SUPPORTING DESIGN PLATFORMS BY IDENTIFYING FLEXIBLE MODULES

Raudberget, Dag S. (1); Levandowski, Christoffer (2); André, Samuel (1); Isaksson, Ola (2); Elgh, Fredrik (1); Müller, Jakob (2); Johansson, Joel (1); Stolt, Roland (1)
1: Jönköping University, Sweden; 2: Chalmers University of Technology, Sweden

Abstract

One way for firms to stay competitive is to adapt a platform approach. In product platforms, modules are used as exchangeable design blocks to create a variety in product performance. This is a proven way to get advantages of scale in production by reusing physical parts and investments in manufacturing. To ensure exchangeability between modules, interfaces between modules must be well defined. Hence, from this point of view, there is no such thing as flexible modules. In this research, flexibility refers to the idea of identifying strategic portions of the platform where flexibility is needed and to create the modular division in a way that the assigned modules are de-coupled in theses areas. The presented approach shows how the Design platform concept can be extended by the introduction of flexible modules. These support the Design Platforms by allowing areas of strategic importance to be more flexible and thereby enable room for uncertainties such as fluctuating requirements and future technical development.

Keywords: Product architecture, Set-based Concurrent Engineering, Systems Engineering (SE), Functional modelling, Modularisation

Contact:
Dr. Dag S. Raudberget
Jönköping University
School of Engineering
Sweden
dag.raudberget@ju.se

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

Development is the driving force for most business today; only by a constant pursuit for new products, services or business models a company can prosper in the long term. The rate of innovation is accelerating and companies struggle to keep up with the competition. One way to stay competitive is to adopt a platform approach. In product platforms, modules are used as exchangeable design blocks to create a variety of products. This is a proven way to get advantages of scale in production by reusing physical parts and their corresponding investments in tooling and manufacturing set- up (Meyer and Lehnerd 1997).

In the design phase, modules can be reused in different settings, saving valuable development costs and reducing time- to market. To ensure exchangeability between modules, the key is to keep performance steps well defined and to control and maintain the interfaces between modules. Hence, from this point of view, there is no such thing as flexible modules. In this research, flexibility refers to the concept of identifying strategic portions of the platform where flexibility is needed and to create the modular division in a way that the assigned modules are de-coupled in these areas.

Modular platforms have several drawbacks. They are not useful for supporting product development of highly physically integrated products or for designs with a high degree of customer adaption such as for Engineer- To- Order business models. Moreover, a product developing company has other assets besides its physical components and modules. Robertson and Ulrich (1998) state that “A platform is a collection of assets that are shared by a set of products. These assets can be divided into: components, processes, knowledge, and people & resources. Only if taken together, do these four elements constitute a platform”.

Two approaches that follow this view on platforms are the Configurable Component framework (Johannesson and Claesson 2005) and the Design Platform concept (Elgh *et al.* 2016). The Configurable Component (CC) framework contains abstract functional models of platform architectures describing the Design Rationale, interfaces and constraints. As opposed to a traditional product platform, the Design Platform (DP) also contains various reusable resources such as the Design Rationale, design rules, methods, processes and prepared CAD-models.

The DP stems from a need among suppliers of customized products that are not able to benefit from an artefact-based product platform. This as a means to realize a platform consisting of development resources as defined by Robertson and Ulrich (1998). The Engineer- To- Order business models are different from the end consumer market where the customer is introduced in the sales and distribution phase of the product life cycle. The product development in the Engineer- To- Order businesses is often characterised by projects running for several years in cooperation with the customer. This introduces several sources for changes and fluctuations in requirements due to the evolution of interfacing systems. The DP acknowledges that changes are inevitable due to uncertainties and therefore contain flexible and adaptable solutions, which allows for customisation when the requirements change throughout the development projects. Figure 1 visualizes the conceptual DP introduced in the product lifecycle. It is continuously fed with new technologies as they emerge and which are described using Set-based approaches. These can be picked as the need for adaptable solutions approaches due to changing requirements. The DP is updated with new knowledge only when a certain level of maturity and formalization is reached.

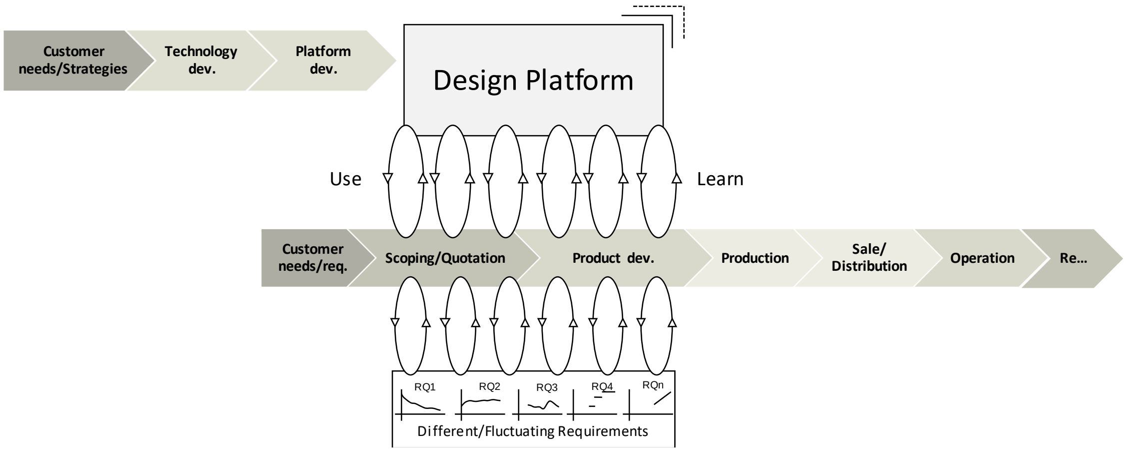

Figure 1: The Design Platform positioned in the product life cycle. Adapted from (Elgh et al. 2017)

The different characters of these platform descriptions make them useful in different contexts. Design Platforms has a wider scope and variety in platform elements than CC as well as the specific goal to support suppliers of customized systems (Elgh *et al.* 2017), hence the need for flexibility in modularisation. However, Design Platforms has no specific support for early phases of development, which is evident by examining figure 1. It would therefore be beneficial to include the CCs as additional platform elements.

This paper presents the first steps towards increased flexibility of DPs by adding a new way for module division based on the methods from the CC framework. The approach aims at accommodating technological evolution or fluctuating requirements that can impact the platform by analysing the character of modular division in early phases of development. The purpose is to extend the DP concept with a suitable support for early phases of platform development, thereby incorporating the flexibility required by suppliers of customized systems. The objective is to create a methodology to incorporate flexibility in strategic modules of the platform to meet changing technology and customer requirements.

2 RELATED RESEARCH

The research community has recognized the benefits of modules and platform-based approaches and its increasing importance for manufacturing companies in the last decades. Platforms enable efficient utilization of the resources in a company through economies of scale. Product platforms also enable rapid generation of new product variants by reusing components and interfaces variants (Meyer and Lehnerd 1997, Halman *et al.* 2003). Several distinct platforms types can be identified (Zhang 2015) and this work builds upon two development oriented platform approaches: The Design Platform concept (Elgh *et al.* 2016) and the Configurable Component framework (Johannesson and Claesson 2005).

One common characteristic for these platform types is that they are supported by Set-based Concurrent Engineering (SBCE). It is characterized by considering sets of design alternatives rather than a specific design. Sobek *et al.* (1999) formulate three principles that define a set-based design process: *map the design space, integrate by intersection and establish feasibility before commitment.* These principles may seem simple, but has proven themselves valuable in industrial case studies (Raudberget 2010) and addresses issues with regular product development. It does so by considering a wide range of alternative design solutions that are systematically narrowed down by eliminating undesirable solutions.

2.1 Modularisation

Modularisation enables the creation of product variants by combining a set of interchangeable modules. Modules are designed to cover performance steps so that the resulting product variants can cover a pre–defined range of customer requirements. Different properties and functionality can be achieved by changing one module for another (Gonzalez-Zugasti and Otto 2000). The interchangeability is the key to modularisation and this is achieved by fixing key interfaces and geometry, which also reduces complexity in the design since there are fewer interactions that need to be managed.

Modularization has several benefits, such as facilitating reuse of modules between product generations, provided that the interfaces are kept fixed. It also enable variation in functionality through changing modules for others (Gu and Sosale 1999). Modularization is also important for concurrent engineering by diving a design into independent units with clear interfaces between modules (Gershenson *et al.* 2003). By defining interfaces, development can proceed independently of the work in other modules by concurrent teams, across organisational functions and by suppliers.

Dividing a product platform into modules can also give negative effects. One evident drawback is that a modular design is not optimised across the platform due to the fact that it is a compromise between commonality and customisation. Another drawback is the lack of flexibility both in the aspects of having rigid interfaces constraining the design possibilities and the optimisation of integrated products.

2.2 Perspectives on development oriented Product Platforms

Platform elements are the ingredients used to compose the platform. This research focuses on development-oriented platforms including other elements than just the physical architecture of the product. In this context, a platform is seen as a set of shared assets which is in line with the view of Robertson and Ulrich (1998) who define a platform as *"the collected assets shared by a family of products".* Besides the common view of platforms being based upon basic architectures that comprise subsystems and modules with interfaces between them (Meyer and Lehnerd 1997), other assets are included in the platform definition.

2.3 Configurable Component framework

The Configurable Components (CC) framework is a structured functional approach based on systems theory principles (Hitchins 2003) and design theory (Andreasen 1991), (Hubka 1982) forming a coherent system model. It contains information about the system solution, system variants and also its underlying requirements and motivations, i.e. its design rationale (Johannesson and Claesson 2005). A CC model is a formalized specification of a technical system based on the Enhanced Function-Means tree (EF-M) (Schachinger and Johannesson 2000) referred to as a "development platform" by Levandowski (2014). The EF-M model is decomposed in subordinate systems by a systematic design approach as seen in Figure 2.

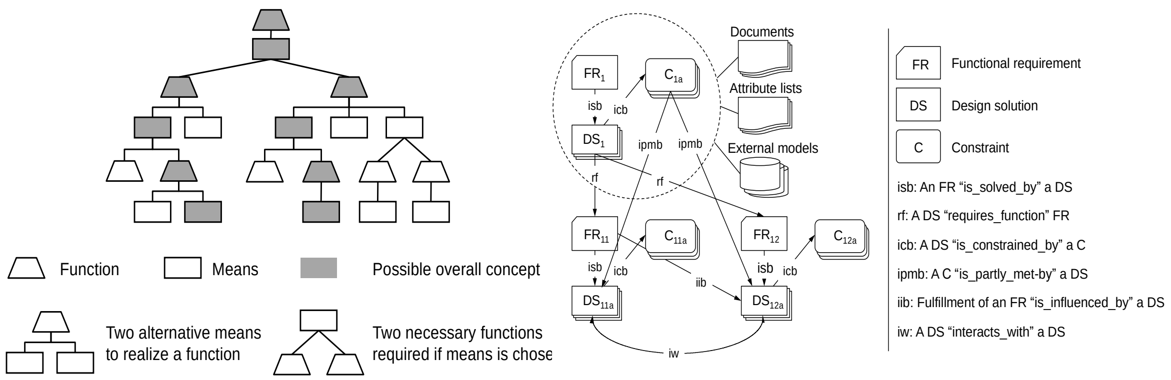

Figure 2: Left: An example of an F-M model representing several alternative overall concepts (adopted from (Svendsen and Hansen 1993)). Right: The Enhanced F-M model and its functional interactions (adopted from (Johannesson and Claesson 2005))

The architecture of a product platform is created through the choices of possible design solutions of the EF-M tree, forming different Architectural Options. EF-M models include features that give a richer description of the proposed system. The definition of its interactions and constraints as can be seen in Figure 2. The *iib* and *iw* relations represents functional interactions that are used to model and analyse platforms at early stages of development. One example of this is given by (Raudberget *et al.* 2015) where different platform alternatives are evaluated without the need for geometrical embodiment in CAD etc.

2.4 The Design platform

The Design Platform (DP) concept is a loosely arranged construct that, among several assets, could include CCs as one platform element. The DP was developed from the needs of engineering-to-order companies and was refined and extended through the Chase project (Elgh *et al.* 2016). Besides physical components and modules, a DP also includes re-use of assets that often are ill-structured and acknowledges their respective contributions to a firm's success. This incorporates other carriers of information and knowledge such as design guidelines, computer programs, formalised design-, quotation- and order processes.

A company´s DP is composed of different objects related to process, synthesis resources, product constructs, assessments resources, solutions and projects. A conceptual image is shown in Figure 3. The figure shows how these different resource types can be linked to a product. The product consists of its generic structure, derived variants, cardinality and attributes. The items of the structure (assemblies and parts) of and its variants in turn points to the different resources used for its realization as well as existing solutions. The solutions are linked to the projects where they were developed. A process resource can be in the form of tasks and execution orders of activities required or intended to support some part of the design process. A knowledge resource can be collected according to the A3- methodology given in (Raudberget and Bjursell 2014). The process and product concept can further be linked to resources to be used in the process steps. These can be synthesis resources that aid the designing of a product by defined guidelines or other methods and models. These can also be assessment resources that supports evaluation of product variants and which embody mathematical models representing behavior and other properties. Geometry resources are commonly parametric computer aided models that can act as a base line for new designs.

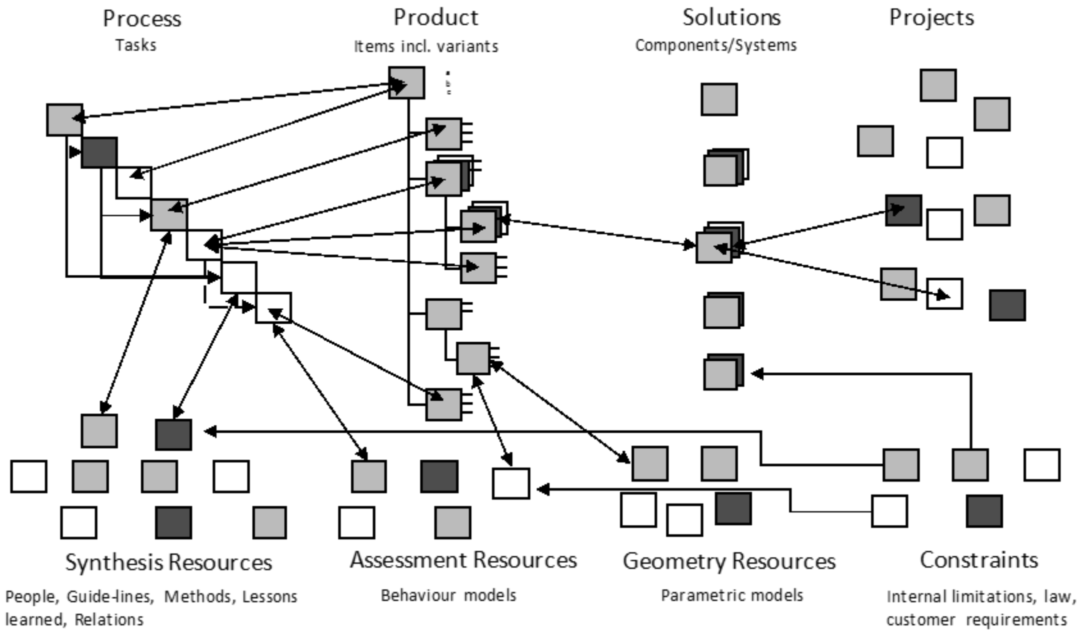

Figure 3. A model of the generic Design Platform with the different assets related to Process, Synthesis Resources, Product Constructs, Assessments Resources, Solutions and Projects. After (Elgh et al. 2016)

The DP concept acknowledges that changes in requirements will occur during product development and prepares for these changes by using a Set-based process. By also adding support in the early phases of development through the CC framework, the ability to accommodate changes could be further enhanced.

3 DEFINITION OF FLEXIBLE MODULES

Flexible modules are physical constructs that deliver specific functionality and performance. The first step for introducing flexible modules it to identify parts of the platform model where an increased amount of design flexibility is desirable. This could be an area where changes in technology or customer needs are likely or planned. The methodology identifies several promising candidates for flexible modules and evaluates these based on complexity and functional couplings.

The suggested process builds on earlier work that provides parts of the methodology. Raudberget *et al.* (2015) describe the system modelling and its alternatives in the form of Architectural Options as well as the analysis of dependencies for each architecture by transforming these into Design Structure Matrixes (DSM) (Steward 1981). The clustering into different modules and the modularity assessment method is presented in Levandowski (2016) and the concept of flexible modules and the identification of these is the scope for the present paper.

3.1 Creating different Architectural Options

The process begins with the approach given in Raudberget *et al.* (2015). The starting point is the requirements defining the platform. Initially, a functional breakdown is created by EF-M modelling where the system is subsequently broken down into smaller parts. Design solutions are elaborated and collected following a Set-based approach, aiming at producing a set of possible solutions to each Functional Requirement.

Following the functional modelling, different platform architectures are created on the basis of system compatibility through the choices of alternative design solutions that divide the platform into fundamentally different directions. These alternatives are the Architectural Options that drives different interactions and functional couplings in the system. To characterise the Architectural Option, each design solution needs to be specified to a sufficient degree. This refers to elaborating on the interactions in the model by identifying significant functional connections. These interactions are transformed into DSMs for each Architectural Option. The Design Solutions forms the rows and columns of the DSM and the significant functional connections are marked in the cells of the matrix.

3.2 Clustering the modules

The Architectural Options formulated as different DSMs are used as the input for the approach presented in Levandowski (2016). Here, modules are created by clustering the DSMs and analysing their complexity in terms of connectivity. The goal is to generate the most suitable platform architecture. Three metrics are used to assess the modularity: internal connectivity, external connectivity and interface complexity. These metrics are used to assess each Architectural Option.

3.3 Identification of flexible modules

The identification of flexible modules uses on the clustering method presented above. In the generation of Architectural Options, the flexibility comes from the approach of distributing functionality differently over the specific platform instances. Instead of using the cluster to eliminate complex architectures as it was intended, the clusters are used to assess the character of the modules for each different type of architecture. The clustering will render different number of modules, module division and module sizes for each architecture. Consequently, it is possible to identify platform instances that has the required amount of flexibility in desired modules, for further development.

4 ILLUSTRATIVE EXAMPLE

The illustrative example is based on data from an earlier research presented in (Raudberget *et al.* 2015). It is a part of a jet engine and the functional relationships between the ingoing Design Solutions and Functional Requirements are modelled in the *Configurable Component Modeller* software (Edholm *et al.* 2009). The functional knowledge built into the platform model is highly structured and well suited as a part of the DP.

Briefly, The E-FM model is created by functional decomposition and provides a systematic way of arranging functions and solutions to functions in a hierarchic tree structure. The result is a functional model containing several alternative design solutions with corresponding functional connections. From the functional model, several instantiations, Architectural Option, are generated and exported to DSMs, listing all Design Solutions and their significant functional connections.

The DSMs are clustered in order to identify potential modules and one example is given in Figure 4 where the different modules are seen as greyed-out squares. A detailed description of this process is presented in (Levandowski 2016).

	Conical surface	Cone TRS flange	Oil tubing	Integrated generator	Active cooling	Bleed air	TRS LPT flange	Rigid hub and shroud structure	Hub	Hub surface	Sensors	Angular vanes	Vanes in gaspath	Integrated vanes	Shroud sufficiently resistant	Shroud surface	Mounting lugs	Rigid load bearing structure from hub to mounting points
Conical surface	x									1								
Cone TRS flange	1	x							1									
Oil tubing			x															
Integrated generator				x	1													
Active cooling				1	x													
Bleed air						x												
TRS LPT flange							x	1	1									1
Rigid hub and shroud structure							1	x	1	1								1
Hub		1	1	1			1		x	1				1				1
Hub surface	1							1	1	x			1			1		
Sensors											x	1						
Angular vanes											1	x		1				
Vanes in gaspath												1	x	1		1		
Integrated vanes									1			1	1	x	1			1
Shroud sufficiently resistant														1	x	1	1	
Shroud surface										1			1		1	x		
Mounting lugs																	x	1
Rigid load bearing structure from hub to mounting points							1	1						1			1	x

Figure 4. Clustered DSM of one Architectural Option. The modules are visualised as greyed-out squares. *After (Levandowski (2016)*

In Figure 3, the significant functional connections described in section 2.3 between Design Solutions in the DSM are indicated by the number "1" in the matrix. These are used to cluster the Design Solutions into modules and in the presented approach the key for assessing the flexibility of each module. The analysis is performed for each Architectural Option, offering an overview of which platform instances that has the potential to give the required flexibility in the key areas.

The module consisting of *Oil tubing, Integrated generator* and *Active cooling* has no significant functional relations outside the module and thereby a minimum of external connections in its interfaces. Hence, it is a good candidate for the development of a flexible module. This is opposed to the module consisting of *Bleed air*, *TRS LPT flange*, *Rigid hub and shroud structure*, *Hub*, and *Hub surface* having connections to a majority of the Design Solutions. This implies that a design change to this module likely will affect several other modules, hence disqualifying it as a candidate for a flexible module. If a change in requirement specification is anticipated for this module it could lead to changes in adjacent systems. To facilitate the development process, the interfaces for this module therefore should be fixed early in the development process.

4.1 Adapting the distribution and number of modules to enable flexible modules

Following the principles of Set-based design, several Architectural Options are generated and analysed and for the sake of brevity only two alternative architectures are presented in this paper. In Figure 5, an overview of two alternative architectures is seen, together with and their corresponding design solutions and functional connections. Here, the candidates for flexible modules are marked with dotted circles.

	Integration in fairing	Fairing on hub	Conical surface	Sensors	Angular vanes	Oil tubing	Integrated generator	Active cooling	Bleed air	TRS LPT flange	Rigid hub and shroud structure	Hub	Hub surface	Mounting lugs	Rigid load bearing structure	Vanes in gaspath	Rod	Shroud sufficiently resistant	Shroud surface
Integration in fairing	x	1	1																
Fairing on hub	1	x	1													1			
Conical surface	1	1	x										1						
Sensors				x															
Angular vanes				1	x														
Oil tubing						x													
Integrated generator							x	1											
Active cooling							1	x											
Bleed air									x										
TRS LPT flange										x	1	1			1				
Rigid hub and shroud structure										1	x	1	1		1				
Hub		1				1	1			1		x	1		1		1		
Hub surface			1								1	1	x			1			1
Mounting lugs														x	1				
Rigid load bearing structure from hub to mounting points										1	1			1	x				
Vanes in gaspath		1			1											x	1		1
Rod												1				1	x	1	
Shroud sufficiently resistant														1			1	x	1
Shroud surface													1			1		1	x

	Conical surface	Cone TRS flange	Oil tubing	Integrated generator	Active cooling	Bleed air	TRS LPT flange	Rigid hub and shroud structure	Hub	Hub surface	Sensors	Angular vanes	Vanes in gaspath	Integrated vanes	Shroud sufficiently resistant	Shroud surface	Mounting lugs	Rigid load bearing structure from hub
Conical surface	x									1								
Cone TRS flange	1	x							1									
Oil tubing			x															
Integrated generator				x	1													
Active cooling				1	x													
Bleed air																		
TRS LPT flange							x	1	1									1
Rigid hub and shroud structure							1	x	1	1								1
Hub		1	1	1			1		x	1				1				1
Hub surface	1							1	1	x			1			1		
Sensors											x	1						
Angular vanes											1	x		1				
Vanes in gaspath												1	x	1		1		
Integrated vanes									1			1	1	x	1			1
Shroud sufficiently resistant														1	x	1	1	
Shroud surface										1			1		1	x		
Mounting lugs																	x	1
Rigid load bearing structure from hub to mounting points							1	1						1			1	x

Figure 5. The candidates for flexible modules are marked with dotted circles in this schematic view of two different Architectural Options. It illustrates the difference in functional rigidity, size and distribution of modules. Redrawn from (Levandowski (2016)).

It is clear that the two Architectural Options have different distribution of functions within the modules as well as different significant functional connections. This information allows engineers to avoid architectures that are heavily connected in modules where a future replacement of modules may be needed to accommodate uncertain requirements or changed functionality. Modules with few and simple interfaces are preferable from a flexibility point of view and the process can identify architectures with the desired distribution of significant functional connections to enable flexibility in desired places.

5 DISCUSSION

The presented approach shows how creating flexibility in strategic modules in a structured way can support the DP. It does so by adding a new way of module division based on the methods from the CC framework. The methodology can support the DP by identifying areas of strategic importance and allowing these to be less constrained, thereby enabling room for uncertainties such as fluctuating requirements and future technical development. This flexibility can also give benefits during development by identifying and freezing the dependencies between systems in the right order thereby allowing uncertainties where the requirements still are unclear. Moreover, since DP has no specific support for early phases of development, CCs could give benefits as additional platform elements.

One example of when to apply the methodology is the case where a technical solution for one function is likely to be replaced during the market life of the system. Here, the methodology can point to a platform architecture that is less sensitive in this aspect. By analysing the distribution of functions and the number of modules where a shift in technology can be expected, functionality can be suitable distributed over the suggested modules. In this way, the character of each resulting platform architecture will be different. For the architectures, functionality is clustered into modules based on significant functional connections. For each architecture, the characters of the modules are assessed for flexibility and the result is used to identify architectures that supply the desired flexibility in the specific modules.

To summarise the contribution, the paper presents a methodology that can identify and incorporate flexibility in strategic modules of a platform to meet changing technology and customer requirements. The main contribution is a new method to distribute functionality over modules based on functional connections and complexity.

One challenge with the methodology is that the clustering algorithm plays an important role in the way that the platform is divided into modules. This implies that potentially superior solutions may not be analysed and the impact of the clustering method needs to be further investigated. Moreover, the illustrative example comes form an earlier study and more work is needed to evaluate the usability of the methodology in a realistic industrial setting. However, the presented results indicate that the methodology supports the Design Platforms concept by creating flexibility in strategic modules to meet changing technology and customer requirements.

REFERENCES

Andreasen, M. M. (1991), “The theory of domains”, in *Workshop on understanding function and function to form evolution*, Cambridge University, UK.

Edholm, P., Wahl, A., Johannesson, H. and Söderberg, R. (2009), “Knowledge-based configuration of integrated product and process platforms”, in *ASME 2009 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, American Society of Mechanical Engineers, 495-504.

Elgh, F., André, S., Johansson, J. and Stolt, R. (2016), “Design Platform: Setting the Scope and Introducing the Concept”, in *DESIGN 2016, 14th International Design Conference*, The Design Society, 1253-1264.

Elgh, F., André, S., Johansson, J. and Stolt, R. (2017), “A Coherent Model for Management and Use of Mixed Design Assets”, in *24rd ISPE Inc. International Conference on Transdisciplinary Engineering*, Singapore.

Gershenson, J. K., Prasad, G. J. and Zhang, Y. (2003), “Product modularity: Definitions and benefits”, *Journal of Engineering Design*, 14(3), 295-313.

Gonzalez-Zugasti, J. P. and Otto, K. N. (2000), “Modular platform-based product family design”, in *Proceedings of ASME DETC*, Baltimore, Maryland, USA, 10-13 September, Citeseer, Paper No. 14238.

Gu, P. and Sosale, S. (1999), “Product modularization for life cycle engineering”, *Robotics and Computer-Integrated Manufacturing*, 15(5), 387-401.

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), 149-162.

Hitchins, D. K. (2003), *Advanced systems thinking, engineering, and management*, Artech House.

Hubka, V. (1982), *Principles of engineering design*, Elsevier.

Johannesson, H. and Claesson, A. (2005), “Systematic product platform design: A combined function means and parametric modeling approach”, *Journal of Engineering Design*, 16(1), 25-43.

Levandowski, C. (2014), *Platform Lifecycle Support using Set-Based Concurrent Engineering* [Dissertation], unpublished thesis Chalmers University of Technology.

Levandowski, C., Müller, J. och Isaksson, O. (2016), "Modularization in Concept Development Using Functional Modeling.", in *23rd ISPE Inc. International Conference on Transdisciplinary Engineering*, Curitiba.

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

Raudberget, D. (2010), "Practical Applications of Set-Based Concurrent Engineering in Industry", *Journal of Mechanical Engineering- Strojniški vestnik*, 56(11), 685-695

Raudberget, D. and Bjursell, C. (2014), "A3 reports for knowledge codification, transfer and creation in research and development organisations", *International Journal of Product Development*, 19(5-6), 413-431.

Raudberget, D., Levandowski, C., Isaksson, O., Kiopuros, T., Johannesson, H. and Clarkson, J. (2015), "Modelling and Assessing Platform Architectures in Pre-Embodiment Phases through Set-Based Evaluation and Change Propagation", *Journal of Aerospace Operations*, 3(X).

Robertson, D. and Ulrich, K. (1998), "Planning product platforms", *Sloan Management Review*, 39(4), 19-31.

Schachinger, P. and Johannesson, H. L. (2000), "Computer modelling of design specifications", *Journal of Engineering Design*, 11(4), 317 - 329.

Sobek, D. K., Ward, A. C. and Liker, J. K. (1999), "Toyota's principles of set-based concurrent engineering", *Sloan Management Review*, 40(2), 67-83.

Steward, D. V. (1981), "The design structure system: a method for managing the design of complex systems", *IEEE transactions on Engineering Management*, (EM-28).

Svendsen, K.-H. and Hansen, T. C. (1993), "Decomposition of mechanical systems and breakdown of specifications", in *Proceedings of ICED 1993*, The Hague, The Netherlands, 17-19 August.

Zhang, L. L. (2015), "A Literature Review on Multitype Platforming and Framework for Future Research", *International Journal of Production Economics*.

ACKNOWLEDGEMENTS

The authors would like to express their gratitude towards the Swedish Agency for Innovation Systems (VINNOVA) for financing this research.