Available online at www.sciencedirect.com

ScienceDirect

Procedia CIRP 50 (2016) 613 – 617

www.elsevier.com/locate/procedia

26th CIRP Design Conference

An approach to integrated modularization

David Williamsson[a], Ulf Sellgren[b]*

[a]Scania CV AB, SE-151 87 Södertälje, Sweden
[b]System and component design, KTH Royal Institute of Technology. SE-100 44 Stockholm, Sweden

* Corresponding author. Tel.: +46-8-7907387. *E-mail address:* ulfse@kth.se

Abstract

A modular architecture is a strategic means to deliver external variety (to the customers) and internal commonality (to the manufacturing organization). A common view is that a module should be a physical and functional building block, with well-defined and standardized interfaces between modules, and that it should be chosen for company specific reasons. Existing methodologies, such as Modular Function Deployment with the Modular Indication Matrix (MIM) representation of identified company-specific module drivers, can be used to assist the task to identify modules. Other approaches, such as clustering of the Design Structure Matrix product representation, may be used to identify modules from a technical complexity point of view. A new methodology for product modularization that integrates technical complexity and company strategies is proposed in this paper. The core of the presented methodology is to adapt the component-DSM with MIM-strategies, before clustering this hybrid representation with the previously presented *IGTA++* clustering algorithm. The proposed methodology is exemplified and logically verified with an industrial test rig modularization case. The modular test rig architecture chosen with the new methodology is shown to have 53% less complexity, as defined by Pugh, compared with the original architecture, and it could potentially reduce the risk of design mistakes, and reduce the development time by up to 70%. It is also estimated that it would be possible to reuse up to 57% of the modules, in future test rig redesign projects, which indicates potentially large savings in cost and development time.

Peer-review under responsibility of the organizing committee of the 26th CIRP Design Conference

Keywords: DSM clustering; Modularization; Product architecture; System design

1. Introduction

Modularization is the decomposition of a product into building blocks (modules with specific interfaces, driven by company-specific reasons [1]). The main purpose of a modular architecture is to provide external variety, that is many possible product variants to the customers, and internal commonality, that is reduction of parts [2].

According to Hölttä-Otto [1], There are three main approaches to modularity that mainly are complementary [3]; *Heuristics,* Design Structure Matrix (*DSM*), Modular Function Deployment (*MFD*). *Heuristics* is based on an analysis of the pattern of flow of matter, energy, and information between function blocks, see e.g. [1]. The main purpose of *DSM*-based approaches is to minimize technical complexity by doing clustering of the system components in a way that the technical interactions between clusters of components are minimized, i.e. complex interactions are grouped within clusters. A very efficient DSM-clustering, algorithm, referred to as IGTA++, was proposed in [4]. Heuristics and DSM approaches address technical complexity, but not strategic objectives [2]. *MFD* [5] is a five-step method for translating customer requirements into a modular architecture, while considering the strategic objectives

Peer-review under responsibility of the organizing committee of the 26th CIRP Design Conference
doi:10.1016/j.procir.2016.04.152

(described using twelve predefined *Module Drivers*). Project data is captured in three core matrices; The QFD House of Quality, the Design Property Matrix (DPM), and a Module Indication Matrix (MIM) relating components and the 12 module drivers, e.g. planned upgrades, separate testability, carry over, etc. An attempt to balance technical independence and product similarity with a hybrid clustering approach, referred to as R-IGTA, integrating DSM clustering with MFD-based DPM and MIM clustering was presented in [6]. The main contribution of the R-IGTA approach is that it offers technical and strategic concerns to be treated and balanced simultaneously. The drawback of the method was the prerequisite to actually define the three matrices, which in an actual development process are created sequentially, and to do that with the same level of detail. The method has thus limited capabilities to assist the product development process in a “smooth” way. An efficient modular clustering method should preferably offer support to the entire process. That is, it should be possible to analyze functional and technical complexity in the early conceptual phase. This can be done with DSM clustering. The model (matrix) should be scalable, allowing details and new features and domains of knowledge to be added as they are created. One method to enable scalability in this respect is to allow the DSM representation to also allow strategic matrix components. Blackenfeldt [2] proposed a logically complete two-layered DSM as a means to treat technical dependencies between system components and strategical conflicts, i.e. intrinsic conflicts in the module drivers recognized in the MIM. Although logically complete, this technical-strategic method has some practical limitations [7] that limit its usability in systems engineering of complex systems. The scope of the research presented in this paper is to develop a scalable DSM-based method that allow strategic considerations to be conditionally represented in a technical DSM, and that proposes module clusters that do not contain strategic conflicts. Such a novel approach is proposed and exemplified in this paper.

The proposed integrated modularization approach, which has a strategically adapted DSM as core representation, is presented in chapter 2, exemplified with a case study in chapter 3, discussed in chapter 4, and concluded in chapter 5.

2. Proposed modularization approach

An approach to integrate the DSM and MFD method will be illustrated in this section. The core of the DSM method is the DSM matrix, while the MIM is the core of the MFD method. Since the DSM, which is an intradomain matrix, and MIM, which is an interdomain matrix, contains different kind of information, they cannot be directly added. We therefore propose a new method, which aims to transform some of the information from the MIM to the DSM, enabling both technical and strategical information to be analyzed simultaneously.

2.1. Transfer data from the MIM to the DSM

The starting point of the proposed methodology is a well-defined product architecture, see Fig. 1, where all technical relations (i.e. flow of energy, geometry, signal, and material) are represented between the technical solutions (A, B, C and D). The technical solutions also need to have boundary conditions, in form of strategies, which are specified in the MIM. One of the main purposes of the MIM is to identify conflicting module drivers, i.e. mismatches in strategies within a module candidate. The proposed method implies that this information could be imported to a DSM, via a *strategy transfer DSM*.

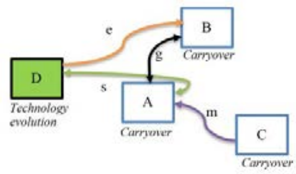

Fig. 1. Example of a graph-based representation of a product architecture.

Before any data could be transformed from the MIM to the strategy transfer DSM, all conflicting module drivers needs to be specified with a minus sign, see the upper part of Fig 2.

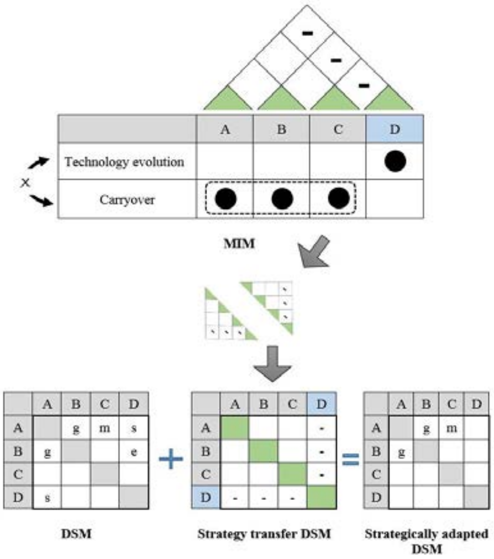

Fig. 2. Illustration of the proposed methodology

In the example shown in Fig. 2, technical solution “D” has a conflicting module driver to the other technical solutions. It therefore needs to be separated and is as a result not allowed to have a relation to any of the other technical solutions. In order to remove all these unallowable relations, a new upper triangular matrix was added on top of the MIM. This triangular matrix contains information (minus signs) on which technical solutions that have conflicting module drivers. Since we want to remove the affected relations in both directions, the lower triangular matrix is symmetric to the upper. In this example, only two module drivers were specified, however the same principle could be used for multiple drivers.

Finally, the *strategically adapted DSM* is calculated by adding the original DSM with the *strategy transfer DSM*. All relations interfering with a minus sign gets removed, while empty cells remain unchanged.

2.2. Cluster the strategically adapted DSM

The Strategically adapted DSM could be treated as a regular component-DSM during the clustering stage, and therefore a normal DSM clustering method could be used. It is however necessary to check that none of the clusters contains conflicting module drivers, after performing the clustering analysis.

2.3. Analyze the result

After performing the clustering analysis of the example product, the modular product architecture shown in Fig. 3 can be found. In this example, all relation weights were assumed to be equal, which is a simplification. It should be stated that the removed technical relations need to be inserted back into the DSM again, after finding the modules, in order to represent all interactions/interfaces.
When adding the strategies to the DSM, the resulting modular architecture will normally take another shape. For example, in our simple modularization example, technical solution “D” would be a part of module 1, if no strategies were added.

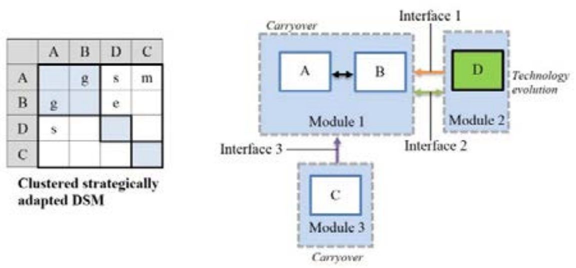

Fig. 3. Result of the modularization example

In this example, adding strategies to the DSM results in an increased number of modules and interfaces, which cause a higher product complexity (108%), compared with clustering the original component-DSM. Product complexity C_f is here quantified with a measure proposed by Pugh [8]:

$$C_f = \frac{K}{f}\sqrt[3]{N_p N_t N_i}$$

where K, f, N_p, N_t, and N_t, is the number of product functions, the number of parts, the number part types, and the number of interconnections/interfaces, respectively.

However, the example architecture had no conflicting module drivers within modules, meaning that this architecture would be strategically more advantageous than the one obtained by clustering the technical DSM.

3. Case study

Since the core of the proposed methodology is to modularize a product by both taking the company strategies and the technical complexity into consideration, the function of each technical solution is of less interest when describing the method. Hence, none of the functions will be illustrated in the figures below.

3.1. The test rig

The engine test rig, named *F16*, is one of the latest and most advanced test rigs at Scania, see Fig 4. It consists of multiple subsystems, which primary purpose is to either measure the engine performance, simulate running conditions, or to prepare the engine for a test. Many of these subsystems include several technical disciplines, and therefore a multidisciplinary approach was essential in the implementation stage.

The original engine test rig was identified to have an integral product architecture, and it was tailor made for a specific testing purpose. It was however of interest to modularize the test rig in order to identify the benefits a modular version of the test rig could potentially offer.

In order to represent the test rig in a compact format, a component structure diagram was created, see Fig 4. This diagram was created after decomposing the technical solutions and their (purpose) functions. In the diagram, the company strategies were specified, in addition to the technical solutions and relations. A technical solution with a green color in Fig 4 indicates a technology evolution module driver, while gray indicates carryover.

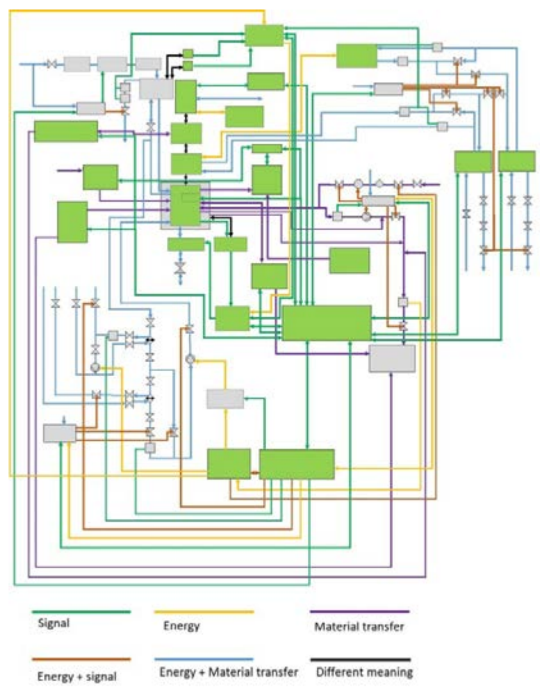

Fig. 4. Component structure diagram of the *F16* engine test rig

3.2. System requirements

The *F16* test rig had an extensive requirement specification, mainly concerning the high performance demands on the different technical solutions. The test rig also had to allow all types of Scania engines to be tested.

From a strategic viewpoint, it was identified that some of the technical solutions would need to be upgraded during the lifetime of the test rig, mainly due to technology evolution. It is also desirable that the modular test rig architecture would enable resource savings, mainly concerning development time and cost.

To cope with all these demands, it was clear that both the technical complexity and company strategies needed to be considered during modularization. The aim was therefore to create a modular test rig architecture, which would save resources, enable large configuration flexibility and reduced product complexity, compared with the original integral architecture.

3.3. Proposed modular architecture

The test rig was modularized by clustering both the original DSM and the new proposed strategically adapted DSM. The starting point of the modularization was the component structure diagram in Fig 4, which was represented with a DSM. The original component-DSM and the strategically adapted DSM were then clustered in MATLAB by using the *IGTA++* clustering algorithm. The relation weights presented in Table 1 were used in the clustering. These relation weights were chosen after analyzing the results from several weight combinations, neither one did show a result that would fulfill the aim of the modularization better.

Table 1. Chosen relation weights.

Type of technical relation	Relation weight
Geometry (also referred as spatial)	2
Signal (also referred as information)	1
Energy	1
Material	2

After the clustering stage, the resulting modular architectures were represented as two schematic illustrations, shown in Fig. 5 and Fig 6. As earlier described, a technical solution with a green color indicates a *technology evolution* module driver, while gray indicates *carryover*. The blue shapes represent the non-conflicting modules, containing the strategically feasible clusters of technical solutions, while the orange shapes represents modules containing conflicting module drivers.

When analyzing the result of the *strategically adapted DSM*, it was clear that none of the modules contained conflicting module drivers. This indicates that the proposed integrated modularization method seems to cover that important strategic aspect successfully.

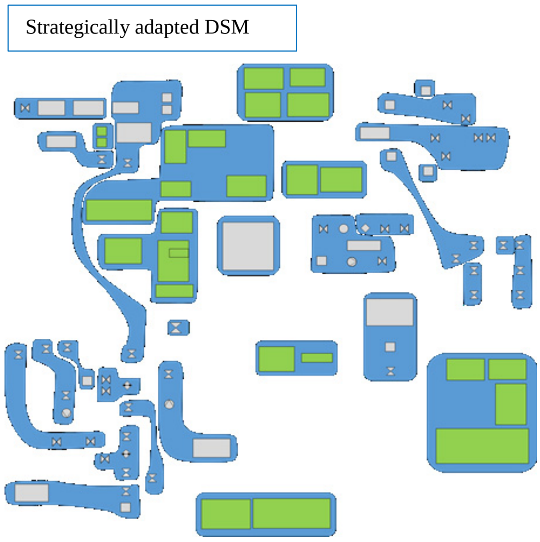

Fig. 5. Schematic illustration of the proposed modular architecture.

When analyzing the result of the original DSM, it was clear that several of the modules contained conflicting module drivers, see the orange modules in Fig. 6. It was also possible to see that the two modular architectures (Fig. 5. and Fig. 6.) did not consist of the same type of modules.

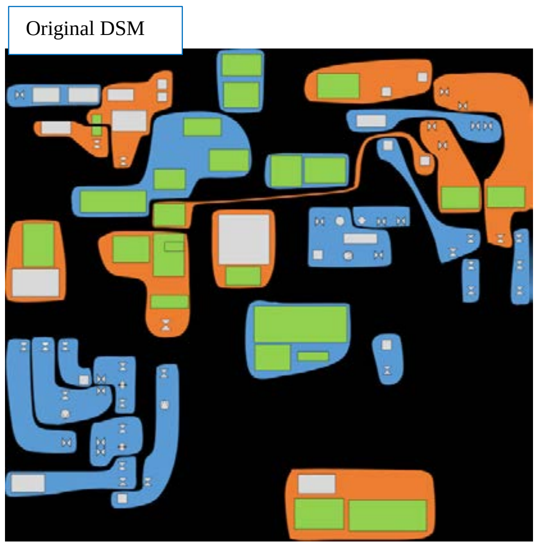

Fig. 6. Schematic illustration of the original modular architecture

To assess how the proposed integrated modularization method affected the result, the technical complexity was then calculated, see Table 2 below.

3.4. Results

By comparing the result of the *strategically adapted DSM* and the original DSM, it was clear that clustering of the original DSM resulted in a slightly lower technical complexity. This was mainly due to the smaller amount of modules. However, the lower complexity of the technically-based clustering came at the expense of conflicting module drivers. It was also possible to see that the two modular architectures had a significantly lower complexity compared to the integral architecture.

Table 2. Complexity factor for the different analyses.

Analysis	Complexity factor
Integral architecture (before modularization)	127
Strategically adapted DSM	60
Original DSM	54

The clustering result from the *strategically adapted DSM* was chosen to be the most beneficial modular architecture for Scania, mainly since it did not contain any conflicting module drivers and it still had a low technical complexity.

The chosen modular test rig architecture was finally evaluated in terms of the benefit it could offer, compared to the integral architecture. The main potential benefits were the reduced design and lead times during the development, enabled by the reduced complexity (-53%) and the ability to develop modules in parallel. It was also estimated that the chosen modular test rig would enable a large configuration flexibility, meaning that up to 57% of the modules could be reused in future redesigning of the test rig.

4. Discussion

For the presented case, the proposed integrated modularization approach provided a modular test rig architecture with 53% less complexity compared to the original integral architecture. The reduced complexity could potentially reduce the design and lead time. Reducing the design time will also lower the development cost. The case study also indicates that the development lead time may be reduced by up to 70%, by enabling parallel development of the modules. Furthermore, the investment cost and lead time could be even more reduced by reusing up to 57% of the carryover modules. It is therefore important that no conflicting module drivers are contained in the modules.

By reducing the technical complexity, the risk of making design mistakes will also be reduced, due to fewer communication points between the design teams.

One potential drawback of the modular test rig is a reduction in the overall performance, e.g. measurement accuracy and/or precision. The quantitative reduction was however not possible to predict, before designing and manufacturing the physical test rig.

5. Conclusions and future work

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

- Clustering of a standard component-DSM, results in a modular architecture with significantly reduced complexity, but with modules that contain conflicting module drivers.
- The case study indicates that the proposed integrated modularization method, based on strategically adapted DSM clustering, proposed modules with significantly reduced complexity (-53% in the presented case) that did not contain conflicting module drivers.

Further cases have to be analyzed in order to verify, generalize, and further improve the proposed approach into a robust and efficient methodology.

Acknowledgements

Scania CV AB is acknowledged for enabling the presented research.

References

[1] Erixon, G. Modular function deployment : a method for product modularisation, PhD. Thesis., KTH, Dept. of Manufacturing systems, Stockholm, 1998.

[2] Blackenfelt, M. Managing complexity by product modularisation : balancing the aspects of technology and business during the design process, Doctoral Thesis., Dept. of Machine Design, KTH Royal Institute of Technology, Stockholm, 2001.

[3] Hölttä-Otto, K. Modular Product Platform Design, Doctoral Dissertation, Helsinki University of Technology, Department of Mechanical Engineering, Machine Design, Helsinki, Finland. 005.

[4] Borjesson, F, Sellgren, U. (2013A) Fast Hybrid Genetic Algorithm for Clustering Design Structure Matrix. ASME Design Engineering Technical Conferences, Portland, OR, 2013, Paper number DETC2013-12041

[5] Ericsson, A., Erixon, G. Controlling Design Variants: Modular Product Platforms. ASME Press, The American Society of Mechanical Engineers, New York, 1999.

[6] Borjesson, F, Hölttä-Otto, K. A module generation algorithm for product architecture based on component interactions and strategic drivers. Research in Engineering Design 25(1):31-51, 2013.

[7] Börjesson, F. Product platform design – architecting methods and tools, Doctoral Thesis., Dept. of Machine Design, KTH Royal Institute of Technology, Stockholm, 2014.

[8] Pugh, S. Total design, Addison-Wesley, 1990.