	Diverse	IT	n.a.	Diverse	Modular System Scope	Diverse			
38	diverse	The overall performance of the modular system must be constantly evaluated against the trap of short-term, isolated goals	Diverse	Evaluation	n.a.	Diverse	Modular System Scope	Diverse			

Id	Detailed Source	Important Factor	Product Design / Life cycle Phase	Aspect Classification	Sorting Classification	Company Function	Scope Classification	Modular System Design / Life cycle Phase	Issue	Important Support	Contradictory to
39	diverse	Important decision about technical concept has to be made early on for all modules - in order to fix interfaces	Diverse	Process	n.a.	Diverse	Modular System Scope	Planning Modular System			
40	diverse	A stable module base (platform??) has to be given before a product development project starts.	Diverse	Process	n.a.	R & D	Modular System Scope	Developing Modular System			
41	diverse	Base for product has to be kept stable and has to be frozen timely.	Diverse	Process	n.a.	R & D	Modular System Scope	Diverse			
42	diverse	Modularisation needs an overall programme/initiative throughout the company	Diverse	Introduction	n.a.	Diverse	Modular System Scope	Diverse			
43	diverse	All company people have to be involved in modularisation	Diverse	Introduction	n.a.	Diverse	Modular System Scope	Diverse	27	27	
44	diverse	Introducing modularisation has to be done very strictly and stringent. No exceptions for local "kings" allowed. However, be aware of their arguments as they may be valid (e.g. "detailed knowledge")	Diverse	Introduction	n.a.	Diverse	Modular System Scope	Diverse	28	28	
45	diverse	Central module and interface responsible have to gain power over local and project responsibles	Diverse	Organisation	n.a.	Diverse	Modular System Scope	Diverse			
46	diverse	It is important to transition slowly and gradually without overwhelming organisation.	Diverse	Introduction	n.a.	Diverse	Modular System Scope	Diverse			
47	diverse	Culture and understanding for complexity management has to be established.	Diverse	Introduction	n.a.	Diverse	Modular System Scope	Diverse			
48	diverse	Engineers have to know that complexity is directly and permanently evaluated.	Diverse	Evaluation	n.a.	R & D	Modular System Scope	Diverse			
49	diverse	Establish responsibilities and evaluations in a way so that single products will not win over product portfolio.	Diverse	Diverse	n.a.	Diverse	Modular System Scope	Diverse	34	40	

Id	Detailed Source	Important Factor	Product Design / Life cycle Phase	Aspect Classification	Sorting Classification	Company Function	Scope Classification	Modular System Design / Life cycle Phase	Issue	Important Support	Contradictory to
50	diverse	Modularisation has to be constantly pulled.	Diverse	Diverse	n.a.	Diverse	Modular System Scope	Diverse			
51	diverse	Support to foster data/information availability, time and motivation are vital for modularisation.	Diverse	Diverse	n.a.	Diverse	Modular System Scope	Diverse			
52	diverse	Understanding , support and evaluation/pull of management OR financial evaluation	Diverse	Diverse	n.a.	Diverse	Modular System Scope	Diverse	59	55	
53	diverse	Coordination of architecture between different projects	Diverse	Process	n.a.	Diverse	Modular System Scope	Diverse	60	55	
54	diverse	People have to be convinced about the importance of modularisation	Diverse	Introduction	Change Management	Diverse	n.a.	n.a.	63	60	
55	diverse	Modularisation requires enough knowledge about market and technology in order to come up with the required pace	Diverse	Process	n.a.	Diverse	Modular System Scope	Diverse	71	68	
56	diverse	Simplify and make things leaner instead of always placing something on top!	Diverse	Introduction	Change Management	Diverse	n.a.	n.a.			
57	diverse	Raise importance of product architecture documents that are the only documents that glue the overall view together.	Diverse	Diverse	n.a.	Marketing / Product Management	Modular System Scope	Diverse			
58	diverse	Motivation of single engineers, motivation of management to drive the overall modular system	Diverse	Diverse	Change Management	Diverse	Modular System Scope	Diverse			

Id	Detailed Source	Important Factor	Product Design / Life cycle Phase	Aspect Classifi-cation	Sorting Classifica-tion	Company Function	Scope Classifica-tion	Modular System De-sign / Life cycle Phase	Issue	Impor-tant Support	Contra-dictory to
59	diverse	General success factors: - Management commitment -> Pull from Management - Team-oriented approach (interdisciplinary - Availability of sufficient resources concerning time and know-how, monetary budget - Creativity - Integration of tools and methods into existing processes - Benefit management and realisation - Effective communication of reasons for change - Effective training scheme for organisation	Diverse	Diverse	Change Manage-ment	Diverse	Modular System Scope	Diverse			
60	diverse	Established measurement of what is good and what is bad modularisation.	Diverse	Evalua-tion	n.a.	Diverse	Modular System Scope	Diverse			
61	diverse	Establish clear decisions for modules: This is our standard module for this function. (modularisation doesn't work if all exra-wishes are tried to be met)	Diverse	Diverse	Standardisa-tion	R & D	Modular System Scope	n.a.			
62	diverse	Everything has to be considered that it comes down to BOMs (e.g. view on product structure)	Diverse	IT	n.a.	R & D	Modular System Scope	n.a.	134	124	
63	diverse	It is important to have top-down and central architecture with project-neutral elements	Diverse	IT	n.a.	R & D	Modular System Scope	Diverse	151	130	
64	diverse	Pursuing a product line approach is a strategic decision - a product line should not be pursued for a single project => it must be a formal decision by company management	Diverse	Introduc-tion	n.a.	Diverse	Modular System Scope	Diverse			
65	diverse	In a product line approach, Configuration Management shall be established for the modular system instead of for the project	Diverse	Introduc-tion	n.a.	Diverse	Modular System Scope	Diverse			
66	diverse	Development documents should be structured to suit the needs of the overall product line => modularly divide into common, optional and variant parts	Diverse	Introduc-tion	n.a.	Diverse	Modular System Scope	Diverse			

Id	Detailed Source	Important Factor	Product Design / Life cycle Phase	Aspect Classifi-cation	Sorting Classifica-tion	Company Function	Scope Classifica-tion	Modular System De-sign / Life cycle Phase	Issue	Impor-tant Support	Contra-dictory to
67	diverse	Establish a change control board to manage the divergence between the product line and products/projects - change affecting the common product architecture - communication to stakeholders	Diverse	Organisa-tion	n.a.	Diverse	Modular System Scope	Diverse			
68	diverse	The product line responsible should have enough power and authority against projects/products. It should be on a higher hierarchy level.	Diverse	Organisa-tion	n.a.	Diverse	Modular System Scope	Diverse			
69	diverse	Compatibilities and constraints between option and variant modules have to be determined and attached to the highest configuration item	Diverse	IT	n.a.	R & D	Modular System Scope	Diverse			
70	diverse	Module variants shall be managed like a "normal" product (e.g. with its own approved documentation and its own Baselines)	Diverse	Diverse	n.a.	R & D	Modular System Scope	Diverse			
71	diverse	Decide what to do when a project cannot wait until the development of a common module: - either have a temporary (alignment later on) deviation from the modular system or a lasting lower level of modu-larity	Diverse	Process	n.a.	R & D	Modular System Scope	Diverse			
72	diverse	Overall LL - Not only concentrate on single platform projects but widen approach towards strategic positioning of product portfolio and prioritization of innovation resources - Include value chain and IT architecture to ensure an holistic approach and full benefit - Get a clear top down commitment based on: a) the mutual understanding that complexity management by modularization is an important enabler for the com-pany's competitiveness b) an overall strategy with a clear picture for the mid and long-term approach	Diverse	Diverse	n.a.	n.a.	Modular System Scope	Diverse			

Id	Detailed Source	Important Factor	Product Design / Life cycle Phase	Aspect Classifi-cation	Sorting Classifica-tion	Company Function	Scope Classifica-tion	Modular System De-sign / Life cycle Phase	Issue	Impor-tant Support	Contra-dictory to
73	diverse	Exclude products with fixed delivery schedules from domain engineering (in PLE approach)	Diverse	Introduc-tion	n.a.	n.a.	n.a.	n.a.			
74	diverse	- Full re-allocation of responsibility and monetary budget is needed - Experience and knowledge is needed for domain engineer-ing - After concept phase, product architecting is by no means completed - Consider the way how the products will evolve over time - Don't design an architecture that shall cover everything and hence, are too costly or do not provide any concrete support - close collaboration between platform and application engineering (in order to avoid double work in both disci-plines) - Don't be too optimistic on actual skills and practices and jump from single product development toward development of modular system at once => give it more time	Diverse	Diverse	n.a.	n.a.	n.a.	n.a.			
75	diverse	There is a real trade-off to make between a too detailed reference architecture that would bring unjustified con-straints on future projects and a too open one that would bring only little added value.	Diverse	Diverse	n.a.	n.a.	n.a.	n.a.			
76	diverse	One ground-breaking modularisation project is needed that pulls the organisation toward attention for modularisation. => Everything that makes marketing for modularisation is helpful	Diverse	Introduc-tion	n.a.	n.a.	n.a.	n.a.			
77	diverse	In order to avoid problems, we have to quickly come from solution-neutral space to technical concept, even though this is very difficult for platforms	Diverse	Process	n.a.	n.a.	Modular System Scope	Diverse			

Id	Detailed Source	Important Factor	Product Design / Life cycle Phase	Aspect Classifi-cation	Sorting Classifica-tion	Company Function	Scope Classifica-tion	Modular System De-sign / Life cycle Phase	Issue	Impor-tant Support	Contra-dictory to
78	diverse	- Targets have to be realistic for development later on. If these targets cannot be met, we have to start at the beginning with modular system development - All project team members and other impacted personnel have to understand the decisions that are made => Communicate and fix realistic decisions, document reasoning, otherwise extra wishes come up again and again	Diverse	Diverse	n.a.	n.a.	Modular System Scope	Diverse			

The following table shows an export of the coding database with a special focus on support for modularisation transition.

Table D – III: Export of coded modularisation transition support

Id	Disguised Source	Support	Product Design / Life cycle Phase	Aspect Classification	Sorting Classification	Company Function	Scope Classification	Modular System Design / Life cycle Phase	Issue	Important Factor	Contradictory to
1	Secondary case	Devoted process phase / organisation for extended requirements engineering	PLC - Product Development: Requirements Phase	Process	n.a.	n.a.	Modular System Scope	Planning Modular System	1	1	
2	Primary case C	At Primary Case C we spent most of the time figuring out which features to include into the modular system=> we did this with feature trees and estimated sales volume	PLC - Product Development: Requirements Phase	Process	n.a.	Marketing / Product Management	Modular System Scope	Planning Modular System			
3	Primary case C	Market volumes & target costs for features and modules; simulations for mid- and top-segment	PLC - Product Development: Requirements Phase	Process	n.a.	Marketing / Product Management	Modular System Scope	Planning Modular System			
4	Primary case C	Important to document restrictions of variety so that management cannot come up later on with wishes for product variance	PLC - Product Development: Requirements Phase	Process	Variant Management	Diverse	Modular System Scope	Planning Modular System			
5	Primary case C	Part number reduction comes close with modularisation and reduction of variants which needs strong discussions with PRM and sales	PLC - Product Development: Requirements Phase	Process	n.a.	Marketing / Product Management	Modular System Scope	Planning Modular System			
6	Primary case B	Without modularisation process/method, engineers might not consider important module drivers, etc. as they are not used to this thinking from the past => Method considering this helps here	PLC - Product Development: Concept Phase	Process	System/Product Architecting	R & D	Modular System Scope	Planning Modular System			

Id	Disguised Source	Support	Product Design / Life cycle Phase	Aspect Classifi-cation	Sorting Classifica-tion	Company Function	Scope Classifica-tion	Modular System De-sign / Life cycle Phase	Issue	Impor-tant Factor	Contra-dictory to
7	Primary case B	Process with step that considers module drivers, PLC issues & that parts are VARIANT, DEVELOPMENT, BASE or OPTION	PLC - Product Development: Concept Phase	Process	Sys-tem/Product Archi-tecting	Diverse	Modular System Scope	Planning Modular System			
8	Primary case B	Module specification with CRs, PPs, Strategic Drivers, Sketches, impacted modules in case of change all other data for module	PLC - Product Development: Concept Phase	Process	n.a.	R & D	Modular System Scope	n.a.			
9	Primary case B	Module variants are built based on the goal values of the product properties . Ideally, each goal value of a product property gets its module variant => configuration matrix	PLC - Product Development: Concept Phase	Process	n.a.	R & D	Modular System Scope	n.a.			
10	Primary case B	Apply black magic for module grouping, i.e. experience in market development, modularisation and technology	PLC - Product Development: Concept Phase	Process	Sys-tem/Product Archi-tecting	R & D	Modular System Scope	Planning Modular System			
11	Primary case B	Size of modules has to be determined with experience.	PLC - Product Development: Concept Phase	Process	Sys-tem/Product Archi-tecting	R & D	Modular System Scope	Planning Modular System			
12	Secondary case	Leave those parts of the product with unstable or unknown customer requirements outside the modular system	PLC - Product Development: Concept Phase	Process	Sys-tem/Product Archi-tecting	R & D	Modular System Scope	Planning Modular System			
13	Secondary case	Involve all company areas: a) show them impacts each time a new MV is introduced, b) show them how they can best benefit from a modular system	PLC - Product Development: Concept Phase	Introduc-tion	Change Manage-ment	Diverse	n.a.	n.a.			
14	Secondary case	Prioritisation of targets (e.g. purchasing price, dynamic appearance, weight, quality,…) in order to resolve issues between module responsible and project responsibles	PLC - Product Development: Concept Phase	Process	Sys-tem/Product Archi-tecting	Diverse	Modular System Scope	Planning Modular System			

Id	Disguised Source	Support	Product Design / Life cycle Phase	Aspect Classification	Sorting Classification	Company Function	Scope Classification	Modular System Design / Life cycle Phase	Issue	Important Factor	Contradictory to
15	Secondary case	De-couple different technologies as they have different life cycles (SW, electronics, mechanics)	PLC - Product Development: Concept Phase	Process	System/Product Architecting	Diverse	Modular System Scope	Planning Modular System			
16	Secondary case	Discussion and evaluation of product architecture much more valuable than tools, methods, etc. a) algorithm => suggestion for modules => discussion/evaluation => modules b) matrix => discussion/evaluation => modules c) visualisation => discussion/evaluation => modules	PLC - Product Development: Concept Phase	Process	System/Product Architecting	R & D	Modular System Scope	Planning Modular System			
17	Primary case A	Group parts into modules through: variance vs. Standard, possibility of change, etc.	PLC - Product Development: Concept Phase	Process	System/Product Architecting	R & D	n.a.	n.a.			
18	Primary case A	When will I make global platform, when local platform, when variant?? => make decision	PLC - Product Development: Concept Phase	Process	System/Product Architecting	R & D	n.a.	n.a.			
19	Primary case C	Justified (with data), triangulated, traceable and documented decisions that we carry throughout the modularisation process => to avoid discussions; constantly monitor this data	PLC - Product Development: Concept Phase	Process	n.a.	Diverse	Modular System Scope	Diverse			
20	Primary case C	Break down target prices into different modules, over importance/requirements	PLC - Product Development: Concept Phase	Process	Financial	Marketing / Product Management	Modular System Scope	Planning Modular System			
21	Primary case C	Cluster very critical and varying success factors into modules with defined interfaces	PLC - Product Development: Concept Phase	Process	System/Product Architecting	Diverse	Modular System Scope	Planning Modular System			

Id	Disguised Source	Support	Product Design / Life cycle Phase	Aspect Classification	Sorting Classification	Company Function	Scope Classification	Modular System Design / Life cycle Phase	Issue	Important Factor	Contradictory to
22	Primary case C	Modules have to be created/suggested from experienced design engineers instead of with method or with consultants!	PLC - Product Development: Concept Phase	Process	System/Product Architecting	R & D	Modular System Scope	Planning Modular System			
23	Primary case C	For modularisation, certainty about technology is necessary => although this contradicts to solution-neutrality, already start from the beginning to think technically, to research, to simulate, to make tests, to develop	PLC - Product Development: Concept Phase	Process	System/Product Architecting	R & D	n.a.	n.a.			
24	Primary case C	Apply modularisation method (architecting part of it) as check rather than as tool to set it up	PLC - Product Development: Concept Phase	Process	n.a.	Diverse	Modular System Scope	Planning Modular System	80	78	
25	Primary case C	Onion peel model with criteria for each layer!	PLC - Product Development: Concept Phase	Process	n.a.	R & D	Modular System Scope	Diverse			
26	Primary case A	Establish same interfaces for the same function (e.g. generate heat, transfer energy)	PLC - Product Development: Concept Phase	Process	System/Product Architecting	R & D	n.a.	n.a.			
27	Primary case A	Negotiations to reduce variety of module variants what is technical reasonable from platform point of view and what is reasonable from market point of view. Be careful, not all reasonable decisions lead to successful products.	PLC - Product Development: Concept Phase	Process	n.a.	Diverse	Modular System Scope	Planning Modular System			
28	Secondary case	Making a module out of an item, makes it sure that the item will also be reused in other projects as long as the interface is described and prescribed for other projects Therefore, all possible applications have to be considered where the module has to be applied (though, difficult from engineering and PRM point of view)	PLC - Product Development: Concept Phase	Process	n.a.	Diverse	Modular System Scope	Planning Modular System			

Id	Disguised Source	Support	Product Design / Life cycle Phase	Aspect Classifi-cation	Sorting Classifica-tion	Company Function	Scope Classifica-tion	Modular System De-sign / Life cycle Phase	Issue	Impor-tant Factor	Contra-dictory to
29	Secondary case	Approval Management - Architecture design - independency of modules, de-coupled interfaces, func-tional independency (Rückwirkungsfreiheit) - separation into safety-relevant and less safety-relevant modules => cluster into modules according to safety-criticality. => only approval of deltas for new version (see research notes on 21.04.2015)	PLC - Product Development: Concept Phase	Process	Sys-tem/Product Archi-tecting	R & D	n.a.	n.a.			
30	Secondary case	In most cases, choosing the right product architecture alternative requires to get agreement by management, customers and other stakeholders.	PLC - Product Development: Concept Phase	Process	Sys-tem/Product Archi-tecting	R & D	n.a.	n.a.			
31	Secondary case	Create different architecture alternatives => then rate them according to the diverse requirements that you consider as important to be fulfilled by the architecture	PLC - Product Development: Concept Phase	Process	Sys-tem/Product Archi-tecting	R & D	n.a.	n.a.			
32	Secondary case	Each variant to be generated has to be justified with a validated Return on Investment, etc.	PLC - Product Development: Concept Phase	Process	n.a.	Marketing / Product Management	Modular System Scope	Planning Modular System			
33	Secondary case	Architecting: -> Choosing between different alternatives based on differ-ent criteria (e.g. effectiveness, performance, cost, schedule) - static analyses of architectures - dynamic analyses of architectures - experiments - trials	PLC - Product Development: Concept Phase	Process	Sys-tem/Product Archi-tecting	R & D	Modular System Scope	Planning Modular System			
34	Secondary case	Proper architecture planning, storing and making product architecture information available across projects	PLC - Product Development: Design Phase	IT	n.a.	Diverse	Modular System Scope	Diverse			

Id	Disguised Source	Support	Product Design / Life cycle Phase	Aspect Classifi-cation	Sorting Classifica-tion	Company Function	Scope Classifica-tion	Modular System De-sign / Life cycle Phase	Issue	Impor-tant Factor	Contra-dictory to
35	Secondary case	Transparency: Installing identification mechanisms to identify already existing modules	PLC - Product Development: Design Phase	IT	n.a.	R & D	Modular System Scope	Developing Modular System			
36	Primary case B	Interface specification with a… b) …c)	PLC - Product Development: Design Phase	Process	n.a.	R & D	Modular System Scope	n.a.			
37	Secondary case	Define and display unchangeable platform dimensions	PLC - Product Development: Design Phase	IT	Sys-tem/Product Archi-tecting	R & D	Modular System Scope	Developing Modular System			
38	Secondary case	Define PDM with modularisation drawings as leading system, modularisation item masters in ERP, but different manufacturing BOMs in ERP	PLC - Product Development: Design Phase	IT	n.a.	R & D	Modular System Scope	Developing Modular System			
39	Primary case A	During development process audits, during quality gates and milestone reviews, do not ask about the product, ask about the platform and modules	PLC - Product Development: Design Phase	Evalua-tion	n.a.	R & D	Modular System Scope	Developing Modular System			
40	Primary case C	Think more in terms of BOMs (i.e. DEV, MFG, Service) than in terms of abstract methodologies (CAD, PDM, ERP, IM)	PLC - Product Development: Design Phase	IT	Sys-tem/Product Archi-tecting	R & D	Modular System Scope	Developing Modular System	134	56	
41	Primary case C	Single modules have to be clearly managed by an expert with enough capacity to manage a module (up to namings, classification, KPIs, IT-Integration, LOV, maintaining trans-fer documents, morphological box, assigning always same parts) and knowledge to still oversee all interfaces to the whole product (At primary case C we made the mistake to assign all mod-ules to one engineer, see automotive industry where all teams work on modules and dedicated interfaces only)	PLC - Product Development: Design Phase	Organisa-tion	n.a.	R & D	Modular System Scope	Diverse			

Id	Disguised Source	Support	Product Design / Life cycle Phase	Aspect Classifi-cation	Sorting Classifica-tion	Company Function	Scope Classifica-tion	Modular System De-sign / Life cycle Phase	Issue	Impor-tant Factor	Contra-dictory to
42	Secondary case	From systems engineering, internal and external interfaces (up to drawings, etc) are treated equally to components. They first have to be detailed before they are given to internal or external designer => well knowing that this is only valid for single system development.	PLC - Product Development: Design Phase	Process	n.a.	R & D	n.a.	n.a.			
43	Secondary case	Giving "meat" to different concepts: - break even point for single product development versus full configuration on the other side of the scale - different product reuse categories (see secondary case Y) or different LEVELS OF MODULARITY	PLC - Product Development: Design Phase	Diverse	n.a.	R & D	n.a.	n.a.			
44	Secondary case	Dedicated process to find out whether newly created modu-lar solution can fulfil customer needs or not	PLC - Product Development: Testing	Process	n.a.	Marketing / Product Management	Modular System Scope	Diverse	4	5	
45	Secondary case	Dedicated process to find out whether technical concept of new modules is feasible or not	PLC - Product Development: Testing	Process	n.a.	R & D	Modular System Scope	Diverse			
46	Secondary case	Dedicated process to find out whether modular system can be realised or not (e.g. through own process or to gradual process implemented in development projects)	PLC - Product Development: Testing	Process	n.a.	R & D	Modular System Scope	Diverse	6	9	
47	Primary case A	Product Architecture Process (PAP) has to at least end with feasibility study of modular system => more than A-Sample	PLC - Product Development: Testing	Process	n.a.	R & D	Modular System Scope	Planning Modular System			
48	Primary case C	We blocked a lot of time with no other work only for testing	PLC - Product Development: Testing	Process	n.a.	R & D	Modular System Scope	n.a.			
49	Primary case C	In order to convince engineers, it is important to make a dry run or a conceptual study before one fully starts with modularisation.	PLC - Product Development: Testing	Process	n.a.	R & D	Modular System Scope	Diverse			

Id	Disguised Source	Support	Product Design / Life cycle Phase	Aspect Classification	Sorting Classification	Company Function	Scope Classification	Modular System Design / Life cycle Phase	Issue	Important Factor	Contradictory to
50	Primary case C	Before one details the modular system and before freezing these details (in NPD), it is important to come up with A-/ and B-Samples	PLC - Product Development: Testing	Process	n.a.	R & D	Modular System Scope	Developing Modular System	91	50	
51	Primary case A	Benchmark partner V: shift from product plants to module plants that deliver different final assembly plants for products	PLC - Product Development: Production Ramp-up	Organisation	n.a.	Production	Modular System Scope	Diverse			
52	Secondary case	Engineering change process that controls the creation of new module variants according or in alignment with plans	Evolution of Products, Reuse, Design Modifications, Engineering Change	Process	n.a.	R & D	Modular System Scope	Maintenance of Modular System			
53	Secondary case	Measurement and reporting whether the modular system is developed according to the plans	Evolution of Products, Reuse, Design Modifications, Engineering Change	Evaluation	n.a.	R & D	Modular System Scope	Diverse			
54	Secondary case	Single projects have to have benefit in sticking to modular system (e.g. through fulfilling important metrics that in turn have an effect)	Evolution of Products, Reuse, Design Modifications, Engineering Change	Evaluation	n.a.	R & D	Modular System Scope	Developing Modular System			
55	Secondary case	Description of phase of evolvement of modular system must be available.	Evolution of Products, Reuse, Design Modifications, Engineering Change	Process	n.a.	Diverse	Modular System Scope	Diverse			

Id	Disguised Source	Support	Product Design / Life cycle Phase	Aspect Classification	Sorting Classification	Company Function	Scope Classification	Modular System Design / Life cycle Phase	Issue	Important Factor	Contradictory to
56	Secondary case	Dedicated change process to change modules, parts within modules or interfaces	Evolution of Products, Reuse, Design Modifications, Engineering Change	Process	n.a.	R & D	Modular System Scope	Maintenance of Modular System			
57	Secondary case	Modules can be kept stable if changes to them always go over module responsible	Evolution of Products, Reuse, Design Modifications, Engineering Change	Process	n.a.	R & D	Modular System Scope	Maintenance of Modular System			
58	Secondary case	If requests from marketing/sales cannot be fulfilled with existing modules (over product configuration), sales has to show financial benefits through introduction (e.g. volume > 300 pcs.) and modular system responsible has to outweigh internal costs for it.	Evolution of Products, Reuse, Design Modifications, Engineering Change	Process	n.a.	Marketing / Product Management	Modular System Scope	Maintenance of Modular System			
59	Secondary case	Starting with a few fixed modules and then modularising the product portfolio gradually	Evolution of Products, Reuse, Design Modifications, Engineering Change	Process	n.a.	R & D	Modular System Scope	Maintenance of Modular System			
60	Secondary case	Permanently measure complexity in projects.	Evolution of Products, Reuse, Design Modifications, Engineering Change	Evaluation	n.a.	R & D	n.a.	n.a.			
61	Secondary case	More important than method is coordination of architecture between different projects	Evolution of Products, Reuse, Design Modifications, Engineering Change	Process	n.a.	Diverse	Modular System Scope	Maintenance of Modular System			

Id	Disguised Source	Support	Product Design / Life cycle Phase	Aspect Classification	Sorting Classification	Company Function	Scope Classification	Modular System Design / Life cycle Phase	Issue	Important Factor	Contradictory to
62	Primary case C	Establish process for changes on architecture (e.g. interfaces, module envelopes, variance goals)	Evolution of Products, Reuse, Design Modifications, Engineering Change	Process	n.a.	R & D	Modular System Scope	Maintenance of Modular System			
63	Primary case C	As we cannot detail the modules from the very beginning, it is important to constantly maintain architecture documents with overall information that "glues" the fragile modular system together.	Evolution of Products, Reuse, Design Modifications, Engineering Change	Process	n.a.	Diverse	Modular System Scope	Diverse			
64	Primary case A	Integrate modular system development into quality gates of product development process!	Evolution of Products, Reuse, Design Modifications, Engineering Change	Process	n.a.	R & D	Modular System Scope	Diverse			
65	Primary case C	Comparison to machinery industry: central module development with scope on different applications, for new modules a request to central department had to be written and this was developed or rejected	Evolution of Products, Reuse, Design Modifications, Engineering Change	Organisation	n.a.	R & D	Modular System Scope	Maintenance of Modular System			
66	Primary case A	Introduce release engineering as interface between product structuring and PLC	Evolution of Products, Reuse, Design Modifications, Engineering Change	Process	n.a.	R & D	Modular System Scope	Diverse			
67	Secondary case	For changes on modules: - change boards/change meetings - dedicated senior architecture roles > project roles	Evolution of Products, Reuse, Design Modifications, Engineering Change	Process	n.a.	R & D	Modular System Scope	Maintenance of Modular System	145	n.a.	

Id	Disguised Source	Support	Product Design / Life cycle Phase	Aspect Classification	Sorting Classification	Company Function	Scope Classification	Modular System Design / Life cycle Phase	Issue	Important Factor	Contradictory to
68	Secondary case	Separate Interfaces versus Interfaces within modules => Make decision how to handle fixation of interfaces - What changes more frequently: Module or interface - What is intended to be kept stable? - What is intended to be interchanged? - What is intended to be reused across product generations? Pro separation of interfaces: - stability of interface < stability of module - reuse of interface, interchanging interface - potential of interface commonality - effort to change interface > effort to separately handle interface - if I change the interface, I don't have to change the modules Pro interfaces inside modules - interface less an end in itself - focus on reusing modules instead of interfaces - stability of interface - interface vital part of interchange ability of modules - high effort to handle interface separately (see research comment from 30.04.2015 -> Example that the same interface is used to connect different modules. If the interface changes, all modules have to be adopted for interface inside modules. If the interface changes for separate interfaces, the interface only has to be handled once while no module has to be touched.	Evolution of Products, Reuse, Design Modifications, Engineering Change	Diverse	n.a.	R & D	Modular System Scope	Diverse			

Id	Disguised Source	Support	Product Design / Life cycle Phase	Aspect Classifi-cation	Sorting Classifica-tion	Company Function	Scope Classifica-tion	Modular System De-sign / Life cycle Phase	Issue	Impor-tant Factor	Contra-dictory to
69	Secondary case	Regular or need-based product line reviews to decide whether to make amendments to the modular system or whether to develop a new modular system and to assess performance against the plan (reduced time to market, reduced cost, cost saving and quality improvement through common tests) => product line evolutions and updating the plan.	Evolution of Products, Reuse, Design Modifica-tions, Engineer-ing Change	Process	n.a.	Diverse	Modular System Scope	Maintenance of Modular System			
70	Secondary case	Regular review of product line strategy: based on customer and competitiveness expectations	Evolution of Products, Reuse, Design Modifica-tions, Engineer-ing Change	Process	n.a.	Marketing / Product Management	Modular System Scope	Maintenance of Modular System			
71	Secondary case	Architecture change management: Monitor technology changes, monitor business changes, Arrange meetings of Architecture Board	Evolution of Products, Reuse, Design Modifica-tions, Engineer-ing Change	n.a.	n.a.	R & D	Modular System Scope	Maintenance of Modular System			
72	Secondary case	Finding tested and usable building-blocks => - reference in databases/catalogues for non-authors/non-direct-developers - documentation of building blocks is written also for non-authors/non-direct-developers	Evolution of Products, Reuse, Design Modifica-tions, Engineer-ing Change	IT	n.a.	R & D	Modular System Scope	Diverse			
73	Secondary case	Feature models have to be reviewed by stakeholders regu-larly in order to remove obsolete feature, to add new one and to update configurations with new bids; that allows updating product line road maps. Building feature models are iterative processes, the product line design has to remain stable whole protecting and sufficient adaptability in order to answer to evolution of market needs.	Evolution of Products, Reuse, Design Modifica-tions, Engineer-ing Change	Process	n.a.	Diverse	Modular System Scope	Maintenance of Modular System			
74	Secondary case	In production there has also to be a series responsible compared to a project responsible	Value Stream - Production	Process	n.a.	Production	n.a.	n.a.			

Id	Disguised Source	Support	Product Design / Life cycle Phase	Aspect Classification	Sorting Classification	Company Function	Scope Classification	Modular System Design / Life cycle Phase	Issue	Important Factor	Contradictory to
75	Secondary case	Organisational entity that controls the creation of new module variants according to original plans	Diverse	Organisation	n.a.	R & D	Modular System Scope	Diverse			
76	Secondary case	Install new modular system development process "above" product development project with dedicated budget and organisation or give extra budget for product development projects for modularisation.	Diverse	Process	n.a.	Diverse	Modular System Scope	Diverse			
77	Primary case B	Process integration of modularisation relevant activities into company standard processes	Diverse	Process	n.a.	Diverse	Modular System Scope	Diverse			
78	Primary case B	Separate module development process (e.g. in parallel to product development)	Diverse	Process	n.a.	Diverse	Modular System Scope	Diverse			
79	Primary case B	Get the understanding of an informed and committed management.	Diverse	Introduction	Change Management	Top Management	Modular System Scope	Diverse			
80	Primary case B	Central modular system development organisation equipped with enough resources.	Diverse	Organisation	n.a.	Diverse	Modular System Scope	Diverse			
81	Primary case B	For gradual development: separation in more or less to be fixed module variants, modules and interfaces which can be fixed at different stages	Diverse	Process	n.a.	R & D	Modular System Scope	Developing Modular System	23	23	
82	Primary case B	Separate module and interface management process	Diverse	Process	n.a.	R & D	Modular System Scope	Diverse	24	24	
83	Secondary case	Higher hierarchy levels have to make stringent decisions throughout the process from introduction onwards	Diverse	Process	n.a.	Top Management	n.a.	n.a.			
84	Secondary case	Introduction of central and strong module responsibles (with knowledge about commonalities) with power over local and project responsibles. Process must go over central responsible.	Diverse	Organisation	n.a.	Diverse	n.a.	n.a.	29	29	
85	Secondary case	Technical modular system responsible who protects modules but also interfaces on platform drawings.	Diverse	Organisation	n.a.	R & D	n.a.	n.a.			

Id	Disguised Source	Support	Product Design / Life cycle Phase	Aspect Classification	Sorting Classification	Company Function	Scope Classification	Modular System Design / Life cycle Phase	Issue	Important Factor	Contradictory to
86	Secondary case	Module organisation with: Development Tasks List, Module Roadmap, Module Meetings, Module Review, Module Day, Documentation	Diverse	Organisation	n.a.	R & D	n.a.	n.a.			
87	Secondary case	PRM and engineers have to be trained and guided by their managers	Diverse	Introduction	n.a.	Diverse	n.a.	n.a.			
88	Secondary case	Make a shift from single product and project centred approaches (modules, evaluations, responsibilities, costing) toward module-centred approaches	Diverse	Diverse	n.a.	Diverse	Modular System Scope	Diverse	34	34	
89	Secondary case	Add overhead cost for complexity to product cost calculations.	Diverse	Evaluation	Financial	Diverse	Modular System Scope	Diverse			
90	Secondary case	Constant measurement of complexity and modularisation to pull the programme.	Diverse	Evaluation	n.a.	Diverse	Modular System Scope	Diverse			
91	Secondary case	Modularisation over highest hierarchy and on the agenda on the company's strategy	Diverse	Introduction	n.a.	Diverse	n.a.	n.a.			
92	Secondary case	Constant monitoring of development of market demand and target costs => functional-technical dependency gets relevant here.	Diverse	Evaluation	Financial	Marketing / Product Management	n.a.	n.a.			
93	Primary case A	Vital to get agreed concept (modular system and processes) and cost reduction for the concept at a quite early state	Diverse	Introduction	n.a.	Diverse	n.a.	n.a.			
94	Primary case A	Major support: financial facts about optimum variety, etc. AND time, motivation, information	Diverse	Diverse	Change Management	Diverse	n.a.	n.a.			
95	Primary case A	Metrics to make success and benefit of modularisation transparent => start with goal values for metrics from the very beginning.	Diverse	Evaluation	n.a.	Diverse	Modular System Scope	Diverse			
96	Primary case A	Establish either financial evaluation OR pull of management for modularisation	Diverse	Evaluation	n.a.	Diverse	Modular System Scope	Diverse	59	38	

Id	Disguised Source	Support	Product Design / Life cycle Phase	Aspect Classification	Sorting Classification	Company Function	Scope Classification	Modular System Design / Life cycle Phase	Issue	Important Factor	Contradictory to
97	Secondary case	Different levels of architectural strategy with different requirements, support, pros and cons => decide before project	Diverse	Introduction	n.a.	Diverse	n.a.	n.a.	61	40	
98	Primary case A	Primary case A, Secondary case Y: Product Line Review per PB - where are we, where do we want to go to, what are our figures (sales per feature, comparison to plan, target cost, see secondary case Y Excel-Files)	Diverse	Evaluation	n.a.	Diverse	Modular System Scope	Planning Modular System			
99	Primary case A	Tailoring of modularisation process: for new product development, for redesign of existing products	Diverse	Process	n.a.	Diverse	n.a.	n.a.			
100	Primary case A	First we have to know the overall picture before we can proceed with such a huge process	Diverse	Introduction	Change Management	Diverse	n.a.	n.a.	63	41	
101	Primary case A	Modularisation should not (only) be integrated into the process by adding new QG Questions and columns in templates => overall initiative	Diverse	Introduction	Change Management	Diverse	n.a.	n.a.	63	41	
102	Primary case C	If the risk is too high, if we don't know enough and if we don't have the time, an offensive decision against modularisation has to be made and marketed	Diverse	Introduction	n.a.	Top Management	n.a.	n.a.			
103	Primary case C	For modularisation under technological constraints and time pressure it is better to focus on a few core modules one is confident with	Diverse	Process	n.a.	Diverse	Modular System Scope	Diverse	71	42	
104	Primary case C	For support, try to make simple and lean things instead of always adding something on top (even though it might be necessary to shift workload with modularisation)	Diverse	Diverse	n.a.	Diverse	n.a.	n.a.			
105	Primary case C	Process integration and scaling modularisation based on: a) level that shall be achieved, b) knowledge about market, c) knowledge about product, d) available time and resources, e) existing modules, architecture, commonalities, f) platform risk / fluctuation	Diverse	Introduction	n.a.	Diverse	Modular System Scope	n.a.			

Id	Disguised Source	Support	Product Design / Life cycle Phase	Aspect Classification	Sorting Classification	Company Function	Scope Classification	Modular System Design / Life cycle Phase	Issue	Important Factor	Contradictory to
106	Primary case C	Roadmapping: products, modules, etc. and constant monitoring of underlying data very important	Diverse	Process	n.a.	Diverse	Modular System Scope	n.a.			
107	Primary case C	Let final decisions agree by top management!	Diverse	Diverse	n.a.	Top Management	n.a.	n.a.			
108	Primary case C	Accompany modularisation (workshops) with technical workshops to clarify the detailed solution and with testing	Diverse	Process	n.a.	R & D	Modular System Scope	n.a.			
109	Primary case C	In order to keep the overall overview about the whole modular system, a module-module variant matrix and a module - product family matrix with roadmaps proved to be helpful	Diverse	Process	n.a.	Diverse	Modular System Scope	Diverse			
110	Primary case C	Modular systems definitely work in different industries (e.g. automotive industry, home appliances, machine industry), why? => a) modular systems serve all customer demands, b) no development outside modular system	Diverse	Diverse	n.a.	Diverse	Modular System Scope	Diverse			
111	Primary case C	Top-down support to focus on an overall platform has to be available while considering all consequences of such a strategy => that has to be constantly measured	Diverse	Evaluation	n.a.	Top Management	Modular System Scope	Diverse			
112	Primary case C	Layer model: development process triggered by change of a module dependent on position on layer model: a) Inner: Full NPD Project b) Middle: Modification Project c) Outer: Engineering change process without product development project (quickly), see quick fix	Diverse	Process	n.a.	R & D	Modular System Scope	Diverse			
113	Primary case C	Feature tree good tool to visualise overall variance of modular system => we don't have another tool for that so far.	Diverse	Process	n.a.	Diverse	Modular System Scope	Diverse			
114	Primary case C	KPIs with purpose to measure stability of modular system, how is the modular system developing, how successful are we?	Diverse	Evaluation	n.a.	Diverse	Modular System Scope	Diverse			

Id	Disguised Source	Support	Product Design / Life cycle Phase	Aspect Classifi-cation	Sorting Classifica-tion	Company Function	Scope Classifica-tion	Modular System De-sign / Life cycle Phase	Issue	Impor-tant Factor	Contra-dictory to
115	Primary case C	Lessons Learned from Study XY: Think modules properly to the end, roadmapping for platform and modules, Interdisci-plinary (production, SCM, controlling), evolution of modular system and variant projects in parallel, definition of respon-sibilities	Diverse	Diverse	n.a.	Diverse	Modular System Scope	Diverse			
116	Primary case A	Remove Plant/Project anarchy	Diverse	Organisa-tion	n.a.	Diverse	Modular System Scope	Diverse			
117	Primary case A	Ratio-Projects / Benchmark projects on modules instead of on single products	Diverse	Diverse	n.a.	R & D	n.a.	n.a.			
118	Primary case A	In order to think in modules, secondary case X removed the term "platform"	Diverse	Introduc-tion	n.a.	Diverse	n.a.	n.a.			
119	Primary case A	Modularisation needs high management commitment, changed mindset and new approaches throughout the whole organisation	Diverse	Introduc-tion	Change Manage-ment	Diverse	n.a.	n.a.			
120	Primary case A	Develop modules in module centres instead of in product development projects (or only very short product develop-ment projects) Benchmark partner V structured products into independent modules and develops modules relatively independently. However, through that they can be combined and used for all brands. =>> For that, all appliances first need the same product structure with the same interfaces!! Therefore, the architecture has to be quite mature	Diverse	Organisa-tion	n.a.	R & D	Modular System Scope	Diverse			

Id	Disguised Source	Support	Product Design / Life cycle Phase	Aspect Classification	Sorting Classification	Company Function	Scope Classification	Modular System Design / Life cycle Phase	Issue	Important Factor	Contradictory to
121	Secondary case	Benchmark partner B: Platform has to be developed until B-Sample (quality gate 0 - quality gate 2), (3-5 years DEV of platform, platform life cycle 8 - 10 years), afterwards maintenance of platform (exchange with Mr. XY from UZ) => customer project (h) << platform project, deviations from platform are not known, some special customer wishes have to be paid for themselves => designated processes - platform covers as much as possible => very expensive - platform checklist with different areas - outside platform decision by management - for customer projects only delta FMEAs => savings - no dedicated methodology for platform design, discussions and review	Diverse	Process	n.a.	R & D	Modular System Scope	Diverse			
122	Primary case A	Management has to understand that they have to invest more money at the beginning => projects have to be organised differently => modular system development project as pre-development, benefit does not come before variant projects	Diverse	Process	n.a.	Diverse	Modular System Scope	Planning Modular System			
123	Primary case A	Two different possibilities to transition toward modular system development: - Gradually like Primary case A, con: for a very long time you cannot see anything -> gradually increase number of products that are covered by modular system, very bumpy - Once, centrally like WZ: con: a lot of effort without benefit for long time	Diverse	Introduction	n.a.	Diverse	Modular System Scope	Diverse			
124	Primary case A	Transfer documents: - Full architecture requirement engineering-Template - Feature Tree - Module-Product-Matrix - Module-Variant-Matrix - Interface-Matrix - Variant Tree	Diverse	Process	n.a.	Diverse	Modular System Scope	Planning Modular System			

Id	Disguised Source	Support	Product Design / Life cycle Phase	Aspect Classification	Sorting Classification	Company Function	Scope Classification	Modular System Design / Life cycle Phase	Issue	Important Factor	Contradictory to
125	Primary case A	Introduce modularisation over complexity cost approach 2000 / 3000 / 6000 versus 2000 / 5000 / 30000	Diverse	Evaluation	Financial	Diverse	Modular System Scope	Diverse			
126	Primary case A	Modularisation as innovation and feasibility project for variant projects => however, it is questionable whether this is the holy grail or not	Diverse	Process	n.a.	R & D	Modular System Scope	Diverse			
127	Primary case A	Establish rules for BOMs: BOM rules or BOM guideline	Diverse	IT	n.a.	R & D	Modular System Scope	Developing Modular System			
128	Primary case C	Don't establish modular system if there is no commitment for frontloading.	Diverse	Introduction	n.a.	R & D	Modular System Scope	Developing Modular System			
129	Secondary case	Modular systems are pure trust in the concept by management. Otherwise it does not work (see secondary case X).	Diverse	Introduction	n.a.	R & D	Modular System Scope	Diverse			
130	Primary case B	No need to follow method, it is necessary to define what steps have to be done with which purpose and to control that	Diverse	Process	System/Product Architecting	R & D	Modular System Scope	Diverse			
131	Primary case A	Measure with KPIs so that no one can claim what is good and what is bad modularisation => quantifiable	Diverse	Evaluation	n.a.	R & D	Modular System Scope	Diverse			
132	Primary case A	Establish measurement at the point where complexity is created => there, complexity has to hurt	Diverse	Evaluation	n.a.	Diverse	Modular System Scope	Diverse			
133	Primary case A	The critical point is that management sets complexity on the broader scope as stringent target!	Diverse	Introduction	n.a.	Diverse	Modular System Scope	Diverse			

Id	Disguised Source	Support	Product Design / Life cycle Phase	Aspect Classifi-cation	Sorting Classifica-tion	Company Function	Scope Classifica-tion	Modular System De-sign / Life cycle Phase	Issue	Impor-tant Factor	Contra-dictory to
134	Primary case A	Different strategies how to transition toward modular system: - modular system by products - modular system by functions - hybrid strategy (further options: see research notes 12/06/2012)	Diverse	Process	n.a.	R & D	Modular System Scope	Diverse			
135	Primary case A	Acceptance on modular system depends on degree to what extend the modular system is already filled!	Diverse	Introduc-tion	Change Manage-ment	R & D	Modular System Scope	Diverse			
136	Primary case A	It is vital to show that this sort of modularisation is some-thing NEW: Show the difference of what we intent now to what we have done in the past: - modular products vs. modular system - fixed and broad requirements - scope - interfaces - commonality planning - stability of commonality - cross-brand, cross-site, cross-XYZ - ...	Diverse	Introduc-tion	Change Manage-ment	Diverse	Modular System Scope	Diverse			
137	Primary case A	Share of product architecture information between different derivative teams: - documentation - PLM - regular meetings - workflow - heavy-weight platform manager	Diverse	IT	n.a.	Diverse	Modular System Scope	Diverse			
138	Primary case A	Modularisation can be pulled on engineer level if they are triggered by their managers: e.g. through audit => if they get asked: "What are you doing to keep your modular system sustainable?" => They will come and ask what they can do!	Diverse	Evalua-tion	n.a.	Diverse	Modular System Scope	Diverse			

Id	Disguised Source	Support	Product Design / Life cycle Phase	Aspect Classification	Sorting Classification	Company Function	Scope Classification	Modular System Design / Life cycle Phase	Issue	Important Factor	Contradictory to
139	Secondary case	Central configuration department at secondary case Z: They realised that de-central does not work for central product architectures. That is why they established a central coordination office that harmonizes between market and manufacturing and that sets rules for the platform	Diverse	Organisation	n.a.	R & D	Modular System Scope	Diverse			
140	Primary case C	Modularisation can be pulled (change in mindset??) by coupling modularisation targets with bottom-up and top-down target agreement of employees	Diverse	Evaluation	n.a.	R & D	Modular System Scope	Diverse			
141	Secondary case	We need organisational entity that balances out the interests between PRM and ENG, or between commonality and variety.	Diverse	Organisation	n.a.	Diverse	Modular System Scope	Diverse			
142	Secondary case	Architecting should start on high-level and be broken down until design engineers can start to design their components based on specifications and interface descriptions => lower levels have to stick to higher level design & architecture rules. Design on lower level should fit into the design of higher levels, in case changes on higher level architecture is necessary, there should be a change process under consideration of all impacts	Diverse	Process	n.a.	R & D	n.a.	n.a.			
143	Secondary case	- Pre-thinking of modular system/ generic architecture: rules, roadmaps & plans - prescribed product structure - central, neutral elements which will be linked - planned reuse of elements - measurement / architecture or commonality targets	Diverse	IT	n.a.	R & D	n.a.	n.a.			
144	Secondary case	Each time different naming, no classification, you don't find anything => normative framework in process and IT is a great chance to get transparency about those issues, though, high amounts of resources have to be invested	Diverse	Introduction	n.a.	Diverse	n.a.	n.a.			

Id	Disguised Source	Support	Product Design / Life cycle Phase	Aspect Classification	Sorting Classification	Company Function	Scope Classification	Modular System Design / Life cycle Phase	Issue	Important Factor	Contradictory to
145	Secondary case	Define the architectural design iteratively on different levels: - needs representation - System interactions and functions: - Logical architecture: a) create high level logical architecture, b) perform the internal functional analysis, c) develop the logical architecture, d) allocate requirements - Physical architecture that formalises how the technical solution performs the required operations within the deployed system a) identify alternative physical solutions, b) select solution from different alternatives, c) justify the technical solution - System Breakdown that formalises the acquisition of its components a) finalise the system design, b) consolidate the configuration items requirements, c) control the design process d) control the quality of the system design and perform the gate review (from systems engineering)	Diverse	Process	System/Product Architecting	Diverse	Product / Project Scope	n.a.			
146	Secondary case	The organisational concept of a market-PLM > product manager versus a design authority > architecture responsible > product developer	Diverse	Organisation	n.a.	Diverse	Modular System Scope	Diverse			
147	Secondary case	Separation of generic module development and development of products. However, products have to stick to generic modules later on.	Diverse	Organisation	n.a.	R & D	Modular System Scope	Diverse			

Id	Disguised Source	Support	Product Design / Life cycle Phase	Aspect Classification	Sorting Classification	Company Function	Scope Classification	Modular System Design / Life cycle Phase	Issue	Important Factor	Contradictory to
148	Secondary case	Design constraints as design requirements for respective products/systems: e.g. - Architecture specification requirements - Make / Buy / Reuse strategy results - Reuse opportunities Use of COTS, COSS (Open Source Software), freeware Product line components Reusable software components from the company's library functional models - architecture specification => link these requirements to product (integration) tests or the like	Diverse	Process	n.a.	R & D	Modular System Scope	Diverse			
149	Secondary case	Like in the PLE approach, distinguish between modular system development (domain engineering, coming from product policy, market needs) and development for projects (application engineering, coming from "manage bid", customer needs)	Diverse	Process	n.a.	Diverse	Modular System Scope	Diverse			
150	Secondary case	Establish following points for PLE / preconditions: - standardised market for development of generic products - PRM has to know that the products will stay inside variability limits and that going outside these limits is impossible and fraught with costs and delays - separation of modular system development and project development in order to avoid project-driven specifities. Modular system owners must have a weight larger than project heads. - funding must be ensured to sustain the intended implementation of the product policy with building blocks	Diverse	Process	n.a.	Diverse	Modular System Scope	Diverse			

Id	Disguised Source	Support	Product Design / Life cycle Phase	Aspect Classification	Sorting Classification	Company Function	Scope Classification	Modular System Design / Life cycle Phase	Issue	Important Factor	Contradictory to
151	Primary case A	Apart from a good modularisation method, following is needed: - Modularisation approach integrated into process landscape to reduce internal effort and to avoid double work or separate method - more than 30% of the required steps are already part of the product development process (e.g. requirements engineering, quality function deployment, function analysis) but they are not clearly and continuously linked to modularisation - Excellent and stable market input including such tools market segmentation/positioning as a base for modularisation, rather mature markets - Internal modularisation coaches - Establish awareness and culture of modularisation and complexity management including necessity for frontloading - Strong management commitment and stringency (see good example at site XY) - Stringent rules and discipline concerning modules and complexity management - Establish module organisation and module administration (module owner who protects modules and interfaces from unauthorised changes)	Diverse	Diverse	n.a.	Diverse	Modular System Scope	Diverse			
152	Primary case A	Conditions to realize complexity reduction potentials: - Launch of modular system - Strong management commitment from top management and from all sites - Strong cooperation with product management - Build up know-how - Full integration into processes, IT and organisation	Diverse	Diverse	n.a.	Diverse	Modular System Scope	Diverse			

Id	Disguised Source	Support	Product Design / Life cycle Phase	Aspect Classifi-cation	Sorting Classifica-tion	Company Function	Scope Classifica-tion	Modular System De-sign / Life cycle Phase	Issue	Impor-tant Factor	Contra-dictory to
153	Primary case A	BU-approach to collect synergies in product lines, even though cross-BU synergies and systems approach might fall short	Diverse	Organisa-tion	n.a.	Diverse	Modular System Scope	Diverse			
154	Primary case A	Build up internal knowledge for modularisation in order to spread it to product management and DEV (e.g. through central department and internal consultants)	Diverse	Introduc-tion	Change Manage-ment	Diverse	Modular System Scope	n.a.			
155	Secondary case	Domain dictionary or glossary or standardised entries in IT-systems	Diverse	Introduc-tion	Change Manage-ment	Diverse	n.a.	n.a.			
156	Secondary case	Different types of organisational scheme: - Distinguish between central development and project oriented development - assign following activities to one of above development types => organization ranges from centralized toward integrated and decentralized: bids & projects, project engineering, product/platform engineering, definition of product customization, application engineering module engineering, technology development => synergies versus products with very specific market requirements fulfilment	Diverse	Organisa-tion	n.a.	Diverse	Modular System Scope	Diverse			

Id	Disguised Source	Support	Product Design / Life cycle Phase	Aspect Classifi-cation	Sorting Classifica-tion	Company Function	Scope Classifica-tion	Modular System De-sign / Life cycle Phase	Issue	Impor-tant Factor	Contra-dictory to
157	Secondary case	Triangle representation: Relative weight of ... Platform engineering (resources organized by product line, priority on leveraging product line benefits) VERSUS Project engineering (resources dedicated to specific pro-jects, priority on project delivery and client proximity) VERSUS Engineering discipline (Sys, SW, HW) (resources pooled by technical competences, priority on skills, technologies and specialities transverse to products & projects) = With modular system development: shift from projects toward platform while keeping engineering disciplines stable.	Diverse	Organisa-tion	n.a.	Diverse	Modular System Scope	Diverse			
158	Secondary case	Intra-group sharing: - Product line responsible to address product sharing opportunities within company - a library of components or providing access to platform-related resources (documents, code, test results…)	Diverse	Diverse	n.a.	Diverse	Modular System Scope	Diverse			
159	Secondary case	Different ways of how to transition: - develop the modular system first: develop the scope first and use it as a mission statement. When the core assets are developed, products may come quickly to market with minimum development. Requires upfront investment and predictive knowledge. - starting with one or more products: from them, generate the product line core assets and then future products; the scope may evolve significantly. Requires, to start with, a base line robust, extensible and appropriate to future product line needs.	Diverse	Introduc-tion	n.a.	Diverse	n.a.	n.a.			

Id	Disguised Source	Support	Product Design / Life cycle Phase	Aspect Classification	Sorting Classification	Company Function	Scope Classification	Modular System Design / Life cycle Phase	Issue	Important Factor	Contradictory to
160	Primary case A	Implementation schedule for modularisation see elements and timeline of different implementation charts of supporting consultancies and what was done at Primary case A	Diverse	Introduction	n.a.	n.a.	n.a.	n.a.			
161	Primary case C	Modularisation has to be coordinated with other processes: - e.g. validating features with customers during user experience - validation and testing in order to get feedback on feasibility of modular system	Diverse	Process	n.a.	Diverse	Modular System Scope	Diverse			
162	Primary case A	See overall framework for modularisation, standardisation variant management (see Wildemann Ordnungsrahmen)	Diverse	Diverse	n.a.	Diverse	Modular System Scope	Diverse			

Appendix E: Pre-selected questionnaire for metric requirement prioritisation within company

The following Table E-I shows a questionnaire which was used to prioritise requirements for evaluation of modularisation within the primary case company. The questions were already pre-selected in order to ensure a high reply rate (i.e. adequate number of questions to be asked) and to make sure that also non-modularisation experts can contribute to the quantified requirements collection (i.e. not too deep, abstract or theoretical). Several modularisation-savvy engineering managers contributed to the pre-selection of requirements during iterative sessions. The questions were sent out by email to mainly engineering managers and engineers. Results of the requirements collection are given in Section 7.3.1. Amongst other criteria, these results have been used to develop and validate the modularisation metrics of Chapter 7. Another purpose of the requirements prioritisation was to collect awareness and justification for later implementation of the metrics within the company.

Table E-I: Questionnaire for requirements prioritisation of modularisation metrics

Requirement Prioritization for Evaluation of Modularization

Purpose:

In order to develop a company-wide evaluation approach, the aim of this prioritization sheet is to find out *what* you want to evaluate and how you want to evaluate modularization within the company (e.g. process audits vs. measuring one technical property)

Instructions:

Please prioritize preselected requirements by assigning the status "low", "medium", or "high" in the priority column. Free rows can be used for adding requirements from your side. You may also add additional comments at the end of this sheet.

Requirements on "what" to evaluate of modularization	Priority
Evaluation has to consider the modular **platform** (e.g. appliance or system platform)	
Evaluation has to consider the **project** (e.g. one or several appliances or systems)	
Evaluation has to consider single **products** (e.g. one appliance or system)	
Evaluation has to consider the modularization **process**	
Evaluation has to consider how well **roles and responsibilities** are aligned to modularization	
Evaluation has to consider how much **modularization knowledge** is available in the organization	
Evaluation has to consider **external** variance (e.g. # product variants)	
Evaluation has to consider **internal** complexity (e.g. # of parts)	
Evaluation has to consider **reuse** of e.g. modules/interfaces for next generation or cross products/platforms	
Evaluation has to consider the point of **variance creation** in the production sequence	
Please feel free to add additional requirements here if needed	
Please feel free to add additional requirements here if needed	

Requirement on "how" to evaluate modularization	Priority
Evaluation has to consider **one** input factor (e.g. # parts)	
Evaluation has to consider **more than one** input factors (e.g. # parts, # variants, and # interfaces)	
Evaluation has to be based on **qualitative** criteria	
Evaluation has to be based on **quantitative** criteria	
Evaluation has to be done with **one single key figure**	
Evaluation has to be done with a **key figure system**	
Evaluation has to be done through **integration in milestone reviews or quality gates (QG)**	
Evaluation has to be done by a **"stand-alone" assessment**	
Evaluation has to be done by a **neutral assessor**	
Evaluation has to be done by business unit **or site** in the course of the project	
Evaluation has to be done **once** in a project	
Evaluation has to be done **several times** in a project (e.g. in all or several QGs)	
Evaluation has to be done **several times** during **product architecture lifecycle** (sustainability of architecture)	
Evaluation has to be aggregated from **product/project** to business unit or company **level**	
Evaluation has to compare **sequential projects** (e.g. compare 1st and 2nd generation)	
Please feel free to add additional requirements here if needed	
Please feel free to add additional requirements here if needed	

Additional comments:

Please send the completed requirement prioritization list back to [Organization] Heilemann.
Due date: Highly appreciated until [Date] (Latest submission: [Date]).
In case of questions or further comments, please do not hesitate to contact [Organization]-Heilemann.

Thank you very much for your kind support!

Appendix F: Overview of selected existing modularisation metrics

The following Table F-I shows the formulas of selected existing modularisation metrics of the state of the art Section 7.2. The focus here is on metrics assessing modularity principles and on complexity metrics. These are the categories to which the developed architecture-related metrics (see Section 7.4.5) and result-oriented metrics (see Section 7.4.4) could be assigned to. It is the purpose of this appendix to give readers some sort of feeling how the characteristics of the developed modularisation metrics of Section 7.4 are compared to existing modularisation metrics. It is not the purpose of this appendix to provide a complete overview of all existing modularisation metrics.

Table F-1: Formulas for existing modularisation metrics

References	Formula of metric
<u>Assessment of modularity principles</u>	
<u>Functional structures:</u>	
Holtta and Otto (2005, 2003)	$\text{Complexity metric} = \frac{\text{estimated degree of additional design work}(\%)}{\text{degree of change of functional flow}(\%)}$ e.g. a 1 % change in a functional flow requires 1,5 % (0-3 % in general) re-design effort of the original design effort of the module, i.e. the complexity metric is 1,5
Stone (1999)	$\Lambda = \hat{N}^T \hat{N}$ = customer weighted sub-function similarity, λ_{ij} = elements of Λ, projection of the *i*th product on the *j*th product = similarity index Where: $\hat{N}$ = matrix of unity normalized product vectors (each vector of N is renormalized to one), N = normalized version of Φ, Φ = m x n product-function matrix with elements Φ_{ij} for the *i*th function of the *j*th product and with m as the total number of different sub-functions for n products, v_{ij} = elements of N $v_{ij} = \Phi_{ij}\left(\frac{\bar{\eta}}{\eta_j}\right)\cdot\left(\frac{\mu_j}{\bar{\mu}}\right)$, $\bar{\eta} = \frac{1}{n}\sum_{i=1}^{m}\sum_{j=1}^{n}\Phi_{ij}$ = average need rating, $\eta_j = \sum_{i=1}^{m}\Phi_{ij}$ = total customer need rating for *j*th product, $\mu_j = \sum_{i=1}^{m}H(\Phi_{ij})$ number of functions in the *j*th product, $\bar{\mu} = \frac{1}{n}\sum_{i=1}^{m}\sum_{j=1}^{n}H(\Phi_{ij})$ = average number of functions, H = Heaviside step function or unit step function whose value is zero for negative argument and one for posi-

	tive argument
Stone (1999)	$s_j = \frac{1}{n}\sum_{p=1}^{n}\sqrt[f_j]{\prod_{i=1}^{f_j} v_{ip}}$ = aggregate customer need rating for module j Where: n = number of products, g = number of modules in the product under investigation, j = 1...g indicates the module to which the value corresponds, f_j = the number of sub-function in module j, v_{ip} = elements of N corresponding to the *i*th sub-function of the module *j* in the *p*th product
Functional-physical relations:	
Steva et al. (2006)	F-C_i = $\frac{\sum_{j=1}^{n} CF_{ij}}{\text{number of functions (n) performed by component i}}$, F-$C_i$ = Function-Component Frequency of component i, CF_{ij} = $\frac{\text{\# products where component i performs function j}}{\text{total number of products}}$, CF_{ij} = Component Frequency Score of component i for function j, e.g. The shutter regulates electromagnetic energy in 100 % of the products and actuates mechanical energy in 50 % of the products which makes a F-C_i of (100 + 50)/2 = 75 %. This kind of analysis can be taken to analyse which parts or functions to include into the platform.
Mattson and Magleby (2001)	Modularity metric (M) = $\frac{N_{modules}}{N_{functions}}$ e.g. M = 1 for maximum modularity while the degree of modularity decreases with the value of M
Physical interactions between elements:	
Gershenson et al. (1999)	Relative Modularity = $\frac{S_{in}}{S_{in}+S_{out}} + \frac{D_{in}}{D_{in}+D_{out}}$ Where: S_{in} = Similarity between components within a module, S_{out} = Similarity between a component of a concerned module and other components outside of the module D_{in} = Dependency between components within the module D_{out} = Dependency between a component within a module and a component outside of the module
Guo and Gershenson (2004)	$MM = \frac{\sum_{k=1}^{M}\left(\sum_{i=n_k}^{m_k}\sum_{j=n_k}^{m_k} R_{ij} / (m_k - n_k + 1)^2\right) - \sum_{k=1}^{M}\left(\sum_{i=n_k}^{m_k}\left(\sum_{j=1}^{n_k-1} R_{ij} + \sum_{J=m_k+1}^{N} R_{ij}\right) / (m_k - n_k + 1)(N - m_k + n_k - 1)\right)}{M}$ Where: MM = Modularity Metric measuring n_k = index of first component in *k*th

	coupling between components inside and between modules in a modularity matrix/DSM, R*ij* = each value of the *i*th row and *j*th column in the matrix	module, m_k = index of last component in *k*th module, M = total number of modules in the product, N = total number of components in the product
Holtta-Otto and De Weck (2007)	$SMI = \frac{1}{N} \arg\min_{\alpha} \sum_{i=1}^{N} \left| \frac{\sigma_i}{\sigma_1} - e^{-[i-1]/\alpha} \right|$ Where: SMI = Singular Value Modularity Index which measures the degree of modularity and type of modularity of a product based on its internal connectivity structure, N= number of components = number of rows and columns in the DSM_{ij} with the *i*th row and *j*th column,	σ_i = singular values 1 to N which are the square roots of the eigenvalues of $DSM^T DSM$ and corresponding orthogonal eigenvectors, they are obtained by performing a singular value decomposition on the binary DSM matrix showing connections between components of the product
Holtta-Otto and De Weck (2007)	$NZF = \frac{\sum_{i=1}^{N}\sum_{j=1}^{N} DSM_{ij}}{N(N-1)}$ Where: NZF = Non-Zero Fraction which is the fraction of non-zero entries in the DSM, excluding the diagonal, measuring sparsity of connections of a system,	N= number of components = number of rows and columns in the DSM_{ij} with the *i*th row and *j*th column
Mikkola and Gassmann (2003), Mikkola (2006)	$M(n_{NTF}) = e^{-n_{NTF}^2/2Ns\delta}$ Where: $M(n_{NTF})$ = modularisation function measuring degree of modularisation embedded in product architectures, n_{NTF} = number of new-to-the-firm (NTF) components, N= total number of components, s = substitutability factor, $s = \frac{\text{no. of product families}}{k_{NTF}(avg)} = \frac{\sum_{j=1}^{L} PF_j}{\frac{\sum_{i=1}^{K} k_{NTF}}{K}}$, Where: L = number of product families PF_j, K = total number of interfaces of NTF components, k_{NTF} = interfaces of NTF components,	δ= degree of coupling, $\delta_i = \frac{\sum k_c}{n_c}$ $\delta_{subsystem} = \frac{\sum_{i=1}^{I} \delta_i}{I}$ Where: I = number of subsystems i k_c = total number of interfaces in subsystem i n_c = number of components in subsystem i
Martin	Based on a matrix representing estimated strength of couplings between	

and Ishii (2002)	components (on a stepwise "unregular" scale from zero to nine), the researchers calculate two coupling indices: CI-S = The coupling index–supplying indicates the strength (or impact) of the specifications that a component supplies to other components. CI-R = The coupling index–receiving indicates the strength (or impact) of the specifications that a component receives from other components.
Sosa et al. (2005)	$M(ID)_i = \frac{x_{\max} \cdot (n-1)}{x_{i+}}$ Where: $M(ID)_i$ = In-Degree Modularity of component i, it is equal to the number of other components that component i depends on for functionality, n= number of components of product, , the higher the metric the higher the degree of modularity, $x_{i+} = \sum_{j=1}^{n} X_{ij}$, x_{max} = maximum value that X_{ij} can take, X_{ij} = non-zero elements of design dependency matrix X, if component *i* depends for functionality on component *j*, thus it indicated the strength of the design dependency
Sosa et al. (2005)	$M(OD)_i = \frac{x_{\max} \cdot (n-1)}{x_{+i}}$ Where: $M(OD)_i$ = Out-Degree Modularity, i.e. the number of other components *j* that depend on component *i*, the higher the metric the higher the degree of modularity, n= number of components of product, $x_{+i} = \sum_{j=1}^{n} X_{ji}$, x_{max} = maximum value that X_{ji} can take, X_{ji} = non-zero elements of design dependency matrix X, if components *j* depend for functionality on component *i*, thus it indicated the strength of the design dependency
Sosa et al. (2005)	$M(IT)_i = \frac{\sum_{j=1}^{n} d(i,j)}{n-1}$ Where: $M(IT)_i$ = In-Distance Modularity, measures how distant component *i* is from all other components in the product, the more distant the component is the more modular the product, n = number of components in product, d(i,j) = geodesic of design dependency between component *i* and component *j*
Sosa et al. (2005)	$M(OT)_i = \frac{\sum_{j=1}^{n} d(j,i)}{n-1}$ Where: $M(OT)_i$ = Out-Distance Modularity, measures how distant components *j* are from component *i*, the more distant the components are the more modular

	the product is, d(j,i) = geodesic of design dependency between components *j* and component *i*
Sosa et al. (2005)	$$M(B)_i = \frac{[(n-1)(n-2)/2]}{\sum_{a<b,i\neq a,i\neq b} nd_{ab}(i)/nd_{ab}}$$ Where: $M(B)_i$ = Bridge Modularity of component *i* based on the number of times it is on the path of two other components, components lying on most geodesics are the one bridging most components and therefore are least modular, n = total number of connected components in product, $nd_{ab}(i)$ = total number of geodesics between two components, a and b, which contain component *i*, nd_{ab} = total number of geodesics between component a and b
Allen and Carlson-Skalak (1998) in Holtta-Otto (2012)	$$Interactions = \frac{\sum_{k=1}^{M}\left(\sum_{i=n_k}^{m_k}\left(\sum_{j=1}^{n_k-1}R_{ij} + \sum_{j=m_k}^{N}R_{ij}\right)\right)}{M} = \frac{\#\text{ inter module interactions}}{\#\text{ of modules}}$$ Where: "Interactions" = ratio of interaction inside a module to the total number of modules, M = total number of modules, N = total number of components in the product, n_k = index of the first component in the *k*th module, m_k = index of the last component in the *k*th module, R_{ij} = value of the *i*th row and *j*th column element in the modularity matrix
Whitney et al. (1999) in Holtta-Otto (2012)	$$WI = \frac{\sum_{k=1}^{M}\left(\sum_{i=n_k}^{m_k}\sum_{j=n_k}^{m_k}R_{ij} + \left(\sum_{i=n_k}^{m_k}\left(\sum_{j=1}^{n_k-1}R_{ij} + \sum_{j=m+1}^{N}R_{ij}\right)\right)\right)}{N} = \frac{\#\text{interactions in DSM}}{\#\text{elements in DSM}}$$ Where: WI = Whitney Index which measures the ratio of interactions in a modularity matrix to the number of elements in a modularity matrix, M = total number of modules, N = total number of components in the product, n_k = index of the first component in the *k*th module, m_k = index of the last component in the *k*th module, R_{ij} = value of the *i*th row and *j*th column element in the modularity matrix
Sosa and Eppinger (2003) in Holtta-Otto (2012)	$$Modularity = \frac{\sum_{k=1}^{M}\left(\sum_{i=n_k}^{m_k}\left(\sum_{j=1}^{n_k-1}R_{ij} + \sum_{j=m_k}^{N}R_{ij}\right)\right)}{\sum_{k=1}^{M}\left(\sum_{i=n_k}^{m_k}\sum_{j=n_k}^{m_k}R_{ij} + \left(\sum_{i=n_k}^{m_k}\left(\sum_{j=1}^{n_k-1}R_{ij} + \sum_{j=m+1}^{N}R_{ij}\right)\right)\right)}$$ $$= \frac{\#\text{ interactions in modules}}{\#\text{ interactions in DSM}}$$ Where: Modularity = ration of interactions in modules to the total number of interactions in the DSM, M = total number of modules, N = total number of components in the product, n_k = index of the first component in the *k*th module, m_k = index of the last component in the *k*th module, R_{ij} = value of the *i*th row and *j*th column element in the modularity matrix

Whitfield et al. (2002) in Holtta-Otto (2012)	$MSI = \frac{\sum_{i=n_i}^{n_2}\sum_{j=n_1}^{n_2} w_{i,j}}{(n_2-n_1)^2-(n_2-n_1)} - \left(\frac{\sum_{i=0}^{n_1}\sum_{j=n_1}^{n_2} w_{i,j}+w_{i,j}}{2n_1(n_2-n_1)} + \frac{\sum_{i=n_2}^{N}\sum_{j=n_1}^{n_2} w_{i,j}+w_{i,j}}{2(N-n_2)(n_2-n_1)} \right)$ Where: MSI = Module Strength Index for estimating the goodness of a single module, n_1 = index of the first components in the module, n_2 = index of the last components in module, N = number of components in the DSM, w_{ij} = dependency weights while *i* and *j* are column indices
Yu et al. (2005) in Holtta-Otto (2012)	$MDL = \frac{1}{3}\left(n_c \log n_n + \log n_n \sum_{i=1}^{n_c} cl_i \right) + \frac{1}{3}S_1 + \frac{1}{3}S_2$ Where: MDL = Minimum Description Length based on the information needed to describe a modularity matrix, n_c = number of modules, n_n = number of rows or columns in the DSM, cl_i = size of module *i*, S_1 and S_2 measure additional information needed to describe the DSM beyond listing the module and bus numbers in sizes, in short they measure: S_1 = number of cells that are in a module or on a bus, but are empty, S_2 = number of cells that is one in between the modules and buses, n.b. above mentioned equation has been simplified by Holtta-Otto (2012) by substituting equal weights 1/3 for all terms in the overall equation as suggested by Yu et al. (2005)
Assessment of internal complexity / commonality	
Collier (1982, 1981)	$DCI = \frac{\sum_{j=1}^{d}\phi_j}{d}, \quad 1 < C < \beta$ Where: DCI = Degree of Commonality Index, Φ_j = the number of immediate parents component *j* has over a set of end-items or product structure level(s), d = total number of distinct components in the set of end-items or product structure level(s), β= $\sum_{j=1}^{d}\phi_j$ = the total number of immediate parents for all distinct component parts over a set of end-items or product structure level(s)
Wacker and Treleven (1986)	$TCCI = 1 - \frac{d-1}{\sum_{j=1}^{d}\Phi_j - 1}$ Where: TCCI = Total Constant Commonality Index, Φ_j = the number of immediate parents component *j* has over a set of end-items or product structure level(s), d = total number of distinct components in the set of end-items or product structure level(s)

Martin and Ishii (1996)	$CI = \frac{u}{\sum_{j=1}^{V_n} p_j}, \quad 0 < CI \leq 1$ Where: CI = Commonality Index, u = # unique part numbers, p_j = # parts in model *j*, v_n = final # of varieties offered
Beisheim and Stotz (2013)	SDU = Standardisation Degree regarding part usage SDC = Standardisation Degree regarding part consumption SD = Standardisation Degree overall
Sinigalias and Dent-soras (2015)	$I_c(a) = \frac{w_m(a) + w_s I_s(a)}{w_m + w_s}, \quad I_c(a) \in [0,1]$ Where: $I_c(a)$ = Composite Standardisation Index, i.e. percentage of common parts being used in the system, $I_m(a)$ = Commonality Index for the assembly a, i.e. compliance of all parts with the pertinent standards, $I_s(a)$ = Absolute Standardisation Index for the assembly a, w_m = weight factor for commonality index of assembly a, w_s = weight factor for absolute standardisation index of assembly a
Kota et al. (2000)	$PCI = \frac{\sum_{i=1}^{P} CCI_i - \sum_{i=1}^{P} MinCCI_i}{\sum_{i=1}^{P} MaxCCI_i - \sum_{i=1}^{P} MinCCI_i} \cdot 100$ $PCI = \frac{\sum_{i=1}^{P} n_i \cdot f_{1i} \cdot f_{2i} \cdot f_{3i} - \sum_{i=1}^{P} \frac{1}{n_i^2}}{(P \cdot N) - \sum_{i=1}^{P} \frac{1}{n_i^2}} \cdot 100$ Where: PCI = Product Line Commonality Index, CCI_i = Component Commonality Index for component i = $n_i \cdot f_{1i} \cdot f_{2i} \cdot f_{3i}$, $MaxCCI_i$ = Maximum possible Component Commonality Index for component *i* = N, $MinCCI_i$ = Minimum possible Component Commonality Index for component *i* = $n_i \cdot \frac{1}{n_i} \cdot \frac{1}{n_i} \cdot \frac{1}{n_i} = \frac{1}{n_i^2}$, P = Total number of non-differentiating components that can potentially be standardised across models, N = Number of products in the product family, n_i = Number of products in the product family that have component *i*, f_{1i} = Size and shape factor for component *i*, f_{2i} = Materials and manufacturing processes factor for component i, f_{3i} = Assembly and fastening schemes factor for component *i*
Siddique et al. (1998), slightly adapted	$\%C = \sum_{i=1}^{4} I_i \cdot C_i = I_c \cdot C_c + I_n \cdot C_n + I_l \cdot C_l + I_a \cdot C_a$ Where:

by Thevenot and Simpson (2006)	%C = Percent Commonality Index, I_i = Importances (Weighting Factors), $C_c = \frac{100 \cdot \text{common components}}{\text{common components} + \text{unique components}} =$ Commonality of Components, $C_n = \frac{100 \cdot \text{common connections}}{\text{common connections} + \text{unique connections}} =$ Commonality of Connections, $C_l = \frac{100 \cdot \text{common assembly component loading}}{\text{common assembly loading} + \text{unique assembly loading}} =$ Component Loading Sequence, $C_a = \frac{100 \cdot \text{common assembly workstations}}{\text{common workstations} + \text{unique workstations}} =$ Assembly WorkStation Commonality
Jiao and Tseng (2000)	$CI^{(C)} = \frac{\sum_{j=1}^{d}\left(P_j \sum_{i=1}^{m}\phi_{ij}\sum_{i=1}^{m}\left(V_i Q_{ij}\right)\right)}{\sum_{j=1}^{d}\left(P_j \sum_{j=1}^{m}\left(V_i Q_{ij}\right)\right)}, \quad 1 \le CI^{(C)} \le \alpha$ Where: $CI^{(C)}$ = Component Part Commonality Index, d = total number of distinct component parts used in all the product structures of a product family, j = index of each distinct component part d_j, $\forall j = 1,2,...,d$, P_j = price or estimated cost of each component part, m = total number of end products in a product family, i = index of each member product of a product family, $\forall i = 1,2,...,m$, Φ_{ij} = the number of immediate parents for each distinct component part d_j over all the product levels of product *i* of the family, V_i = volume of end product *i* in the family, Q_{ij} = quantity of distinct component part d_j required by product *i*, this can also be calculated by multiplying quantity per operation *q* through the levels of the product tree, i.e. $Q_{ij} = \sum_{h=1}^{n_h}\left(\prod_{k=0}^{n_k} q_{hk}\right), \quad \forall h = 1,2,...,n_h \text{ and } \forall k = 0,1,2,...,n_k \quad \text{for each item } d_i \text{ in product i}$ h = one particular path from the item d_j to the end item node through the levels of the product tree for a particular end product in the family, n_h = total number of paths for d_j within product *i,* n_k = total number of parent nodes on path *h,* k = index of the nodes on path *h*, q_{hk} = quantity per operation (either manufacturing or assembly) of node k required by its immediate parent node along path h
Johnson and Kirchain (2010)	$C_{Piece} = \frac{\sum_{j=1}^{m}\left(\frac{\sum_{j=1}^{m}\gamma_{ij} - 1}{m-1}\right)}{d}$ $\qquad C_{\Phi} = \frac{\sum_{i=1}^{d}\left(\frac{\sum_{j=1}^{m}\gamma_{ij} - 1}{m-1}\right)}{\sum_{i=1}^{d}\phi_i}$ C_{Piece} = Piece-based commonality

	metric measuring whether a part is shared C_{Φ} = Weighted commonality metric with following suggestions to be weighted: $C_{\Phi=Mass}$ = Mass-weighted metric based on the relative mass of a component $C_{\Phi=Cost}$ = Cost-weighted metric based on the fabrication cost per piece $C_{\Phi=Invest}$ = Investment-weighted metric based on the fabrication investment required for a component $C_{PV}=\frac{\sum_{i=1}^{d}\left(\left(\frac{\sum_{j=1}^{m}\gamma_{ij}-1}{m-1}\right)\left(\frac{\sum_{j=1}^{m}PV_{ij}-PV_{\min}}{PV_{Tot}-PV_{\min}}\right)\right)}{d}$ C_{PV} = Production volume-weighted metric calculated using the relative production volumes required for each variant $C_{PV/\phi}=\frac{\sum_{i=1}^{d}\left(\phi_i\left(\frac{\sum_{j=1}^{m}\gamma_{ij}-1}{m-1}\right)\left(\frac{\sum_{j=1}^{m}PV_{ij}-PV_{\min}}{PV_{Tot}-PV_{\min}}\right)\right)}{\sum_{i=1}^{d}\phi_i}$ $C_{PV/\Phi=invest}$ = Production volume/investment-weighted metric combining relative production volume and fabrication investment weightings Where: γ_{ij} = binary variable, $\gamma = 1$ if variant *j* contains component *i*, $\gamma = 0$ if this is not the case, m = total number of product variants, d = number of distinct items in the bill of material, Φ_i = weighting factor reflecting the importance of component *i* like mass, piece cost or fabrication investment
Blecker and Abdelkafi (2007)	$TCI=\left(\frac{1}{n}\sum_{i=1}^{n}\prod_{k=1}^{n_{h_i}}(N_k)_{h_i}\sum_{j=1}^{n_i}\frac{\omega_{ij}^2}{n_i}\right)+\left(\frac{A}{m}\sum_{i=1}^{m}\alpha_i\prod_{k=1}^{n_{h_i}}(N_k)_{h_i}\sum_{j=1}^{m_i}\frac{\omega_{ij}^2}{m_i}\right),\quad where$ $\left(\frac{1}{n}\sum_{i=1}^{n}\prod_{k=1}^{n_{h_i}}\frac{(N_k)_{h_i}}{n_i^2}\right)+\left(\frac{A}{m}\sum_{i=1}^{m}\alpha_i\prod_{k=1}^{n_{h_i}}\frac{(N_k)_{h_i}}{m_i^2}\right)\leq TCI\leq$ $\left(\frac{1}{n}\sum_{i=1}^{n}\prod_{i=1}^{n_{h_i}}\frac{(N_k)_{h_i}}{n_i}\right)+\left(\frac{A}{m}\sum_{i=1}^{m}\alpha_i\prod_{k=1}^{n_{h_i}}\frac{(N_k)_{h_i}}{m_i}\right)$ Where: n = Number of must-generic items, m = Number of option categories (can-generic items), m_i = Number of variations in an option category, h_i = Path in the generic BOM from node i to end product, $(N_k)_{hi}$ = Quantity per operation of node k required by its immediate parent node along path h_i, A = Probability that the end product is equipped with an option,

	α_i = Conditional probability that the end product is equipped with option category i, knowing that the end product is equipped with an option
Romanowski and Nagi (2005)	$S = S_{ab} = \frac{G_a \oplus G_b}{\lvert V_{\acute{a}} \cup V_b \rvert^2} = \frac{D_j}{n^2}$ Where: S = Sparsity value, calculated for the sparsity matrix D_{ab}, which represents the ration of the sum of non-zero entries to the toral number of entries in the matrix. This represents the similarity between two BOM-trees, D_j = Sum of non-zero entries in D_{ab}, n^2 = Number of entries in the matrix, D_{ab} = Delta-matrix between the two adjacency matrics A and B, $D_{ab} = \lvert A - B \rvert$, A = Smaller adjacency matrix of G_a and G_b, constructed of a graph G, B = Larger adjacency matrix of G_a and G_b, constructed of a graph G, $G_a = G_a(V_a,E_a)$, $G_b = G_b(V_b,E_b)$, V = set of vertices (nodes in a graph representing the BOM tree, E = set of edges (arcs) in a graph representing the BOM tree, G(V,E) = a graph made up of vertices V and edges E, $\oplus$ = ring-sum operator, $\cup$ = union operator
Thevenot and Simpson (2007)	$CMC = \frac{\sum_{i=1}^{P} n_i \cdot (C_i^{\max} - C_i) \cdot \prod_{x=1}^{4} f_{xi}}{\sum_{i=1}^{P} n_i \cdot (C_i^{\max} - C_i^{\min}) \cdot \prod_{x=1}^{4} f_{xi}^{\max}}$ Where: P = Total number of components, n_i = Number of products in the product family that have component i, f_{1i} = Ratio of the greatest number of products that share component *i* with identical size and shape to the number of products that have component *i* (n_i), f_{2i} = Ratio of the greatest number of products that share component *i* with identical materials to the number of products that have component *i* (n_i), f_{3i} = Ratio of the greatest number of products that share component *i* with identical manufacturing processes to the number of products that have component *i* (n_i), f_{4i} = Ratio of the greatest number of products that share componenti with identical assembly and fastening schemes to the number of products that have component *i* (n_i), $f_{1i}^{\max}$ = Ratio of the greatest number of products that share component *i* with identical size and shape to the greatest possible products that could have shared component *i* with identical size and shape schemes, $f_{2i}^{\max}$ = Ration of the greatest number of products that share component *i* with identical materials to the greatest possible number of products that could have shared component i with identical materials, $f_{3i}^{\max}$ = Ratio of the greatest number of products that share component *i* with identical manufacturing processes to the greatest possible number of products that could have shared component *i* with identical manufacturing processes, $f_{4i}^{\max}$ = Ratio of the greatest number of products that share component *i* with identical assembly and fastening schemes to the greatest possible number of products that could have shared component *i* with identical assembly and

	fastening schemes, C_i = Total cost for component *i*, $C_i = \sum_{j=1}^{n_i} C_{ij}$, where C_{ij} is the total cost for component i variant j, $C_i^{\min}$ = Minimum total cost for component *i* (obtained when the component is common between all the products having component i), $C_i^{\max}$ = Maximum total component cost (obtained when the component is variant in each of the products having component *i*)
Alizon et al. (2009b)	$CDI_{Family_P} = \frac{1}{F}\sum_{i=1}^{F}\frac{1}{K_{ij}}\sum_{k=1}^{K_{ij}}\frac{1}{G_{ik}}\cdot\sum_{m=1}^{G_{ik}}\left(1-\frac{non\_allowed\_com\_div_{ikg_m}}{\max div_{ikg_m}}\right)$ Where: CDI_{Family_P} = Commonality-Diversity-Index for the family of products P, F = Number of functions in the family, K_{ij} = Component *j* of function *i*, G_{ik} = Subgroup *k* of components of function *i*, $non\_allowed\_com\_div_{ikg_m}$ = Non-allowed commonality/diversity for subgroup g_m, $\max div_{ikg_m}$ = Ideal maximum diversity for subgroup g_m