



## PRODUCT ARCHITECTURE TRANSITION IN AN EVOLVING MULTI-BRAND ORGANISATION

D. Williamsson, U. Sellgren and A. Söderberg

### Abstract

A modular architecture is a strategic means to deliver external variety and internal commonality. A methodology for product modularization that integrates complexity and strategies is proposed and logically verified with an industrial case from the heavy truck business area. The case study indicates that the new methodology is capable of identifying and proposing reasonable module candidates that address product complexity as well as company specific strategies.

*Keywords: product architecture, design structure matrix (DSM), variant management, characteristics and properties*

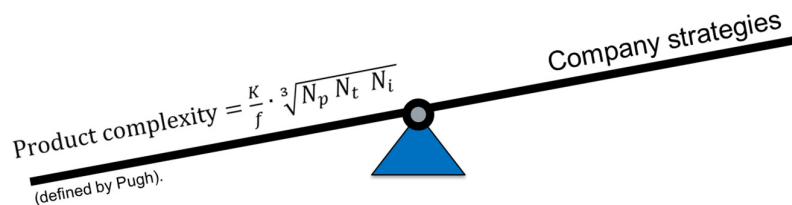
### 1. Introduction

Ulrich (1995) gives a general definition of Product architecture as "the scheme by which the function of a product is allocated to physical components", and more specifically "(1) the arrangement of physical elements; (2) the mapping from functional elements to physical components; (3) the specification of the interfaces among interacting physical components". Product architectures can be categorized as being modular or integral (Höltkä-Otto, 2005). A hybrid architecture is one that is a mix of these both types of architectures.

In an integral architecture, interfaces between components are coupled (Ulrich and Tung, 1991). This implies a complex mapping of functions to components (Höltkä-Otto, 2005). Suh (1990) proposed a metric called Reangularity to determine whether a system is highly coupled (Reangularity close to zero) or uncoupled (Reangularity close to unity).

Products with a modular architecture are configured from predesigned modules. A module, which is a function carrier, has standardized interfaces, is selected for company-specific reasons (Erixon, 1998), which means it supports a company specific strategy. An interface can be thought of as a contract between two functional blocks, i.e. modules, which defines the spatial orientation and/or the exchange of information, matter, or energy (Börjesson, 2014). The main purpose of a modular architecture is to enable external variety to be provided, that is many possible product variants to the customers, and at the same time internal commonality, that is reduction of parts (Blackenfelt, 2001). The term "module variant" refers to a physical incarnation of a module with a specific performance level. A modular architecture can, thus, be defined as the collection of module variants by which all the required end products, i.e. a family of products, can be built (Börjesson, 2014). Stake (2000) explores the many ways modular architecture can support company strategy, and in particular the use of Module Drivers (Erixon, 1998). The Module Drivers (MDs) are generic, but their importance is company specific, e.g. modules can reduce capital needs and bring economies in parts sourcing (Ulrich and Tung, 1991; Baldwin and Clark, 2000). Modules can also support design re-use (Smith and Duffy, 2001), planned product changes, upgrades, and outsourcing/insourcing (Ericsson and Erixon, 1999), etc.

According to Hölttä-Otto (2005), there are three main approaches to modularity, which mainly are complementary; Heuristics, Design Structure Matrix (DSM), and Modular Function Deployment (MFD). Börjesson (2010) compared these three main methods and two extended DSM-based methods, and reported benefits and drawbacks for all five methods. Heuristics is based on an analysis of the pattern of flow of matter, energy, and information between function blocks, see e.g. (Erixon, 1998). The main focus of DSM-based approaches is to minimize technical complexity by clustering the system components in a way that minimize the technical interactions between clusters of components, i.e. complex interactions are grouped within clusters. A very efficient DSM-clustering, algorithm, referred to as IGTA++, was proposed in (Börjesson and Sellgren, 2013). Heuristics and DSM approaches address technical complexity, but not strategic objectives (Blackenfelt, 2001). MFD (Ericsson and Erixon, 1999) is a five-step method for translating customer requirements into a modular architecture, while considering the company specific strategic objectives described using twelve predefined MD that are represented by a Module Indication Matrix (MIM) relating the physical function carriers, i.e. the components, and the twelve MDs. The MDs are the main reasons or purposes to group elements into modules. The "reasons" are related to development, variety, production, procurement, quality, and after sales. Ulrich and Eppinger (1995), Sanchez (1994), Smith and Reinertsen (1995), and Ulrich (1995) list a range of reasons that both complement and overlap the MDs of Erixon. Augmentation of a New Product Development (NPD) process with additional analyses to deal with the architecting task is treated in (Otto et al., 2013), but there is currently no accepted method on how sudden changes in business strategies could or should be efficiently and effectively reflected as changes of an existing modular product architecture. In an attempt to balance the technical complexity represented by a DSM and business strategies represented by a MIM, Stake (2000) presented examples from manual clustering of a DSM and a MIM. Blackenfelt (2001) presented a method on how the module drivers could be condensed into four generic groups, i.e. Carry over, Commonality, Make or by, and Life cycle, and represented the relations between those four groups for each component as a DSM, but performed no further analysis.



**Figure 1. Integrated modularization enables architecture trade-off studies**

A compact and scalable approach to represent product architecture is to use a DSM, thus making it suitable as the main representation, while the MIM is a structured way of assigning and representing strategic aspects as additional architecture information. Williamsson and Sellgren (2016) addressed the challenge to perform trade-offs between technical complexity and company specific business strategies, as visualized in Figure 1, with a new method referred to as Integrated Modularization Methodology (IMM), that integrates company specific module drivers with a component-DSM, followed by a clustering analysis of the strategically adapted DSM. The pros and cons of this proposed matrix-based method for integrated modularization needs to be further worked on. The aim of this paper is therefore to investigate if the IMM could add valuable contributions, compared to a component-DSM clustering and elaborating with the MIM, when re-architecting a technically complex product due to changed business strategies. Or more specifically, is the IMM capable of identifying module candidates according to both product complexity and business strategies?

Hence, this paper presents an industrial case, where a truck manufacturer, further on referred to as Corporate A, with a unique business strategy, based on extensive in-sourcing, in a relatively short time period had to modify parts of its modular architecture to efficiently become also a First-Tier OEM-supplier to another large truck manufacturer, here referred to as Corporate B, with a completely different business and product architecting strategy. The new business relation between the two truck

manufacturers affects the relative importance of some of the generic module drivers for a technically very complex product. The studied case, which is presented in chapter 2, is analysed with the previously proposed IMM, from both technical complexity and module driver perspectives in chapter 3, and discussed in chapter 4. The main conclusions and a path for future research are given in chapters 5 and 6.

## 2. Case study

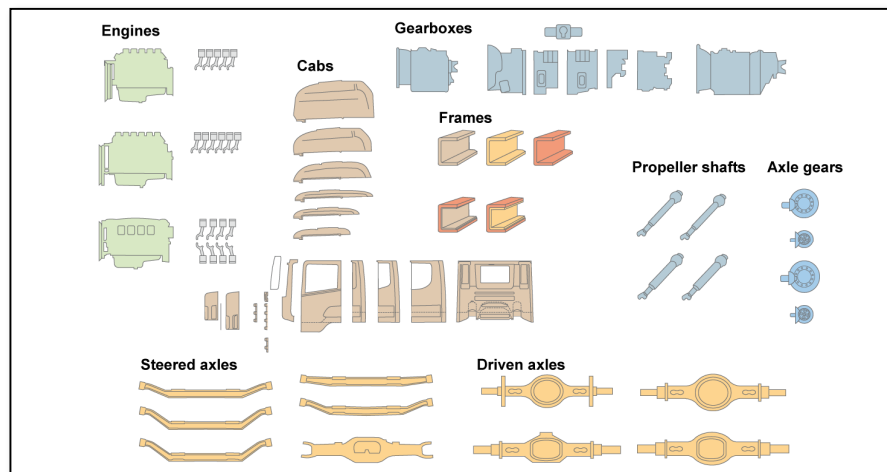
This case study illustrates how a product architecture may be transformed, and the reasons for this transformation as well as possible improvements, when it is integrated into a multi-brand organisation. The case study was conducted at Corporate A and KTH Royal Institute of Technology, in collaboration with Corporate B. The original product architecture of the studied system was initially developed and used as a product module by Corporate A. However, when Corporate A became a part of a larger Business organization, Global Corporate, the Corporate A product architecture had to be changed in various ways in order to fulfil the new requirement to allow for representing different distinct, and competitive, brands, i.e. Corporate A and Corporate B, of Global Corporate.

### 2.1. Corporate A product architecture

Corporate A has a successful history on the global heavy truck market and are many times used as a role model for modularisation. In order to efficiently describe the many product variants, Corporate A represents the modular product as a generic product structure. A generic product structure does not describe a single product variant, but rather the entire product portfolio, which is referred as the Corporate A Modular Toolbox, see (Figure 2).

At Corporate A, the modularisation process starts and ends with the customer. This means that Corporate A modularise their product to fit every customers needs as well as possible.

The core of Corporate A's modularisation principle is carefully balanced performance steps, i.e. module variants, with standardised interfaces that can be combined to satisfy different customer needs, with a limited number of components. Hence, Corporate A strives to maximise the number of product variants (external variety), while keeping the number of technical solutions low (internal commonality). The product architecture at Corporate A can therefore be seen as the result of the specific modularisation process and principles of Corporate A. You should also be noted that Corporate A almost exclusively relies on in-sourcing.



**Figure 2. The Corporate A modular toolbox**

To handle product variation and customisation efficiently, Corporate A strives towards late variant definition, meaning that the configuration of the modules and module variants should take place as late as possible, i.e. preferably after the customer has ordered the product.

Making the variant configuration as late as possible in the development process and production chain, enables a company to benefit from several advantages, such as improved inventory savings and customer service, and lowering the overall costs.

In order to efficiently enable trade-offs between high degree of configuration flexibility and high overall performance, the electrical system and the embedded software are designed to be decentralised, with multiple ECUs (Electronic Control Unit) distributed in the different main vehicle modules.

Almost all embedded software at Corporate A is included in the module variants, independent of product configuration. However, the software still needs to be adapted depending on the specific hardware configuration. The software is therefore parametrised to fit the different physical configurations, and is therefore modularised in "another way" than the hardware.

If the software is linked to the hardware depends on what layer the software belongs to in the software structure. In a lower layer, there is usually a strong connection between hardware and software, e.g. software that control an actuator implies a strong interdependence between software and hardware and consequently there is a strong driver to integrate the software and hardware into a common module. However, the higher software layers, e.g. gear shifting or smart cruise controller, are not tightly connected to any specific hardware, since these software components realise system functions on a higher level and only need real-time computing power, wherever it is available.

As mentioned above, Corporate A manufactures most of its truck modules in-house. This means that outsourcing is generally a weak module driver at Corporate A, compared to many other performance related module drivers. Hence, the architecture does not have to be strictly developed according to some predefined architecture rules, but rather to the solution which works best at the moment for the customer and Corporate A.

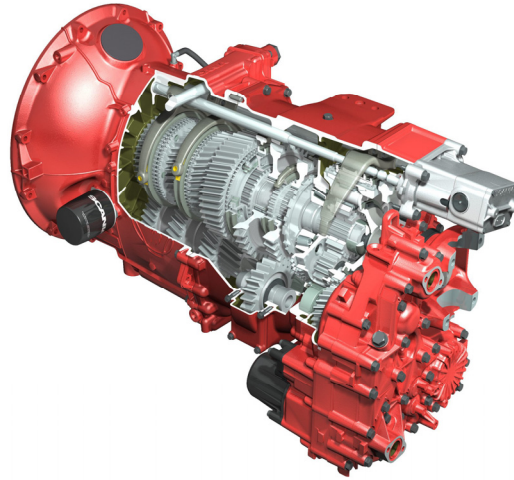
Corporate B is more of a system integrator than Corporate A is. Consequently, Corporate B has outsourced manufacturing, and also development, of many components and modules to several external suppliers. Corporate A is, thus, offering products with a different architecture than Corporate B, which relies on a supplier-robust architecture that also protects corporate specific strategic knowledge and plans. Furthermore, Corporate B has chosen a largely centralized monitoring and function control strategy compared to Corporate A, which has a more decentralized control strategy, i.e. more of the intelligence is embedded in modules.

## **2.2. Integrating a product architecture**

A specific heavy duty gearbox was selected, see Figure 3, and is used here to exemplify how a specific Corporate A truck subsystem with heterogeneous technology has been transformed and evolved, during the collaboration with Corporate B. The reason for selecting this driveline module is because Corporate A develop and manufacture this gearbox in-house and deliver it to Corporate B as a First-Tier supplier. Corporate A is therefore responsible for the product architecture and the complete design of the gearbox module, based on the requirements stated by Corporate B. The gearbox is also a good representation of how Corporate A architects its product. It should be stated that it is only the gearbox variant which is delivered to Corporate B that is treated in the presented case study. Hence, all other variants which are used by Corporate A have their original architecture intact.

A multidisciplinary approach was needed in this investigation, and therefore an analysis of the mechanical and electrical systems, including embedded software was initially performed.

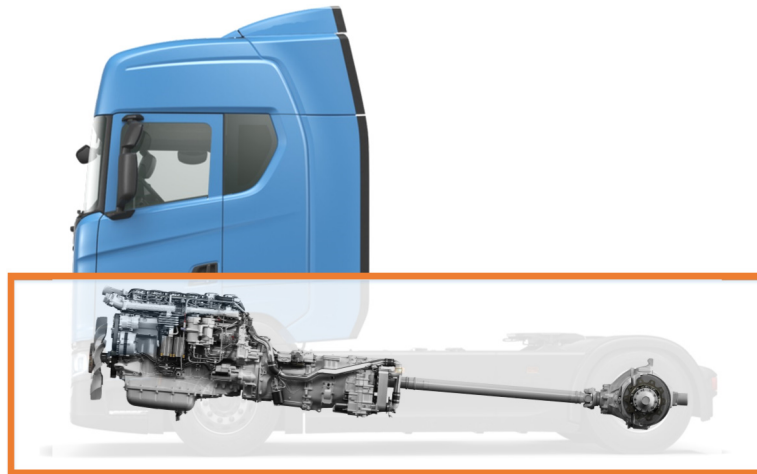
In order for the gearbox from Corporate A to be used in a new brand, several changes had to be made. These changes mainly relates to the integration of the gearbox into a completely new system, with different requirements and interfaces. In this new environment, the gearbox is subjected to a different type of heat (different exhaust pipe routing), vibrations, cooling water/oil flow, air pressure, gear shift control strategy, engine torque etc.



**Figure 3. Illustration of a heavy duty Corporate A gearbox**

Even if the gearbox was treated as a relatively independent module at Corporate A, it was still originally designed and optimised to be a part of a Corporate A powertrain, in a Corporate A truck, see Figure 4. Hence, some gearbox software components were located in the engine ECU in order to gain gear shifting performance benefits.

At Corporate B, gearboxes from multiple suppliers are used, a business model that makes it necessary to treat the gearbox as a highly independent module with some predefined functions. It was therefore not possible to simply use the original gearbox (with the original Corporate A software) in a Corporate B truck, without first performing some redesign, or adaptation.



**Figure 4. Cutaway illustration of a Corporate A truck powertrain**

In addition to the technical aspects, the different brand strategies should also be considered. Examples of a brand strategies may be how many product variants that are desirable, if the variants should be defined during the early or late stage of the product development process (early or late variant definition) and if the modules should be designed to be as uncoupled as possible, or if the overall system performance is more important.

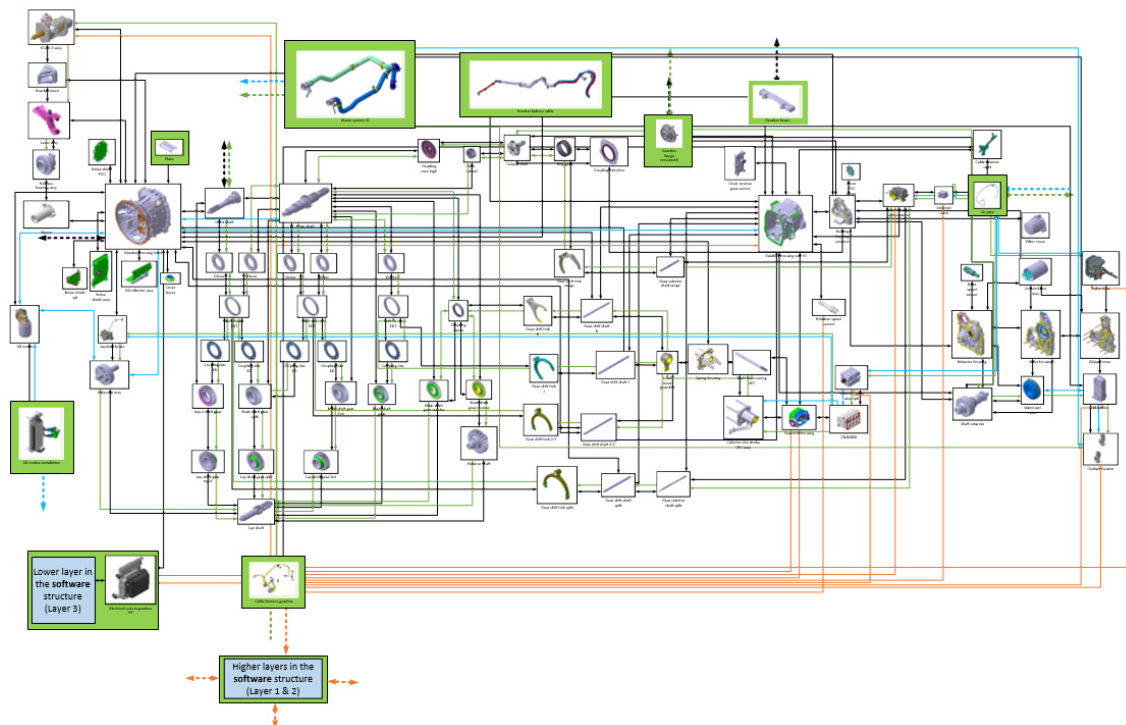
### **2.3. Changed hardware and software components**

To visualise the required changes of the gearbox, its components must first be identified. This was done by studying the physical decomposition represented in the product structure at Corporate A. To limit the

number of components to a reasonable level, all screws, O-rings, gaskets and other small parts, were not considered when decomposing the technical solutions.

How the 94 identified components interact with each other was then represented with a component structure diagram, see Figure 5. The interactions are described with the principal technical function flow and spatial relations, where black indicates a spatial relation, green indicates energy flow, blue indicates material transfer, and orange indicates information flow. The layout of the components in the component structure diagram is not fully representing the actual physical layout of the gearbox, since the goal was to focus on the relations. However, it gives a conceptual understanding of the gearbox design structure. Finally, the eleven components which had to be changed are marked with green in Figure 5. In addition to these components, some unions and brackets also had to be redesigned. It should be stated that these changes were identified and performed by expert engineers at Corporate A and at Corporate B, who has a large amount of product architecting experience and knowledge. As seen in Figure 5, there are many relations and thereby component interfaces which needed to be redesigned. Since an interface is a relation between features of a pair of components, a changed interface means that multiple components must be considered if one component is redesigned, which results in a quite complex redesign phase.

In addition to that, the spatial interface locations and standard are not the same at Corporate A and Corporate B. For example, due to the different vehicle architectures, the interfaces of the electrical system and cooling water pipes are located at different positions in the two brands. In addition to that, Corporate B requires an oil heat exchanger to be located on the side of the gearbox, which is not the case at Corporate A.



**Figure 5. Changed components in the modular architecture targeted for Corporate B**

Furthermore, the routing of cables and pipes in the different brands are not the same. The gearbox at Corporate B therefore needs to have a battery cable attached to the gearbox housing, even though this cable has nothing to do with the gearbox functions and operations.

Finally, the gearbox is used in different ways in the two brands. At Corporate B, the crawler gear is used as a driving gear. However, the crawler gear was originally only designed for starting the truck and was therefore only dimensioned for static loads.

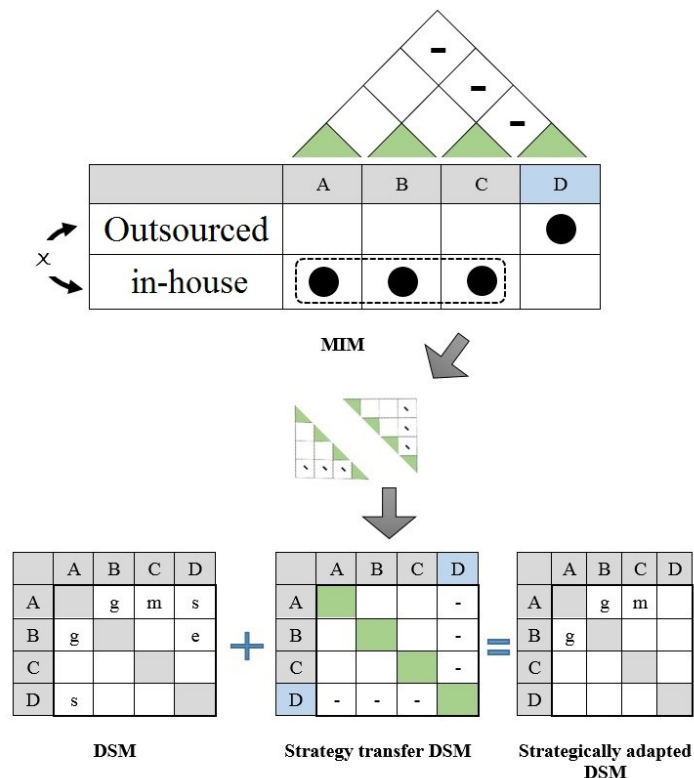
The strategic part of the gearbox software, developed by Corporate B, includes the higher layers in the software structure (Layer 1 & 2), e.g. predictive cruise controls and gear selection strategy etc. The lower level in the software structure (Layer 3) is however still developed for the gearbox module by Corporate A. This type of software is closely linked to the hardware e.g. clutch actuation and shifting actuation. Since Corporate A develop some parts of the software and Corporate B the other parts, the interfaces must be standardised according to Corporate B requirements. Corporate A therefore had to make relatively large changes to their original software design, in order to comply with the new required standardised interface. The software which is developed by Corporate B, is "physically" located in the main centralised ECU, while the Corporate A software is located in the gearbox ECU.

### 3. Analysis

The initial gearbox architecture, which was developed in-house at Corporate A, was transformed based on expert judgement, to an outsource module, to be integrated into the Corporate B system product.

The transformed gearbox architecture was represented and analysed with the DSM and the new IMM clustering methodology proposed in (Williamsson and Sellgren, 2016), see Figure 6, to verify if the IMM clustering methodology could be used to assist the highly complex task to change an architecture, from both reduced technical complexity and changed business strategy points of view.

In the IMM-based analysis, the components which were actually redesigned to enable efficient integration of the gearbox architecture into a Corporate B driveline were treated as outsourced components. The remaining components were treated as developed in-house, according to the Module Driver (MD) definition, shown in Table 1. In order to gain company specific strategic benefits, the components which are outsourced should not be clustered with the components which are developed in-house, according to the MFD methodology.



**Figure 6. The integrated matrix-based systems architecting model IMM**

The company specific strategic module drivers considered here are the twelve MDs of the six groups that are used in the MFD method, as listed in Table 1.

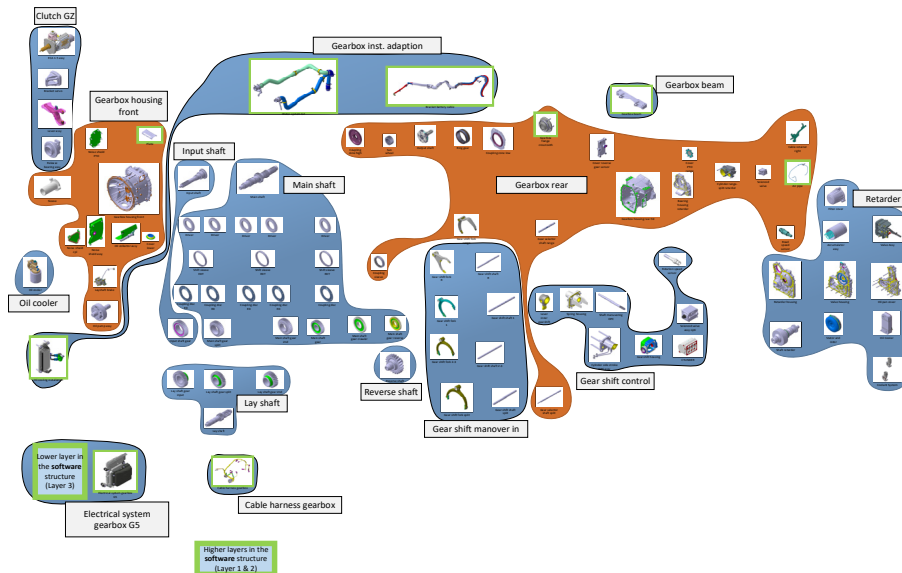
**Table 1. The twelve MFD Module Drivers, from Erixon (1998)**

Module Driver group	MD #	Module Driver description
Development and design	1	Carry over
	2	Technology push
	3	Product planning
Variance	4	Different specification
	5	Styling
Production	6	Common unit
	7	Process/organization
Quality	8	Separate testing
Purchase	9	Black-box engineering
After sales	10	Service/maintenance
	11	Upgrading
	12	Recycling

In order to represent the modular gearbox architecture in a compact format, the component structure diagram seen in Figure 5 was used as a starting point, excluding the technical relations. The modules are visualised as blue and orange shapes (clusters) in Figure 7, 8 and 9.

### 3.1. The transformed architecture

The blue and orange shapes in Figure 7 are the existing modules developed by Corporate A. They were identified by the physical decomposition represented in the Corporate A product structure. In the component structure diagram, the company strategies were also specified in addition to the technical solutions and the modules. An orange shape indicates that the module contains components with conflicting module drivers, i.e. in this case components which should be developed in-house and be outsourced at the same time. As seen in Figure 7, there are multiple strategically conflicting modules, which may result in a less efficient architecture from a strategic viewpoint. For example, the module "Gearbox housing front" contains a plate which should be outsourced, while the remaining components should be developed in-house. The blue shapes indicate that there is no strategic conflict within the module. Also notice that the modules in Figure 7 are highly linked to how the product is manufactured and assembled, i.e. to manufacturing complexity, but not necessarily according to product complexity.



**Figure 7. Schematic illustration of the transformed architecture used at Corporate B**



### 3.2. Clustering analysis

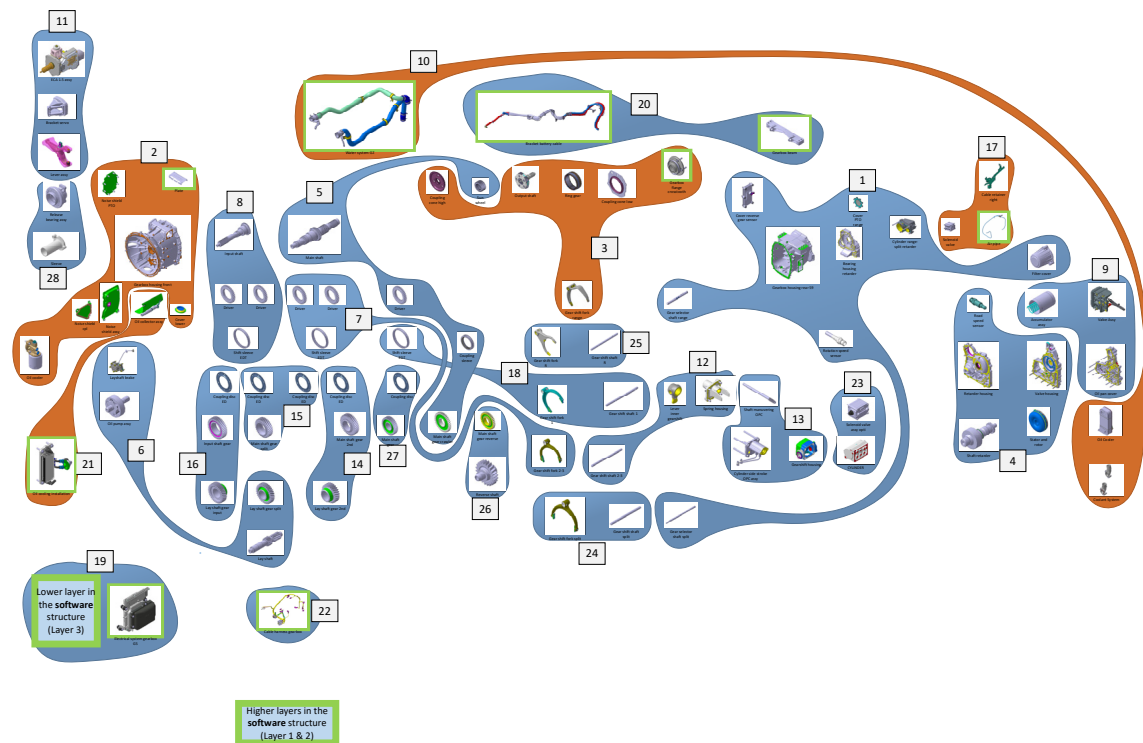
The transformed architecture is now analysed according to how it could have been modularised if using the DSM or the proposed IMM methodology. This is done in order to identify potential architectural improvements, and to logically verify the proposed IMM methodology. This analysis is potentially useful in the future, if new modules should be integrated into different (external) product architectures.

**Table 2. Chosen relation weights**

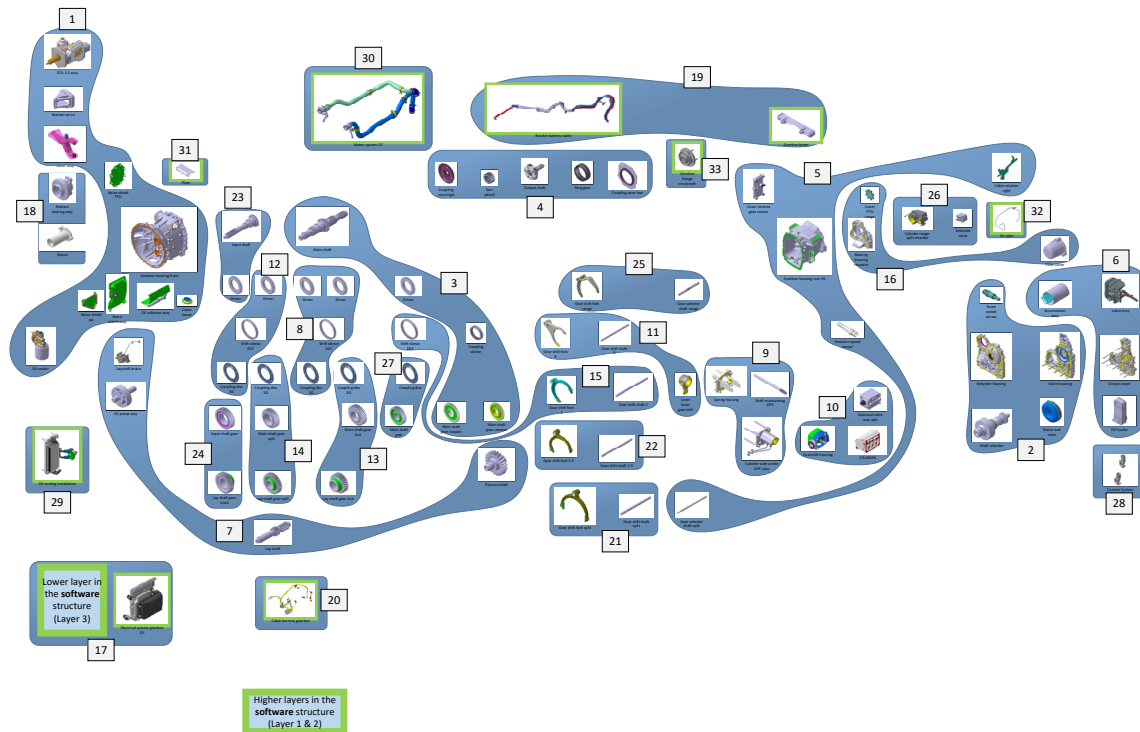
Type of technical relation	Relation weight
<b>Spatial</b>	3
<b>Information</b>	1
<b>Energy</b>	2
<b>Material</b>	2

The technical relation weights, which were used in the clustering analysis, were selected based on assumptions by the authors, see Table 2. It should therefore be stated that the assumed relation weights may affect the result of the clustering analysis, and that a future investigation is needed in order to fully identify the robustness of the result. Convergence of the result was found after 1500 iterations by using the IGTA++ clustering algorithm in MATLAB.

The result of the DSM clustering is presented in Figure 8. This proposal is relatively different from the transformed architecture adapted for Corporate B, which indicates that the existing modules were most likely not created according to product complexity concerns. Figure 8 also shows that several strategic conflicts within the clusters exist. For example, the oil cooler is a part of the oil collector assembly (cluster #21), which is not desirable from a strategic viewpoint since the oil cooler is not used at Corporate A. This result was expected, since a normal DSM does not contain any strategic aspects. In order to consider the strategic aspects, the IMM methodology was applied, with the result presented in Figure 9. As seen, the resulting clusters are somewhat different compared to the DSM, but now no strategic conflicts are present.



**Figure 8. Schematic illustration of the architecture proposed by DSM clustering**



**Figure 9. Schematic illustration of the architecture proposed by IMM clustering**

#### 4. Discussion

The original gearbox architecture used at Corporate A is a result of a large number of small design changes over many years. This approach may be beneficial in some aspects (e.g. improved quality etc.). However, sometimes larger architecture changes need to be made due to changes in business strategies or technology change, e.g. becoming a First-Tier supplier or electrification and digitalization of a system or subsystem. If only small changes are made over a long period of time, there is a high risk that the transformed product architecture becomes sub-optimised for the new task. It is highly important to use a robust methodology which supports the highly complex task when performing larger changes of a product architecture, otherwise important aspects may not be treated in an efficient and effective way. Since there is currently no accepted method on how sudden changes in business strategies could or should be reflected as changes of an existing modular product architecture, a new methodology is clearly needed. The core of the studied IMM method is a strategic DSM, which is a DSM that integrates a component-DSM with the Module Indication Matrix (MIM) that is a central tool in the MFD method. Hence, the component-DSM, which targets technical complexity, does not contain any strategic information and is therefore not capable of handling strategic concerns during the clustering stage. This limitation of pure DSM clustering has been illustrated and confirmed in this study. The presented case study showed that the IMM proposed a modular product architecture without conflicting MDs, and with reasonable module candidates from a technical complexity point of view. This indicates that IMM is a promising methodology, although several future case studies are needed to verify the limitations and advantages of the method and to identify, implement, and verify potential methodological improvements.

One possible improvement to IMM could be to identify and add new module drivers or to replace present MDs, e.g. MDs that represent strategic concerns in a multi-brand organisation. In addition, module drivers concerning global product performance (i.e. mass, natural frequency etc.) may also be useful, since it is sometimes possible to increase the performance substantially if components are integrated within a module. MDs concerning the dependency between software and hardware would also be desirable. For example, if the software is linked to the hardware depends on what layer the software belongs to in the software structure. In a lower layer, there is usually a strong connection between

hardware and software, e.g. software needed to control an actuator implies a strong interdependence between software and hardware and consequently there is a strong driver to integrate the software and hardware into a common module. However, the higher software layers, e.g. gear shifting or smart cruise controller, are not tightly connected to any specific hardware, since these software components only need real-time computing power. These aspects are currently not represented by the existing MDs, and hence future work within this area would be desirable.

Since the IMM currently only involves removing technical solutions due to strategic concerns, it would be highly interesting to investigate the result of a clustering analysis were strategic relations (based on the MDs) are added between the technical solutions, i.e. not only removing existing technical relations. These new strategic relations may add a new dimension to the original product architecture representation, enabling new ways to identify robust, low complexity and strategic modules.

When analysing the case study result of the DSM and IMM clustering, it is clear that both results are relatively different compared with the transformed architecture developed for Corporate B, which indicates that the existing modules were most likely not created according to product complexity concerns, but rather to other strategic aspects related how the product is manufactured and assembled (manufacturing complexity) and the need for different performance. This illustrates the importance of considering strategic aspects during the clustering stage, where the IMM has shown promising results.

In the case study, the DSM did not only contain physical technical solutions, since software components at a low level of decomposition also were introduced. The result of the case study indicates that this approach may be very useful, if the function allocation of the software is of interest during the clustering stage. This may be one way to represent the complete product architecture of a multidisciplinary product containing hardware, electronics and software, in order to enable its functions.

The weights for the technical relations used in the clustering analysis, were selected based on assumptions and experience by the authors. In addition to the assumed weights, multiple other weight combinations were also analysed. It should still be stated that the assumed relation weights will affect the result of the clustering analysis, and that a separate investigation is needed in order to fully identify the robustness of the result. By lowering the relation weight of the information flows, and at the same time increasing the importance of spatial relations, the result of the clustering analysis tends to look more similar to the existing modular architecture used at Cooperate A. On the other hand, if the weight of information flow is assigned with a relatively high importance compared with other types of relations, the proposed clusters tends to be extremely different from the existing architecture, indicating that the gearbox was probably modularised according to its mechanical design before any electronics or control software was added.

## 5. Conclusions

The experiences from the presented case study can thus be summarized as follows:

- Clustering a component-DSM with interactions that represent spatial relations and functional flows of energy, matter and signals may propose module candidates that address technical complexity, but not strategic business concerns.
- The DSM clustering result depends on the relative weights of the different types of component interactions that are represented by the DSM.
- Reverse engineering of the presented re-architecting case indicates that the IMM methodology is capable of identifying and proposing reasonable module candidates that address product complexity as well as company specific strategies.

## 6. Future work

Further cases, including new product development cases, have to be analysed in order to verify, generalize, and further improve the proposed approach into a robust and efficient methodology.

Since the aim is to develop a robust modular architecting methodology, it is crucial to perform a thorough study of how the weights of the relations in the DSM-representation, affect the clustering result. When the effects from the chosen weights on the proposed clusters are better understood, a logical

next step is to research on how strategic concerns, e.g. the hierarchical control software layer structure, affect the "best" choice of weights for different types of information and energy flows. It would also be highly interesting to investigate the results from a clustering analysis were strategic relations (based on the module drivers) are added as interactions between technical solutions. Finally, it is an open question, if robust modularization, especially in NDP, benefit from a refined or extended set of module drivers that also capture control software architecture and strategic concerns in a multi-brand organisation, system performance targets and dependency between sensors, control software, actuators and the mechanical structure.

## Acknowledgments

The financial support from SCANIA CV AB, as well as the open discussions with Scania CV AB and Volkswagen Truck & Bus GmbH is gratefully acknowledged.

## References

- Baldwin, C.Y. and Clark, K.B. (2000), *Design Rules, Volume 1, The Power of Modularity*, MIT Press, Cambridge MA., USA.
- Blackenfelt, M. (2001), Managing complexity by product modularisation: balancing the aspects of technology and business during the design process, PhD Thesis, Dept. of Machine Design, KTH Royal Institute of Technology, Stockholm, Sweden.
- Börjesson, F. (2010), "A systematic qualitative comparison of five approaches to modularity", *International Design Conference - DESIGN 2010, Dubrovnik, Croatia, May 17-20, 2010*.
- Börjesson, F. (2014), Product platform design - architecting methods and tools, PhD Thesis, Dept. of Machine Design, KTH Royal Institute of Technology, Stockholm, Sweden.
- Börjesson, F. and Sellgren, U. (2013), "Fast Hybrid Genetic Algorithm for Clustering Design Structure Matrix", *ASME Design Engineering Technical Conferences, Portland, OR, 2013*, pp. DETC2013-12041. <https://doi.org/10.1115/DETC2013-12041>
- Ericsson, A. and Erixon, G. (1999), *Controlling Design Variants: Modular Product Platforms*, ASME Press, The American Society of Mechanical Engineers, New York, USA.
- Erixon, G. (1998), Modular function deployment: a method for product modularisation, PhD Thesis, Dept. of Manufacturing systems, KTH Royal Institute of Technology, Stockholm, Sweden.
- Hölttä-Otto, K. (2005), Modular Product Platform Design, PhD Thesis, Helsinki University of Technology, Department of Mechanical Engineering, Machine Design, Helsinki, Finland.
- Otto, K., Hölttä-Otto, K. and Simpson, T.W. (2013), "Linking 10 Years of Modular Design Research: Alternative Methods and Tool Chain Sequences to Support Product Platform Design", *ASME Design Engineering Technical Conferences, Portland, OR*, pp. DETC2013-12523. <https://doi.org/10.1115/DETC2013-12523>
- Sanchez, R. (1994), "Towards a science of strategic product design", *2nd International Product Development Management conference, Göteborg*, pp. 564-578.
- Smith, J.S. and Duffy, A.H.B. (2001), "Re-using knowledge: why, what and where", *Proc. 13th International Conference on Engineering Design (ICED 01), August 21-23, Glasgow, UK*.
- Smith, P.G. and Reinertsen, D.G. (1995), *Developing products in half the time*, Van Nostrand Reinhold.
- Stake, R.B. (2000), On Conceptual Development of Modular Products, PhD Thesis, Department of Production Engineering, KTH Royal Institute of Technology, Stockholm, Sweden.
- Suh, N.P. (1990), *The Principles of Design*, Oxford University Press Inc., New York, NY.
- Ulrich, K. (1995), "The role of product architecture in the manufacturing firm", *Research Policy*, Vol. 24 No. 3, pp. 419-440. [https://doi.org/10.1016/0048-7333\(94\)00775-3](https://doi.org/10.1016/0048-7333(94)00775-3)
- Ulrich, K. and Eppinger, S. (1995), *Product design and development*, McGraw-Hill.
- Ulrich, K. and Tung, K. (1991), "Fundamentals of product modularity", *ASME winter meeting, symposium on issues in design/manufacturing integration, Atlanta*.
- Williamsson, D. and Sellgren, U. (2016), "An approach to integrated modularization", *26th CIRP Design Conference, Stockholm, Sweden*. <https://doi.org/10.1016/j.procir.2016.04.152>

Ulf Sellgren, Prof.  
KTH Royal Institute of Technology, Machine Design  
Brinellvagen 85, SE-100 44 Stockholm, Sweden  
Email: ulfs@md.kth.se