ICED13
INTERNATIONAL CONFERENCE ON ENGINEERING DESIGN, ICED13
19-22 AUGUST 2013, SUNGKYUNKWAN UNIVERSITY, SEOUL, KOREA

TOWARDS THE DEVELOPMENT OF COMMONAL PRODUCT PROGRAMS

Sandra EILMUS, Sebastian RIPPERDA, Dieter KRAUSE
Hamburg University of Technology, Germany

ABSTRACT
Today many companies encounter the challenge of reducing internal variety by developing modular product families. Yet even more potential for reducing internal variety can be gained by even striving for carry-over across product families as industrial case studies showed. A holistic corporate product program strategy for reducing variety supports the identification and exploitation of this potential. How this strategy can be defined in specific corporate context is outlined in this paper. Based from the attributes of an ideal product program steps for definition of product structure strategies are proposed. The resulting development tasks need a deep understanding about a solution that suits corporate needs for differentiation and standardization best. This can be gained by analysis of tearing forces towards differentiation and standardization and comparison of possible solutions that enable commonality in different ways and by this have different effects. To support evaluation of these solutions a complexity cost estimation method is proposed complying with the lack of detailed information in the conceptual phase.

Keywords: product program, commonality, product families, product structuring, portfolio management

Contact:
Sandra Eilmus
Hamburg University of Technology
PKT Institute of Product Development and Mechanical Engineering Design
Hamburg
21073
Germany
sandra.eilmus@tuhh.de

1 POTENTIAL AND CHALLENGES IN PRODUCT PROGRAM DEVELOPMENT

The *Integrated PKT-Approach for Developing Modular Product Families* is a methodical toolkit that provides needed external variety based on possibly low internal product and process variety (Figure 1). It was developed at Institute PKT at Hamburg University of Technology (TUHH). The method units *Design for Variety* and *Life Phases Modularization* support the development of product families using low component variety and modular structures that enable specific corporate processes with possibly low effort. By implementing and evaluating these method units in seven industrial case studies their convenience in reducing variety was demonstrated (Eilmus 2012, Krause et al. 2013, Jonas 2012).

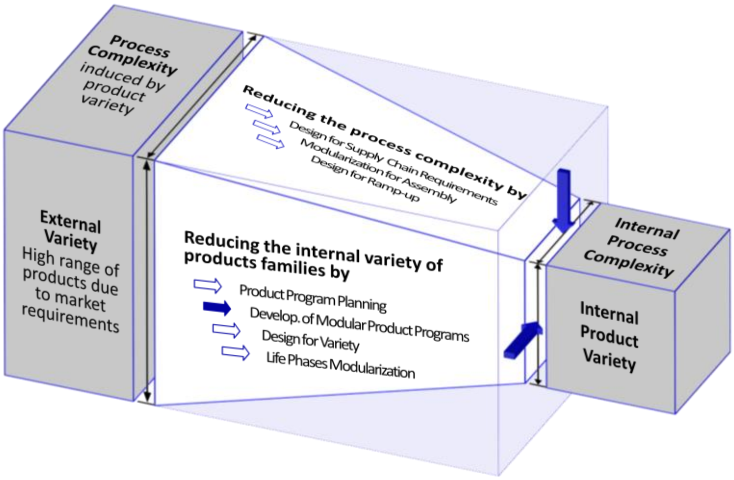


Figure 1. Integrated PKT-Approach for Developing Modular Product Families (Brosch 2012)

The industrial partners providing the case studies had a great need to embed optimisations of single product families into a holistic corporate product program strategy by developing single modules for implementation across several product families. These companies still have a large potential to reuse components or even modules across product families. This has not yet been exploited, as shown in three case studies that had a special focus on carry-over across product families (Figure 2).

		within product families				across product families		
		Herbicide spraying systems	Fork carrier	Lining	Gas detection devices	Displays	Operation devices	Gas inlet valves
Components in total	Existing	46	49	21	37	83	73	53
	New	32	28	13	27	17	28	22
Variant components	Existing	31	43	21	19	82	71	47
	New	14	19	4	14	12	13	9
Standardized components	Existing	15	6	0	18	1	2	6
	New	18	9	9	13	5	15	13

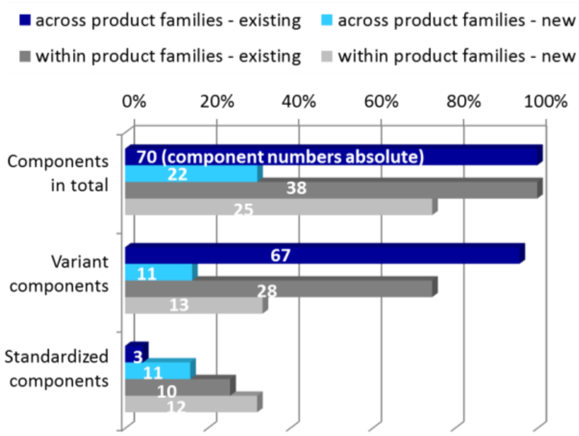


Figure 2. Application of the integrated PKT-approach within and across product families: results from case studies (Eilmus 2012, Krause 2013, Jonas 2012)

In these three case studies, use of the integrated PKT-approach produced an average reduction of variant components of about 80%. Even compared with case studies for variety reduction within product families there was higher potential in reducing the component number in total as well as reduction of variant components. This shows that there is substantial potential to reduce variety, which is not exploited by product family internal approaches. Further needs for research concerning the following questions were identified:

- What is the right strategy to define and exploit the carry-over potential for a product program?
- What is the most beneficial degree of variety when developing modules for use within and across product families?
- What is the most beneficial scope for carry-over of modules across several product families?

These questions show that there is a lack of understanding of what the ideal product program might look like and how to reach it. Answers found in literature are presented next, followed by a section describing the ideal commonal product program and a section presenting actions and methodical support for achieving the aim.

2 EXISTING METHODS FOR REDUCING VARIETY

As the development of product programs is only partly discussed in the literature, the literature research covers helpful sections of the development of product families and product structuring.

Development of product programs

Literature dealing with the development of product programs mainly concentrates on strategic product program planning, as summarised in Jonas (2012). The research project Product Program Planning by Jonas (2012) aims to enhance this planning at a product structure level, while taking into account the lowest possible internal variety by offering appropriate external variety. This methodical unit of the integrated PKT-approach leads to future scenarios and carry-over candidates, i.e. modules with high reuse potential across product families. These carry-over candidates are identified by analysing similarities in the differentiating product properties of the variants. The analysis of similar elements forms the core of a procedure for product program development, proposed by Blankenburg (1997), by analysing similarities in functions and technologies. Reitan (2002) proposes a generic procedure based on the product program-related characteristics complexity, variety, commonality and architecture, as introduced by Andreasen (2004). Existing methods from product family design and product design are recommended for the steps in this generic procedure. Jensen (2006) models the product program using a corporate platform to ‘focus on reuse of assets in a broader sense than traditional product platform development’. This corporate platform is described using the elements market, product and manufacturing. The literature described above works with an understanding of commonality (Blankenburg 1997, Reitan 2002, Jensen 2006). To better understand how this term may serve the Development of Modular Product Programs, it is discussed in the next section. The reviewed literature provides useful procedures for systematic variety analysis of a product program (Blankenburg 1997, Jensen 2006). Methods for exploiting this potential using appropriate design solutions are only given with a focus on a single component (Jensen 2006) or in a generic way (Reitan 2002). Whether these methods are suited to the extra complexity that a product program focus creates is not discussed. Engineering design support known from product family design is far more concrete, thus tools and procedures from this area that might support the development of product programs are summarised in the following.

Development of product families

There are various approaches to the development of modular product families with various focuses in the literature. They are described in depth by Krause (2013) and Zhila (2011). For the development of product programs, approaches that support the modelling of a product family and provide an overview of its configuration are interesting, like, for example, those proposed by Mortensen (2011), Jiao (2000) and Du (2001). Krause (2011), as described above, focuses on analysing product family components to redesign and them and cluster them in a modular structure. By doing this, separation into common and differentiating modules, as proposed by Mortensen (2011), Jiao (2000) and Du (2001), becomes possible for many modules for the first. However, to support engineering designers in developing product programs both handling the complexity of the task by modelling as well as enabling module reuse by appropriate module design are needed.

Product structuring

A product structure is the structured composition of a product by its components (Schuh 2005). Strategies to structure products can be summarised into three groups: modular product families, open modular systems and platforms (Pahl 2007, Schuh 2005). They have different benefits and effects, as elaborated in the next section. The decision which product structure to choose depends on the potential to exploit the benefits of the individual strategies and to cause the desired effect in the company.

Lindemann (2006) understands product structuring not only as an umbrella term for modular strategies but also as a preparatory activity that structures a range of products in order to understand standard, variant or individualised blocks as well as technical or organisational dependencies. This activity is necessary in order to build a modular structure afterwards. For the development of modular product programs this activity was understood as an important step to develop carry-over strategies in the industrial case studies described above. In a whole product program, decisions must be made on which modules to reuse in which product variants.

A helpful framework for this decision is given by Meyer (1997). It classifies the reuse according to the directions of reuse in a product program, as demonstrated in Figure 3 (left). Although Meyer (1997) focuses on platform development, the idea can be adapted to any carry-over activities in a product program. Figure 3 (right) shows a proposal for how to translate the idea of niche-specific, vertical and horizontal platforms to carry-over within and across product families as well as across product lines. The example demonstrates a section of a product program of industrial trucks. The industrial case studies showed that companies succeed in enabling carry-over in product families but that enabling carry-over across product families is difficult for carry-over within and across product lines. This is why this paper focuses mainly on the distinction between carry-over within and across product families, which includes carry-over within and across product lines (or vertical and horizontal carry-over).

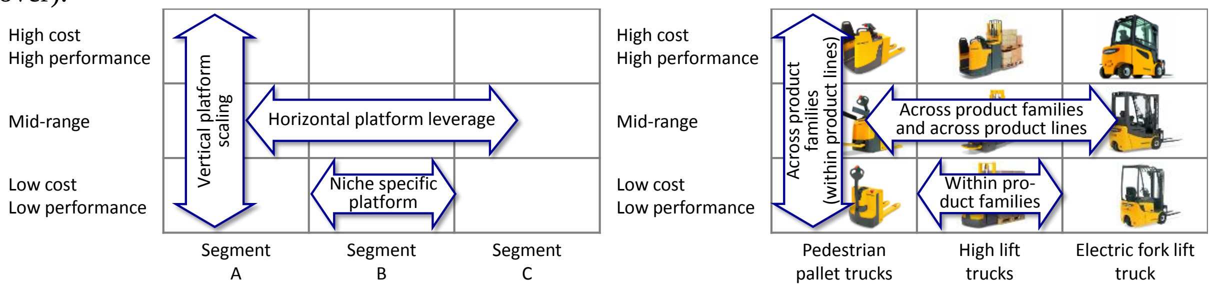


Figure 3. Platform leveraging strategies (Meyer 1997) (left) and translation to product program development using the example of industrial trucks (right)

3 THE IDEAL COMMONAL PRODUCT PROGRAM

In order to support the development of product programs an image of the ideal product program is needed. This is done in the following section, which also elaborates which roles the terms modularity and commonality have in this context.

Attributes of an ideal product program

From a corporate point of view, an ideal product program offers the product range required to address the target group (or groups) on the basis of potentially low costs. This implies that the required product range needs to be produced based on a product and process variety that causes as little effort as possible – yet fulfils the diverse customer requirements in a marketable way. In developing variant products, product development has to find a compromise between conflicting needs, as demonstrated in Figure 4.

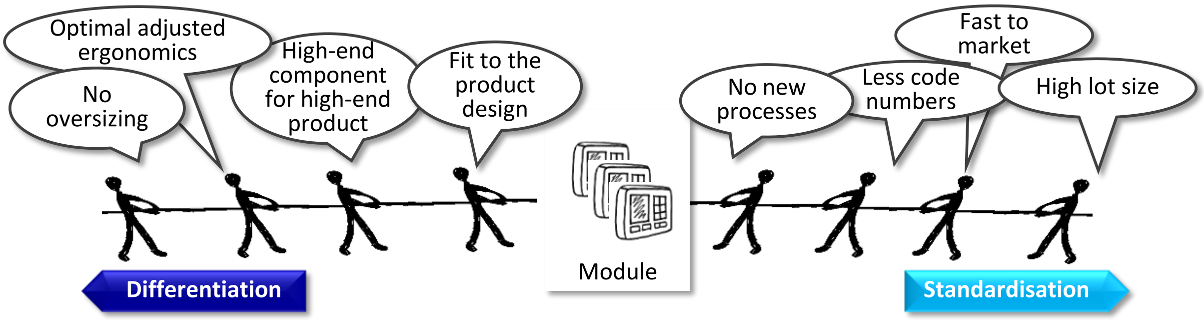


Figure 4. Conflicting needs for differentiation and standardisation (Eilmus 2010)

To fulfil diverse customer requirements sufficiently, several roles in the company advocate more and more differentiation between product variants. With possibly low costs in mind, other roles push for stronger standardisation of modules and processes. It is assumed that both parties represent qualified objectives of their company. Solving the conflict by deciding that one party is right might lead to a solution that does not fulfil the true corporate need. The ideal solution is a solution that satisfies the needs of all of the involved roles within both parties.

This conflict occurs not only when defining the variety of a module but also when discussing the scope of its reuse. Different product structure strategies can be chosen for a product program according to the corporate strategy. A classical product family oriented platform strategy (Figure 5, left) concentrating on module reuse (commonality) within product families can be followed. Other product programs are designed as a configurable modular system of smaller modules with a strong focus on carry-over across product families (Figure 5, right). A lot of product programs might show potential for both directions of commonality – within and across product families (Figure 5, middle). Yet the product structure strategy a company follows for the product program should enable the needed benefits and allow exploitation of the highest possible carry-over potential.

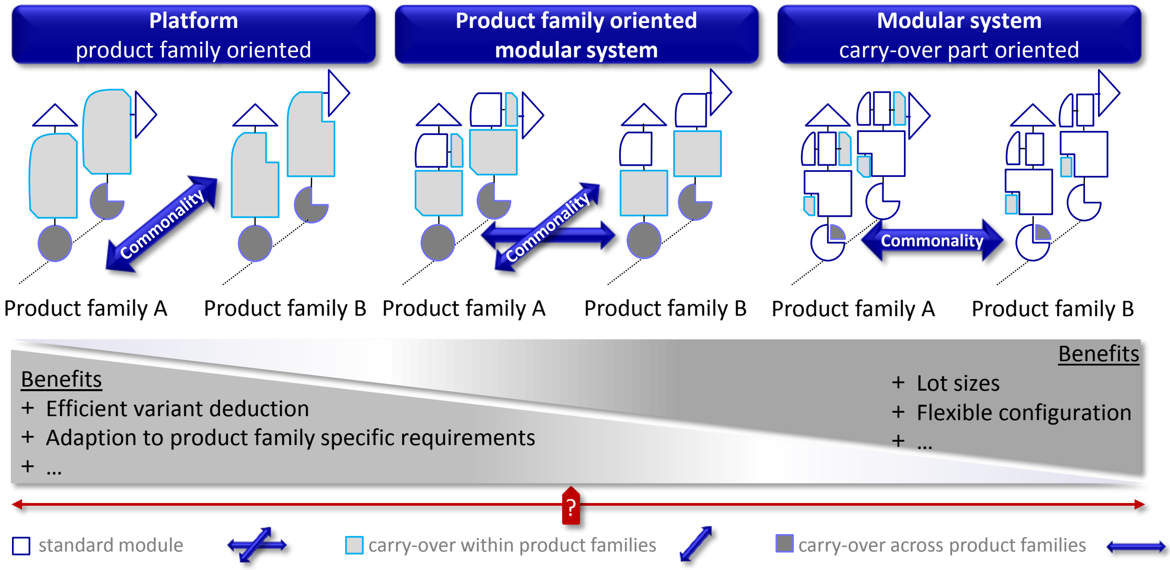


Figure 5. Product structure strategies and their focus within and across product families (Eilmus 2010)

The ideal product program provides as much external variety as needed, requiring as little effort as possible. It is assumed that this ideal compromise is found when:

1. All differentiation drivers and standardisation drivers are balanced according to the aims of their role in the specific corporate context (Figure 4).
2. Carry-over of modules within and across product families is enabled according to the carry-over potential in the specific corporate context (Figure 5).

The role of commonality

In the previous subsection commonality was referred to as the reuse of components. This understanding is based on definitions in Thonemann (2000), Robertson (1998) and Fellini (2006). In order to understand how commonality can relate to the conflicting drivers towards differentiation and standardisation (Figure 4) a broader definition as needed. Based on definitions of Jiao (2000), Dellanoi (2006) and Andreasen (2004), a comprehensive definition of commonality was derived (Eilmus 2012): *Product commonality is the relative property of being designed in a way that the variety of product variants to each other leads to possibly low complexity in a specific company. This may be achieved by the reuse of components and modules, solutions, product structures or interfaces. Common interfaces to the process systems enable process commonality.* This definition shows that commonality is a property that is able to solve the conflict between the opposing drivers for differentiation and standardisation. For example, a variant module can be designed in a way that the interfaces to specific production facilities are the same for each module variant. By doing this, the production driving for standardisation is satisfied as well as the product management driving for differentiation. The aim of product development is to understand the conflicting needs and to find design solutions that suit this conflict in a way that all interests are accounted for in a sufficient way.

The role of modularity

Modularity is a gradual property that can be specified by the attributes commonality, combinability, function binding, interface standardisation and decoupling. Even these attributes are gradual and can take different shapes (Salvador 2007), (Krause 2011). They are not independent from each other; they enable each other. The attributes combinability, function binding, interface standardisation and decoupling serve as prerequisites that enable the commonality of modules. This is why a modular product structure is an important enabler of commonality and many companies strive for modularity in order to reduce variety.

According to the definition above, a module is a group of components for which these attributes were purposefully developed to reach certain corporate aims, such as carry-over, outsourcing, and parallelisation of processes (Krause 2011). A component is understood as the context-specific smallest level of element that is reconsidered for modularisation. This means that when developing a modular structure for an industrial truck, a module may incorporate several sub-assemblies, for example, the drive shaft, pump or control device. Modularising a smaller product or only parts of a bigger product, for example, a control device, components to be clustered to modules may be a single connector, relay or the display screen. However, components are the elements products are decomposed into in order to reintegrate them into modules. As the focus of this contribution is the carry-over of whole building blocks and not only single parts the carry-over of modules is considered. Modularization efforts within product families are a prerequisite enhance commonality in a product program and can be supported by other method units of the integrated PKT-approach (Krause 2011).

4 ACTIONS TOWARDS A COMMONAL PRODUCT PROGRAM

Actions in different fields are necessary to strive towards the ideal common product program. At a strategic level, a holistic strategy for the whole product program needs to be defined. For both resulting tasks of development of product families and product program oriented modules, the development teams need to be supported in understanding the degree of variety that best suits the corporate aims and how to choose a specific technical concept to generate this variety of modules.

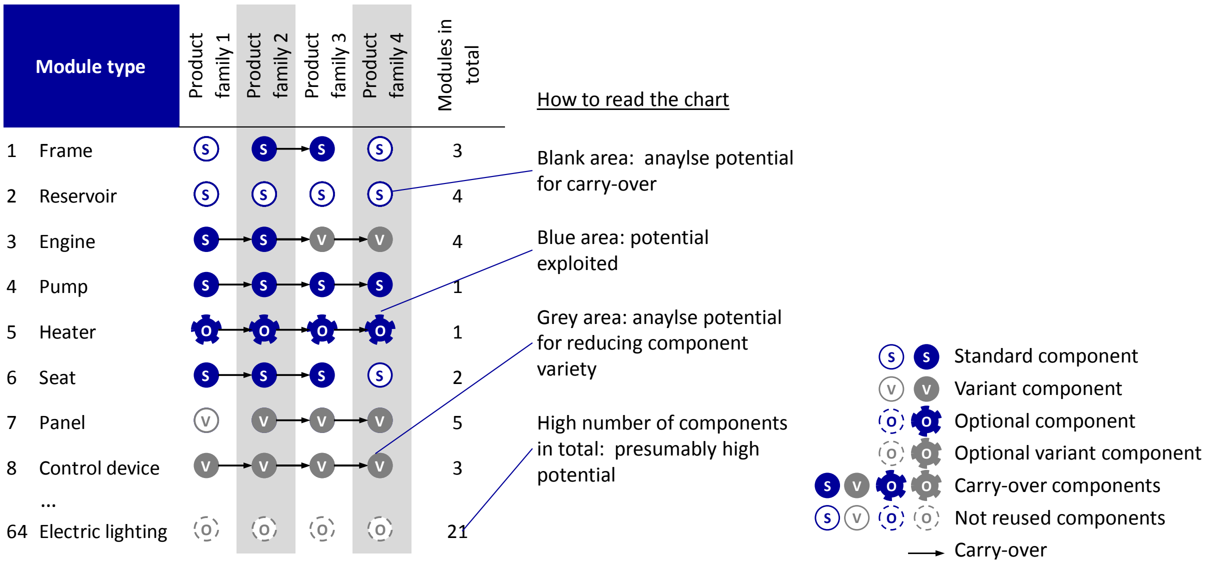


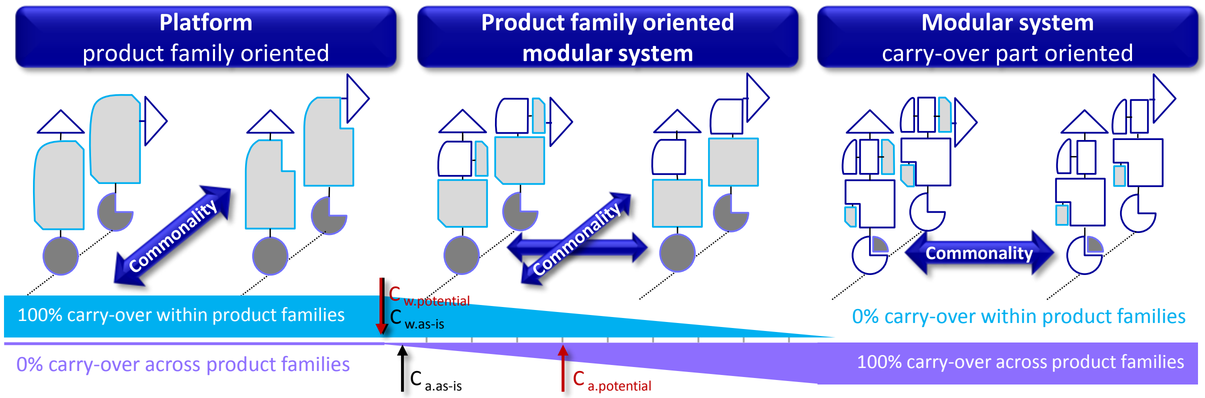
Figure 6. Carry-over Chart to analyse module variety and carry-over

Defining a product structure strategy

By defining a product structure strategy, the corporate approach to exploiting carry-over potential within and across product families is described. This approach should follow the potential for carry-over given by the product type and its variant properties. To understand this potential, an analysis of the carry-over potential of a product program or a smaller section of a product program should be carried out. The Carry-over Assignment Plan by Jonas (2012) is a method for conducting this analysis.

The results of this analysis can be inserted into the chart of product structure strategies (Figure 5). An example of this, using industrial trucks, is given in Figure 7. For this analysis, a representative section of 16 product variants from two product families was chosen. These product variants were decomposed into 64 module types. These module types represent general modules that may be used in several variants. For instance, a module type might be a drive shaft which exists in different variants of shape and performance specifications. For each module type, an as-is-analysis was conducted of how the module variants are carried over to several product variants within the product family or across. For this analysis the Carry-over Chart, as presented in Figure 6, was used. Carry-over within product families was well established: 54 modules were reused in some form within the product family. Across product families only two modules were reused. Analysis of module attributes showed a further carry-over potential in six more modules (60 in total) within product families and 22 modules (24 in total) across product families. In Figure 7 the arrow $C_{w.as\text{-}is}$ marks the actual carry-over within product families. As there are 54 modules with any carry-over, and all of them are reused within product families, $C_{w.as\text{-}is}$ is 100%. In the same manner, $C_{a.as\text{-}is}$ is calculated with 4%. This shows that the existing product program is very close to a pure product family oriented product structure. Analysing the potential for this product program, a significantly higher potential for carry-over across product families was identified. While $C_{w.potential}$ remained at 100% (60 modules with carry-over potential showed potential for carry-over within product families), $C_{a.potential}$ is 40%, as 24 of the 60 modules with carry-over potential show carry-over potential across product families.

This potential shows that a future product structure strategy should enable the exploitation of carry-over potential across product families. The as-is situation (Figure 5, $C_{w.as\text{-}is}$; $C_{a.as\text{-}is}$) reflects the actual strategy of product family oriented development: while all carry-over modules are reused within product families, only a few of them are reused across. For the future, a product family-oriented platform is not recommended as this product structure is not flexible enough for integrating modules that are designed for reuse in the whole product program. To exploit the carry-over potential across product families, a modular system is recommended that is focussed on the high commonalities within product families but better enables the integration of product program oriented modules.



			as is	potential
M	Number of module types		64	64
M_{co}	Thereof with any carry-over		54	60
M_w	Thereof carry-over within product families		54	60
M_a	Thereof carry-over across product families		2	24
C_w	Carry-over within product families	$C_w = \frac{M_w}{M_{co}}$	100%	100%
C_a	Carry-over across product families	$C_a = \frac{M_a}{M_{co}}$	4%	40%

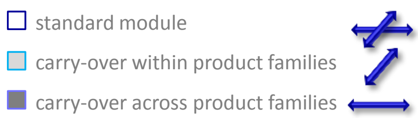


Figure 7. Product structure strategy in a program of industrial trucks

Understanding the suitable degree of variety

In most corporate situations neither strict standardisation nor totally free differentiation are profitable solutions, as described above in Figure 5. It is essential that developing teams gain a deep understanding of the conflicting requirements with which an optimal solution would comply. To do this, a preparative workshop with the team is recommended that collects all the drivers towards standardisation and differentiation that influence the development of the product family or the variant module the team needs to develop. An case study on industrial trucks (which is simplified for publication) showed that applying the integrated PKT-approach on wiring harness variants leads to different possible solutions. All these solutions (Figure 8) comply with the conflicting needs for standardisation and differentiation in different ways. Modularising the wiring harness (Figure 8, left) enables carry-over of the wiring harness but results in an additional assembly step, enhancing component commonality but not process commonality. Modularisation of the electrical components supplied by the wiring harness (Figure 8, middle) enables a standardised wiring harness and shifting of variant assembly processes into the pre-assembly of a variant electrical components module. Oversizing of individual wiring harness variants (Figure 8, right) leads to part number reduction and lot size effects but may also include an additional step in assembly for fixing the additional connector. These three solutions have different advantages and disadvantages. They all enhance commonality in different ways and effect commonality of components, solutions, structures, interfaces and processes to different extends. Which solution is most profitable depends on the specific corporate context, the process strategy, the chain of distribution, etcetera. A way to estimate which solution or which combination of solutions may be most suitable is to refer to the context specific chart of conflicting needs for differentiation and standardisation (Figure 4): How do the solutions comply with the individual requirements of the stakeholders? Which solution would satisfy all differentiation drivers and standardisation drivers to the greatest extend? A more in-depth approach for evaluating the solution is presented in the following subsection.

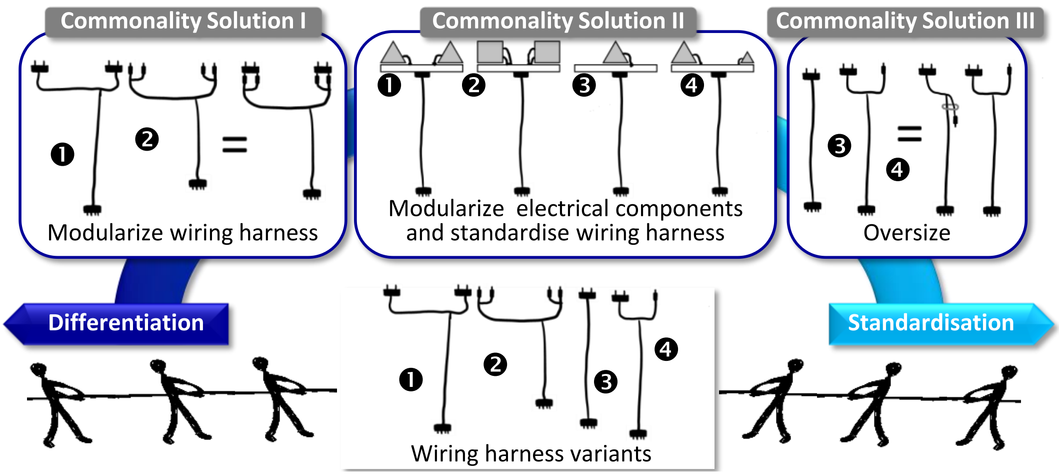


Figure 8. Technical solutions for solving the conflict by enhancing commonality in different ways, using the example of wiring harness variants

Evaluating alternative solutions for generating variety

In many companies an initial cost appraisal is needed in decision making processes for technical solutions. Calculation manufacturing costs is not sufficient to represent how profitable a modular concept is. Effects of reduced complexity, such as handling, storage and administration of reduced part numbers, cannot be included in this way. Although a detailed complexity cost calculation, as proposed by Lindschou Hansen (2012), enables an estimation of these costs, it is not suitable as it requires too-detailed data for the concept phase. Figure 9 shows schematically the example of a cost estimation of wiring harness variants (confidential data has been changed). Different solutions, like the Commonality Solutions I, II and III in Figure 8, were applied to the product variants and combined to create Concepts A, B and C. The manufacturing costs were estimated for each concept and added to the potential complexity cost for each component code number needed to build all variants. Today, the company installing the wiring harnesses evaluates concepts by calculating manufacturing cost only. In this case, Concept C would be chosen as this has the smallest total cost when assuming the complexity cost is 0€. In this graph this changes when the average annual complexity cost per code number is 2000€ or higher. So the question of complexity cost estimation can be reduced to the question of whether the average annual complexity cost per code number for the described code numbers is estimated as more or less than 2000€. In many cases, this simple estimation of higher or lower can be much more easily carried out by controlling and management than a detailed complexity cost analysis in concept phase. Applying this diagram to the case of wiring harnesses, a new understanding of the cost effects in developing concepts for variants generation could be gained by the developing team. However, it is a very rough first estimation and a first step in an approach that will be detailed through further research at the PKT institute.

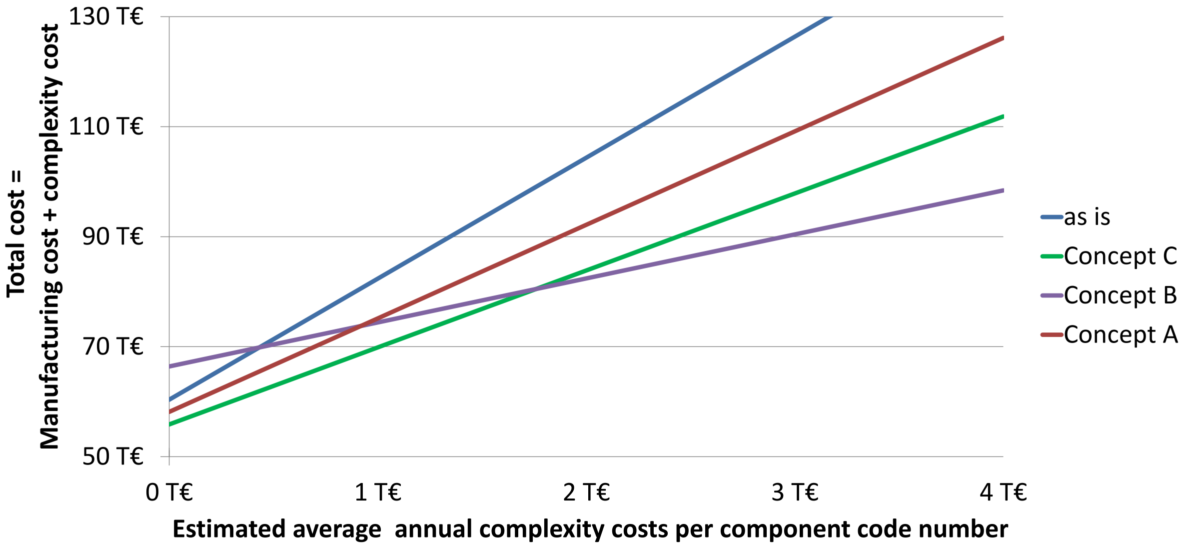


Figure 9. Influence of complexity cost on concept profitability

5 SUMMARY AND PROSPECTS

For the development of variety-optimised product programs, the aim is to exploit commonality within and across product families in a way that best suits corporate needs. A possible approach is outlined in this paper. By starting from understanding the conflicting needs for differentiation and standardisation, understanding of the corporate situation is gained. Analysing the carry-over potential for carry-over within and across product families, a corporate product structure strategy can be derived. To develop modules that are broadly reused in the product program, the developing teams need to develop concepts for the generation of variant products and modules that best suit the conflicting needs for differentiation and standardisation. A first step towards a comparison of complexity cost effects in these concepts is presented. The approach needs to be detailed further. A procedure with the tools to support teams in developing product families and product program oriented modules needs to be derived from the presented ideas. They are based upon tools for product family development, as presented in the framework of the integrated PKT-approach. Methodical support for increasing commonality is currently under development, even for modules where the potential for carry-over is low due to requirements for differentiation.

ACKNOWLEDGEMENTS

Thanks to Jungheinrich AG for provision of case studies and project funding.

REFERENCES

Andreasen, M.M., Mortensen N. H. and Harlou, U. (2004) ‘Multi Product Development: New models and concepts’, *Design for X: 15. Symposium*, 2004, Neukirchen, 2004, Erlangen, Lehrstuhl für Konstruktionstechnik, Univ. Erlangen-Nürnberg, pp. 75–85.

Blankenburg, D. (1997) ‘Procedures of product program development’, *World class design by world class methods - 11th International Conference on Engineering Design*, 1997 Tampere, pp. 111-114

Brosch, M. et al. (2012) ‘Design for Value Chain - An Integration of Value Chain Requirements into the Product Development Process’ *Norddesign 2012*, Aalborg, August 2012, Aalborg, Aalborg University.

Dellanoi, R. (2006), ‘*Kommunalitäten bei der Entwicklung variantenreicher Produktfamilien’*, St. Gallen, 2006

Du, X., Jiao, J. and Tseng, M.M. (2001), ‘Architecture of Product Family: Fundamentals and Methodology’, *Concurrent Engineering*, Vol. 9 No. 4, pp. 309–325.

Eilmus, S., Beckmann, G. and Krause, D. (2010), ‘Modulare Produktstrukturen methodisch in Unternehmen umsetzen - Entwicklung von Standardumfängen und Integration von Erfahrungswissen.’, *Design for X: Beiträge zum 22. DfX-Symposium*, Tutzing, October 2011, Hamburg, TuTech Verl, pp. 99–118.
Eilmus, S., Gebhardt, N., Rettberg, R. and Krause, D., ‘Evaluating a methodical approach for developing modular product families in industrial case studies’, *DESIGN 2012, the 12th International Design Conference*, Dubrovnik, May 2012, Zagreb, Faculty of Mechanical Engineering and Naval Architecture, University of Zagreb pp. 959–968.
Eilmus, S. and Krause, D. (2012), ‘An Approach for reducing Variety across Product Families’, *Norddesign 2012*, Aalborg, August 2012, Aalborg, Aalborg University.
Fellini, R., Kokkolaras, M. and Papalambros, P.Y. (2006), ‘Commonality Decisions in Product Family Design’, in Simpson, T.W., Siddique, Z. and Jiao, J. (Eds.), *Product platform and product family design: Methods and applications*, New York ; London, Springer, pp. 157–185.
Jensen, T. and Hildre, H.P. (2006), ‘The Corporate Platform - a Model to Create a Product Program’, in Magnús Þór Jónsson and Rúnar Unnþórsson (Eds.), *NordDesign 2006 Conference*, Rejkjavik, Iceland, August 2006, pp. 191–200.
Jiao, J. and Tseng, M. (2000), ‘Fundamentals of product family architecture’, *Integrated Manufacturing Systems*, Vol. 11 No. 7, pp. 469–483.
Jonas, H., Gebhardt, N. and Krause, D. (2012), ‘Towards a strategic development of Modular Product Programs’, *DESIGN 2012, the 12th International Design Conference*, Dubrovnik, May 2012, Zagreb, Faculty of Mechanical Engineering and Naval Architecture, University of Zagreb pp. 959–968.
Krause, D. and Eilmus, S. (2011), ‘Methodical Support for the Development of Modular Product Families’, in Birkhofer, H. (Ed.), *The future of design methodology*, London; New York, Springer, pp. 35–45.
Krause, D., Eilmus, S. and Jonas, H. (2013), ‘Developing Modular Product Families with Perspectives for the Product Program’, *Smart Product Engineering - 23rd CIRP Design Conference*, Bochum, March 2013, Berlin, Springer, pp. 543-552.
Krause, D. and Ripperda, S. (2013), A Reflection of methodical Approaches supporting the Development of modular Product Families, *ICED 2013*, Seoul, August 2013.
Lindemann, U., Reichwald, R. and Zäh, M.F. (2006), *Individualisierte Produkte - Komplexität beherrschen in Entwicklung und Produktion*, Berlin Heidelberg, Springer-Verlag.
Lindschou Hansen, C., Mortensen, N.H., Hvam, L. and Harlou, U. (2012), ‘Calculation of Complexity Costs – An Approach for Rationalizing a Product Program’, *Norddesign 2012*, Aalborg, August 2012, Aalborg, Aalborg University.
Meyer, M.H. and Lehnerd, A.P. (1997), *The power of product platforms: Building value and cost leadership*, New York, Free Press.
Mortensen, N.H., Hansen, C.L., Hvam, L. and Andreasen, M.M., ‘Proactive Modeling of Market, Product and Production Architectures’, *DESIGN 2012, the 12th International Design Conference*, Dubrovnik, May 2012, Zagreb, Faculty of Mechanical Engineering and Naval Architecture, University of Zagreb pp. 959–968.
Pahl, G., Beitz, W., Wallace, K., Feldhusen, J., Grote, K.-H. and Blessing, L.T.M. (2007), *Engineering design: A systematic approach*, 3rd ed. , London, Springer.
Reitan, J., Stormo, S. and Aasland, K. (2002), ‘Redesigning Product Programs Focusing On Variety, Complexity and Commonality’, *7th International Design Conference, Design 2002*, Dubrovnik, May 14-17, 2002, Zagreb, Faculty of Mechanical Engineering and Naval Architecture, University of Zagreb; Design Society, pp. 235–239.
Robertson, D. and Ulrich, K. (1998), ‘Planning for product platforms’, *SLOAN MANAGEMENT REVIEW*, Vol. 39 No. 4, pp. 19–31.
Salvador F (2007), ‘Toward a Product System Modularity Construct: Literature Review and Reconceptualization’, *Trans Eng Manag*, No. 54, pp. 219–240.
Schuh, G. (2005), *Produktkomplexität managen*, 2nd ed., München, Hanser.
Thonemann, U.W. and Brandeau, M.L. (2000) ‘Optimal Commonality in Component Design’, *Operations Research*, Vol 48, pp. 1-19.
Zhila P., Wang, G.G. (2011), ‘Recent Advancements in Product Family Design and Platform-Based Product Development: A Literature Review’, *IDETC/CIE 201*, Washington, DC, August 28-31, 2011, ASME.