21ST INTERNATIONAL CONFERENCE ON ENGINEERING DESIGN, ICED17
21-25 AUGUST 2017, THE UNIVERSITY OF BRITISH COLUMBIA, VANCOUVER, CANADA

GOOD PRODUCT LINE ARCHITECTURE DESIGN PRINCIPLES

Mortensen, Niels Henrik; Løkkegaard, Martin
Technical University of Denmark, Denmark

Abstract

Based on existing research concerning product architectures Du et al. (2001), Ericsson and Erixon (1999), Levandowski et al.(2014), Bruun et al. (2015) and studies of more than 200 product architecture projects across a variety of industries, this paper defines ten central principles for design of product line architectures. The first and most important principle is to identify the right number of product architectures to cover a particular market. Having too few or too many architectures can be extremely damaging to profitability and time to market for new products. Despite the importance of having the right set of product architectures, important architecture decisions are often made in individual projects. This is a risky approach, since the total market coverage is not considered, implying that product architectures may overlap or there are areas between product architectures which are not covered. Furthermore, the full benefits of synergies in terms of e.g. increased module/part production volume, increased purchase volume and reduced CAPEX (CAPital EXpenditures) are not harvested.

Keywords: Platform strategies, Product architecture, Product families

Contact:
Prof. Niels Henrik Mortensen
Technical University of Denmark
Department of Mechanical Engineering
Denmark
nhmo@mek.dtu.dk

Please cite this paper as:
Surnames, Initials: *Title of paper*. In: Proceedings of the 21st International Conference on Engineering Design (ICED17), Vol. 3: Product, Services and Systems Design, Vancouver, Canada, 21.-25.08.2017.

1 INTRODUCTION

The role of product architectures (describing how product lines are built up in terms of key modules, key performance steps and interfaces) has during the last ten years received increased attention by academia and the industry, Gonzalez-Zugasti et al. (2000), Hansen (2015), Harlou (1996). It is evident that if architectures are designed right, significant benefits in terms of shorter time to market, cost reductions and improved quality can be achieved, Hultink et al (1997), Jia and Tseng (1999), Krause et al (2013) and Meyer & Lehnerd (1997). Ultimately a certain profit is the goal, but the balance between e.g. cost and time to market is very company specific and product line specific. In the low end, product line architecture is often cost driven and in the high end time to market may be the driving factor.

Based on studies of more than 200 company projects the picture is that many companies have product architectures that have evolved slightly uncontrolled for many years with very little top management attention. The consequences are often that companies have more product architectures than what can be justified from a market point of view. Implications are too high cost and too much complexity in almost every part of the company. Other companies have too few product architectures, which often imply that products are "average", i.e. having too high cost in the low-end segments and not being able to perform in the high-end segments.

In more than 50% of the companies studied, turnover was growing, but the costs were increasing faster than turnover. Among the reasons are: increased number of components, increased stock levels, increased number of new products in the portfolio, etc. In the long run, such a situation is not financially sustainable. One way to address this issue is to introduce new ways of developing products. This paper argues that careful consideration of design principles for product line architectures is of critical importance.

The discussions above can be illustrated by an example from one of the cases studied. A manufacturer of water boiling equipment is producing more than 5.000.000 units per year. In this company, two large development projects were initiated. One project focused on creating a product line for "small" boilers and the other project on creating a product line to cover "large" boilers. Two projects had the task to develop approximately 100 new products (commercial variants) and five new automatic or semi-automatic assembly lines. The results were two individual product architectures and two individual manufacturing architectures. These two product architectures technically covered the market from small to large products, but were problematic from a cost point of view, since costs in the mid-area between small and large boilers were too high. The main reason was that the mid-area boilers were built on a 'scale down' from large boilers. The right principle would have been a scale up from approach from the small products. As the example illustrates, it is important to bring product architecture decisions outside individual projects, since it is not possible to make the right trade off decisions within individual product development projects.

To address such issues, this paper describes ten important design principles to support the design of product line architectures and illustrates these with examples.

Our product development research group at the Technical University of Denmark has during the last seven years been a part of industrial projects in terms of Bachelor, Master, PhD and consultancy projects. The product types that have been studied include power plants (19 studies), white goods (18 studies), consumer electronics (15 studies), toys (17 studies), pumps (12 studies), sensors (13 studies), wind turbines (7 studies), special machinery (30 studies), cell phones (3 studies), control systems (11 studies), boilers (3 studies), filters (4 studies), measurement equipment (11), building equipment (20), and construction equipment (28 studies). 48% of the companies have below 250 employees and the rest ranging from 300 to 80.000 employees. Several of these studies have been published in journals, conferences and in books, Hansen (2015), Harlou (2006), Hansen et al. (2012), Hvam and Mortensen (2007). The data collected in the studies were analysed to derive more general conclusions related to product line architecture design principles. The study of the mentioned case studies of a company's software aspects were not included. This delimitation was introduced to have a clear focus of the research.

2 RELATED WORK

The review of the state-of-the-art literature includes five different groups of literature providing insights into product architectures for product lines Hansen (2015). The five groups identified are: function-based methods, matrix-based models, concurrent engineering, design for manufacturing (DFM), and mathematical models.

Function-based methods: Methods describing the development of modular product architectures often start with the conscious mapping of functional structures into physical modules, Levandowski et al (2014). Functions can be represented in function-based models, e.g. functions-means trees Andreasen (1980), or by schematics of the product including physical elements to a meaningful extent Ulrich (1995).The understanding of product functions can be used in different ways to identify possible modules. To improve the identification of modules and make sure that the modular architecture will serve its objectives Ericsson and Erixon (1990) define a set of module drivers. The module drivers can support the reasoning behind the module identification by elaborating the justification of the modules' existence, e.g. 'planned product changes' module, 'process' module, 'different specification' module, 'technology evolution' module etc. The module drivers are a part of a comprehensive framework called modular function deployment (MFD), which is an analogue to the quality function deployment (QFD) method that provides support for the linking of relationship between the module drivers and technical solutions.

Matrix-based models: Another approach to identify modules is the application of design structure matrices (DSM). This approach takes its point of departure in the decomposition of a product into parts and/or subsystems while identifying the relations (and possible future interfaces) among these Stone et al. (2000). By applying different algorithms and clustering techniques, it is possible to encapsulate functional 'chunks' that have the potential of becoming physical modules, due to their functional interrelations. DSM techniques are the subject of many research initiatives and serve as the basis for an array of derived methodologies. An example of this is the multi domain matrix, Lindemann et al (2009). Alternatively, other design tools focus more on the specific task of examining different functional flows with the aim of identifying modules Otto et al (1998), Stone et al. (2000). These methods are heuristically based. Other more general methods focus on the identification of common features in the existing product program in order to point out the basis of the product architecture. By formulating the design task as a quantitative problem, which can be subject to optimization, this method is balancing inputs from requirements and product variants design with data models of performance and costs. Through iterations, the optimal product variants are designed and evaluated using quantitative performance metrics.

Concurrent engineering: The areas associated with concurrent engineering include research in the concurrent development of product and production architectures, with phrasings such as 'methods supporting the development of product platforms'. In this context, Olesen (1992) introduced a three dimensional methodology superimposing the traditional domains of concurrent engineering, by suggesting the linking of technology, architecture and focus relations in the process, product and supply chain domains. Krause et al (2013) proposed an important step of operationalization of this 3D concurrent engineering approach (3D-CE) by developing a multi-dimensional framework that enables comprehensive assessment of alternative product architectures.

The concept of architecture for product family (APF) is introduced as a conceptual structure, proposing logics for the generation of product families Du et al (2001). The Generic Product Structure (GPS) is then proposed as the platform for tailoring products to individual customer needs. Ko and Kuo (2010) presented another systematic method for concurrent development of product families, involving combination of QFD-based methods with quantified DSM-techniques and morphology analysis to visualize concepts.

Design-for-manufacture (DFM): Olesen (1992) proposed a framework for the concurrent development of manufacturing supported by the Theory of Dispositions. This was done by proposing a set of models aligning the product design and the product life system phase of manufacturing to create a fit. However, the case with DFA and DFM methodologies is that the main focus is single product development. On this basis, Herrmann et al. (2004) argue that an extension of the DFM tools to comprise multi-product development will hold the key to achievement of competitiveness.

Mathematical models: Some researchers have undergone the task of developing methods based on mathematical models. Some methods are based on measures of modularity, which act as subjects of optimization using different techniques Jiao and Tseng (1999). Others seek to integrate product platform, manufacturing process and supply chain decisions through the application of mathematical models, thus extending the concept of the Generic Bills of Materials (GBOM) by quantifying relations between decisions from the different domains.

Conclusion: The above state of the art literature all point in the direction of reducing complexity in products by stabilizing interfaces and reducing the critical interactions between key modules and systems. The concept of functional independence is of importance and is strongly supported by the function models, matrix and mathematical models. What are not explicitly explained in the literature are the more “down to earth” design principles. It is the intension that the design principles defined by this paper should be utilized in the early phases of product architecture projects outside individual projects. Matrix methods and mathematical models are well suited for the later more detailed clarification of product architectures. In other words "do the right things" before "do the right way".

3 PRODUCT LINE ARCHITECTURE DESIGN PRINCIPLES

Based on analysis of the case studies carried out, this paper proposes the 10 following product line architecture design principles, which subsequently are further explained.

1. Determine the right number of product architectures which can be justified from a market point of view.
2. Isolate low volume selling features and options from the core product architecture.
3. Decompose the product architectures in to key module areas based on stable and non-stable key properties.
4. Identify key interfaces that shall be stable over time.
5. Identify right product architecture detail level – ranging from e.g. flow diagram to physical components.
6. Design product architectures to be upwards scalable from low performance to high performance – never from high performance to low performance.
7. Design each key module area to have balanced performance steps aligned between properties in the market and cost in production, supply and delivery. There are discrete performance steps and continuous performance steps.
8. Ensure that product architectures are stable from a production volume point of view.
9. Establish clear link between product architecture and production/supply/delivery architecture, e.g. late customer order decoupling points.
10. Be explicitly prepared for next product launches, e.g. by establishing roadmaps on module level.

3.1 Determine the right number of product architectures which can be justified from a market point of view.

Having the right number of architectures is crucial for all companies, see Figure 1. In this work, product architectures have the following three characteristics: (1) shared core interfaces, (2) core modules/systems exist in balanced performance steps, and (3) architecture(s) prepared for a number of future development projects.

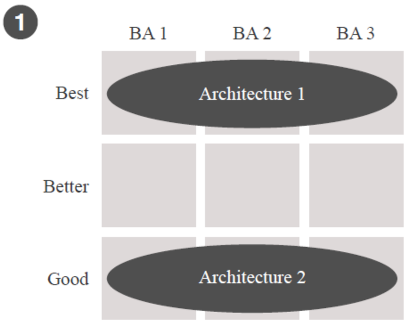

Figure 1. Two product architectures serving “best” and "good" market (BA: business area).

Example from the case studies: In a company manufacturing automotive components four architectures could be justified from a market point of view but eight product architectures existed. This means that this company’s product architectures are way too complex and that there are significant cost optimization possibilities in terms of lower purchasing cost due to volume increase of shared components. Too many solutions are covering the same need/function in the product lines and are therefore non-value adding R&D effort is carried out. This means that when the architectures are reduced from eight to four, it should be possible to reduce time to market significantly and increase number of new products with the same R&D organization. As a vice president stated, “We are the biggest company in our industry – but we do not have the biggest barging power toward our suppliers because of too many product architectures”. "Furthermore automatization cannot be increased further".

3.2 Isolate low volume selling features and options from the core product architecture.

The main idea of product architectures is to harvest synergies in a variety of ways. It is therefore important that product architectures are spot on, cost and performance wise for products with high sales volumes and “trade-offs” can be accepted for lower sales volumes, see Figure 2.
Example from case studies: In a company manufacturing advanced motors with integrated controls, it was decided to integrate the electronics for control of the motor and the display electronics of motor performance due to lower material cost for one complete printed circuit board. Only five percent of the motors are delivered with a display panel. This means that 95% of the motors have cost that are not necessary. Therefore, the right approach would have been to separate electronics into two modules; one module for control of the motor and one module for display of the motor performance.

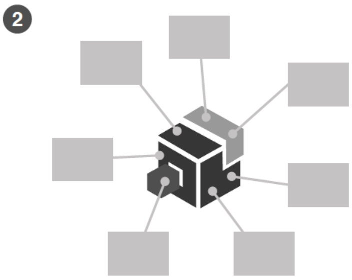

Figure 2. Features (grey boxes) with low sales volumes shall be separated from the core architecture

3.3 Decompose the product architectures in to key module areas based on stable and non-stable key properties.

One very important aspect of product architecture design is to enable explicit preparedness for future product launches Meyer and Lehnerd (1997). In more than two third of the studies cost rationalization has been the main focus. This is all right if the products are stable during the coming years, but can be a disaster if new properties and features have to be introduced, see Figure 3. In some companies significant rationalization has been achieved, but it has not been possible to phase out old products, because the new products are offering the same performance and features as the old products. This means that such companies will have even higher complexity and higher costs when the new product architecture in introduced. As a vice president explained it, “we do not want to be prepared for the previous war but the next war”
Example from case studies: A manufacturer of electric door locks has an architecture where changes of colour (a non-stable property that will be changed due to request from customers) are influencing the moveable mechanics. This means that something relative simple as changing colour will lead to significant utilization of R&D resources and new tool investments. There should have been a clear distinction between the visible and non-visible part of the door lock.

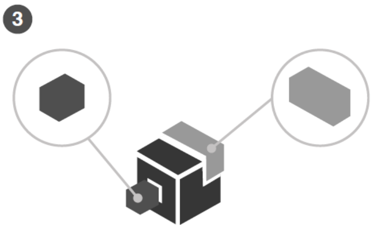

Figure 3. Separate modules according to stable and non-stable properties.

3.4 Identify key interfaces that shall be stable over time

Interfaces are a key concept in all practice and theory on product architectures Erissson and Erixon (1999), Guo et al (2007) and Lindemann et al. (2009). It is clear that interfaces preferably should be stable over time, but in practice this is rarely the case, Mortensen et al. (2016). Among the reasons for this is that decisions in product development in most cases are based on variable cost (direct material and direct labour in manufacturing), see Figure 4. As a senior designer explained, “If decisions are based on variable costs, the answer will always be a new part or new module, it can in most cases be improved – but from a total cost perspective this is not necessary the right answer.” In some companies senior engineers that have been in the same job for many years ensure that this is happening. One company introduced the rule that key interfaces can only be changed if the request is accepted in the board of management.

Example from case studies: An automotive manufacturer has for fifteen years had a stable interface between the gearbox and chassis. This means that when a new gearbox family is developed, it can be applied across all products within the product line. The alternative would have been to execute R&D projects for building the new gearbox into each and every product. The consequence would have been longer time to market and higher R&D effort.

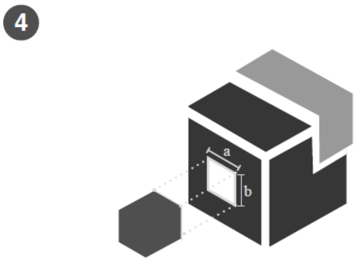

Figure 4. Stable key interfaces over time

3.5 Identify the right product architecture detail level – ranging from e.g. flow diagram to physical components

In literature there is no accepted classification of product architectures. The needs across companies and product lines are very different and therefore the content should most likely be different. An automotive company manufacturing millions of cars and a wind farm company delivering a wind farm every third year will have different architecture needs, implying that the ways synergies should be utilized in such companies differ. One aspect is the level of details. The automotive company would need a detailed architecture on part and module level, while the wind farm company would need a more high level “system” architecture, see Figure 5.

Example from case studies: An Oil & Gas company has implemented product architectures based on modular flow diagrams. This ensures that functionality and key components are the same in each project, while it also promotes flexibility in terms of changing the physical structure for each customer project. The benefits are reduction of engineering hours and increased quality. Such a company would most likely not benefit much from detailed product architecture on the physical part level, because each customer project is highly different and sales volumes are relatively low.

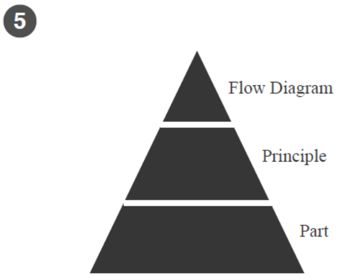

Figure 5. Different product architecture levels

3.6 Design product architectures to be upwards scalable from low performance to high performance – never from high performance to low performance.

There seems to be a tendency that companies start with the most difficult application areas and products. This will however most likely be damaging the profitability of products with lower performance. In the companies studied there were no examples in which a scale-down architecture for physical products was successful, see Figure 6.

Example from case studies: A manufacturer of equipment for analysis of a certain liquid has two product lines: a large one for laboratories where the number of samples are high and a small one for doctor clinics where the number of samples are low. The company decided to develop the architecture for the large product first (laboratory) and then later aimed to down grade it for the small products (doctors clinics). The small products failed to meet the cost target, since it was not possible remove functionality and key components from the large product architecture. As a senior vice president explained, “it is not possible to strip a Rolls Royce and then get a Polo”

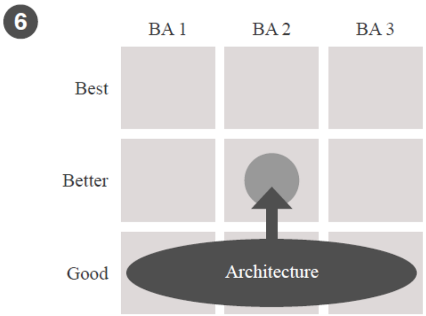

Figure 6. Product architecture shall be scalable upwards

3.7 Design each key module area to have balanced performance steps aligned between properties in the market and cost in production, supply and delivery. There are discrete performance steps and continuous performance steps

In the literature Huang et al (2005), Krause et al (2013), discussions about product architectures containing discrete modules can be found. In many cases, this is a relevant issue, but there are also cases where integration and continuous performance are the right solutions, see Figure 7.

Example from case studies: A wind turbine manufacturer concluded that for wind turbines the different module areas need to allow both discrete and continuous performance steps. An area where continuous performance steps are relevant is the steel tower. Due to the cost of a steel each tower, it shall be optimized according to the load conditions in each position. An example of discrete performance steps is a control system that can be built with modules having discrete performance steps depending on e.g. the power rating and temperature requirements.

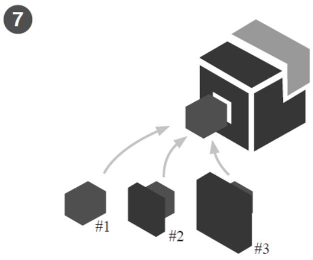

Figure 7. Three different modules with different performance steps

3.8 Ensure that product architectures are stable from a production volume point of view

It is often very difficult to forecast production volume. One way to address this is to build in volume flexibility in a manner allowing increased production volumes to be handled efficiently, see Figure 8.

As a vice president stated, “Product architectures that are designed for automation are also good for manual assembly”.

Example from case studies: A toy manufacturer has been working on product architectures of plastic injection moulds in such a way that e.g. exhaust and cooling functions are established in well-defined modules with well-defined performance steps. The elements that determine the shape of parts are designed individually in each project. In this company it is difficult to make precise forecast on which products will sell in which volumes. Now the company is much more flexible to handle variation in production volume. All the modules, such as the exhaust and cooling modules can now be shared across products. This implies a much faster reaction to changes in production volumes is possible.

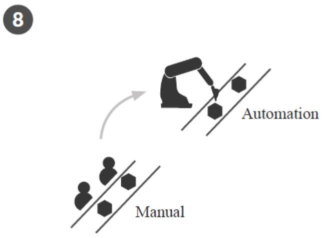

Figure 8. Production volume scalability of a product architecture

3.9 Establish clear link between product architecture and production/supply/delivery architecture, e.g. late customer order decoupling points

Traditional companies are organized according to responsibilities for sales, manufacturing and delivery/supply. The horizontal responsibility definitions across the above areas are therefore often relatively weak, see Figure 9. Significant benefits can however only be obtained if the product architecture work is considered end to end. As a CEO explained, “in our company the work on product architectures has been an exercise in moving cost from one area to another”

Example from case studies: A pressure transmitter manufacturer has worked intensively in automation across many factories. After detailed investigation it was concluded that if the existing product architectures were to be automatized, it would lead to different automation setup in each factory. As a result the number of product architectures were harmonized and reduced with approximately 50%. The benefits were, that the similar factory line architecture could be shared across factories. The consequences are lower investment and increased flexibility, i.e. more products can be produced in several factories.

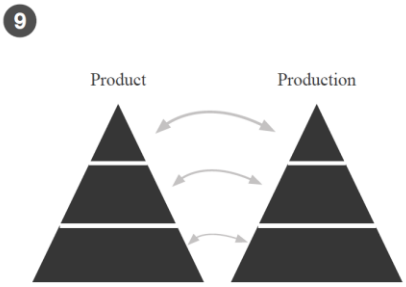

Figure 9. Clear link between product and production architecture

3.10 Be explicitly prepared for next product launches, e.g. by establishing roadmaps on module level.

In most companies, road mapping on product level are introduced, but seldom road maps exist on lover levels such as systems and modules. Working with lower level roads maps is one of the leavers for successful implementation of product architectures, see Figure 10.

Example from case studies: A pump manufacturer has a launch pattern for new products that involved a first launch serving Europe, followed by a US launch, and finally a launch of OEM (Original Equipment Manufacturer) products. A module road map has been introduced in such a way that region (e.g. frequency, UL approvals) and customer specific (colour, size) requirements are isolated in the product architecture. In parallel with development of the European product line explicit product architectures for the US and the OEM are taken care of. This has led to a significant faster time to market for the US and OEM variants.

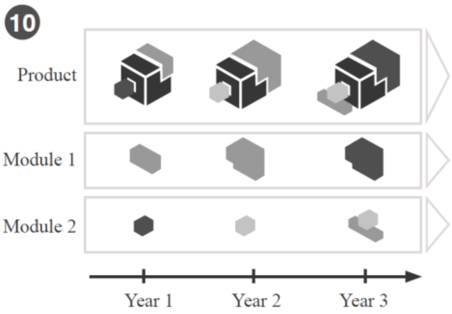

Figure 10. Road maps on both product and module level.

4 CONCLUSIONS

Despite that all the companies studied have had some kind of product planning and road mapping in place, all of them are non-compliant with the first principle, i.e. having the right number of product architectures. The consequences can be quite severe in terms of increased cost and increased time to market. Situations observed in the cases studied include:

- Product architectures are taken into market areas where they technical can cover but have very bad cost/performance. This will lead to low margins.
- Product architectures are too many and overlapping. This means that there are multiple ways of serving the same customers or segments. This again will lead to increased complexity in engineering, production, quality, purchase etc.
- Product architectures are not covering “the middle” areas”. In some of the observed companies this has been important due to unexpected high sales volumes in the middle area between two product architectures.

The first analysis of the data presented in this paper indicates that companies generally have too many product architectures. This has a significant negative impact on financial and innovative performance. The above challenges cannot be handled in individual projects but has to be clarified upfront. Product architecture responsibility is in principle being taking care of by e.g. a program management department, but on the concrete level that is not sufficiently concrete to guide the individual product projects. In one of the companies the cost of “wrong and too many product architectures were quantified”. The cost base was roughly 5% too high measured on direct material and direct labour in production. On top of this there were complexity costs, such are long change-over times on assembly lines, low purchase volumes, high investments in tooling, etc.

Our future research will therefore focus on studies of procedures to identify of the optimal number of product architectures. Such procedures need to include calculations of complexity costs and end-to-end thinking to avoid sub-optimization.

REFERENCES

Andreasen, M.M.(1980), *Machine Design Methods based on a Systematic Approach - contribution to a design theory*, Ph.d. thesis. (Danish) Department of Machine Design, Lund University, Sweden.

Andreasen, M.M. Olesen, J. (1990), “The Concept of Dispositions”, *Journal of Engineering Design*. 1, 17-36.

Du, X., Jiao, J., Tseng, M.M. (2001), “Architecture of Product Family: Fundamentals and Methodology”, *Concurrent Engineering*. 9, 309-325.

Ericsson, A, Erixon, G. (Eds.) (1999), *Controlling Design Variants - Modular Product Platforms*, Society of Manufacturing Engineers, Dearborn, Michigan.

Gonzalez-Zugasti, J., Otto, K.N., Baker, J.D. (2000), “A Method for Architecting Product Platforms”, *Research in Engineering Design*. 12, 61-72.

Guo, F., Gershenson, J.K. (2007), “Discovering Relationships Between Modularity and Cost”, *Journal of Intelligent Manufacturing*. 18, 143-157.

Hansen, C.L. (2015), "*On the identification of architectures for product programs in a complexity cost perspective*", PhD thesis, DTU Mechanical Engineering, 2015.

Hansen, C.L., Mortensen, N.H., Hvam, L. (2012), "On the Market Aspect of Product Program Design: Towards a Definition of an Architecture of the Market", *12th International Design Conference - Design 2012*.

Harlou, U. (2006), *Developing Product Families Based on Architectures - Contribution to a Theory of Product Families*, Department of Mechanical Engineering, Technical University of Denmark, Kgs. Lyngby.

Herrmann, J.W., Cooper, J., Gupta, S.K., Hayes, C.C., Ishii, K., Kazmer, D., Sandborn, P.A., Wood, W.H., (2004), *New Directions in Design for Manufacturing*, IDETC/CIE, 1-9.

Huang, G.Q., Zhang, X.Y., Liang, L. (2005), "Towards Integrated Optimal Configuration of Platform Products, Manufacturing Processes, and Supply Chains", *Journal of Operations Management*. 23, 267-290 146

Hultink, E.J., Griffin, A., Hart, S., Robben, H.S. (1997), "Industrial new product launch strategies and product development performance". *J. Prod. Innovation Management*. 14, 243-257.

Hvam, L., Mortensen, N.H. and Riis, J. (2008), *Product customization*, Springer.

Jiao, J., Simpson, T.W., Siddique, Z. (2007), "Product Family Design and Platform-based Product Development: A State-of-the-Art Review", *Journal of Intelligent Manufacturing*. 18, 5-29.

Jiao, J., Tseng, M.M. (1999), "A Methodology of Developing Product Family Architecture for Mass Customization", *Journal of Intelligent Manufacturing*. 10, 3-20.

Ko, Y., Kuo, P. (2010), "Modelling Concurrent Design Method for Product Variety", *Concurrent Engineering*. 18, 207-217.

Krause, D., Eilmus, S., Jonas, H. (2013), "Developing Modular Product Families with Perspectives for the Product Program", in: *Smart Product Engineering*. Springer, pp. 543-552.

Levandowski, C; Michaelis, M.T.; Johannesson, H (2014), "Set-based development using an integrated product and manufacturing system platform", *Journal of Concurrent Engineering: Research and Applications*, Volume 22, Issue 3, pp. 40-58.

Lindemann, U, Maurer, M., Braun, T. (2009), *Structural Complexity Management: An Approach for the Field of Product Design,* Springer, Berlin.

Mortensen, N.H., Hansen, C.L., Løkkegaard, M. Lars Hvam, L. (2016), "Assessing the Cost Saving Potential of shared Product Architectures", *j. Concurrent Engineering Research and Applications*, Volume 24, Issue 2, 1 June 2016, Pages 153-163.

Meyer, M H, Lehnerd, A.P. (1997), *The Power of Product Platforms - Building Value and Cost Leadership*, The Free Press, New York.

Olesen, J.(1992), *Concurrent Development in Manufacturing - based on dispositional mechanisms*, Ph.d. thesis, Department of Engineering Design, Technical University of Denmark, Kgs. Lyngby.

Otto, K.N., Wood, K.L. (1998), "Product Evolution: A Reverse Engineering and Redesign Methodology", *Research in Engineering Design*, 10, 226-243.

Stone, R.B., Wood, K.L., Crawford, R.H., (2000), "Using Quantitative Functional Models to Develop Product Architectures", *Design Studies*. 21, 239-260.

Ulrich, K. (1995), "The Role of Product Architecture in the Manufacturing Firm", *Research Policy*. 24, 419-440.