



HAL
open science

Multiple-Domain Design Scorecards: A method for architecture generation and evaluation through interface characterization

Marija Jankovic, Vincent Holley, Bernard Yannou

► To cite this version:

Marija Jankovic, Vincent Holley, Bernard Yannou. Multiple-Domain Design Scorecards: A method for architecture generation and evaluation through interface characterization. *Journal of Engineering Design*, 2012, 23 (10-11), pp.743-763. 10.1080/09544828.2012.706270 . hal-00748703

HAL Id: hal-00748703

<https://hal.science/hal-00748703v1>

Submitted on 17 Mar 2013

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Multiple-Domain Design Scorecards: A method for architecture generation and evaluation through interface characterization

Marija Jankovic, Vincent Holley, Bernard Yannou

Ecole Centrale Paris, Laboratoire Genie Industriel

Grande Voie des Vignes – 92290 Chatenay-Malabry, France

Abstract: High-tech mechatronic system design, such as tools for oil-well drilling and exploitation, requires that engineers from multiple domains concur on architectural solutions due to a large number of interdependent system performances and design parameters that are shared across architecture interfaces. Supporting the designers into envisioning probable challenges in architectures early in design, is of utmost importance to ensure project quality while minimizing costs and delays. In this paper, we present the Multiple-Domain Design Scorecards (MDDS) method to facilitate architecture generation and assessment through architecture interface characterisation as well as to support design process management. In order to do so, we propose to semantically enrich the Design Structure Matrix (DSM), Domain Mapping Matrix (DMM) and Quality Function Deployment (QFD) representations to capture project data and expertise related to concept generation. In addition, we propose six types of Design Assessment Cards (DACs) that both support design space exploration and highlight design challenges of potential system architecture. The proposed approach is being implemented and tested in an industry context. A case study in the oil industry brings to the fore some of the advantages and challenges in the implementation process.

Keywords: Multi-disciplinary systems, complexity management, dependency management, collaborative design, concept generation, concept assessment

1 Introduction

Designing complex engineering systems remains challenging. Usually, engineers decompose global product architectures into several subsystems, components and parts (Yassine et al., 2003, Eppinger and Salmien, 2001, Browning, 2009, Yassine and Braha, 2003). The advantages of this approach are, however, rapidly offset by the difficulties associated with tracing the impacts of one subsystem design onto another (i.e., managing interfaces among subsystems) (Yassine et al., 2003). This phenomenon is also referred to as a “dependency problem” as its objective is to predict how the design of one subsystem will affect the respective designs of others.

When addressing the notion of complex engineering systems, we generally regard large systems as being tightly connected subsystems. However, not all complex systems are necessarily large; even relatively small systems, such as high-tech mechatronic systems can be complex, albeit with a large number of dependencies at high impact strength. The importance of managing dependencies or interfaces in these systems lies in the extreme difficulty in tracking the impacts of one subsystem design onto other subsystems. This is due to the high number of interdependent design parameters shared through interface definition innovation integration implying the lack of previous knowledge and in most cases a lack of appropriate methods to support the goal of capitalizing on experience related to the dependencies and their impact on the system

In this design situation, engineers from multiple domains must compromise on possible architectural solutions in concordance with the overall system performances, which itself is under the impact of design constraints; constraints that each domain has for a given architecture interface. With the aim of therefore ensuring project quality and minimizing costs and delays, it is of utmost importance to predict all probable challenges of the architectures under consideration that are consequences of these interdependencies.

In this paper, we propose a method of Multiple-Domain Design Scorecards (MDDS) to support design teams to characterise interfaces and assess their potential impact on the global system architecture early in the design. MDDS is based on the semantic enrichment of classical matrix-based methods (Design Structure Matrices, Domain Mapping Matrices and Quality Function Deployment). Six types of Design Assessment Cards are proposed in support of design space exploration based upon interface definition as well as highlighting design challenges of a potential architecture. In the section 2, we discuss various approaches for concept generation and assessment as well as ones that address modeling complexity and dependencies within concept design. In the section 3, we outline our research method and introduce a case study from the oil industry. After explaining the MDDS method in detail, the results of MDDS implementation are discussed. This paper ends with an overview of potential future research.

2 Background

Analyzing and designing complex systems and managing dependencies is the objective of several areas of research, including graph theory, neural networks, matrix-based approaches and qualitative physics. One major approach aimed at understanding and managing such dependencies is axiomatic design. Suh (1990, 1995, 2001)

demonstrated the correlations between customer needs (customer attributes in the customer domain), their transcription into functions (functional requirements in the functional domain), their manifestation in the physical system (design parameters in the physical domain) and translation into processes (process variables in the process domain). It is generally accepted that axiomatic design systematizes complexity analysis and thus facilitates complexity management. Numerous scholars have addressed these issues by mapping different dependencies so as to manage and reduce complexity in the design process (Sinha et al., 2001, Yassine et al., 2003).

In conceptual design, several studies propose to integrate different interface definition in architecture definition and generation. Moullec et al. (2012) define two types of physical interfaces: crisp and uncertain. The developed Bayesian Network model takes into account these two types of interfaces in the generation of all possible architectures and estimate global architecture performance uncertainties. Albers et al. (2011) propose the use of Contact and Channel Approach (C&C-A) for system architecture generation. The base for this approach towards incremental creation and the modeling of mechatronic systems is that the technical system's function is considered to be form dependent. C&C models represent the interactions between systems, subsystems and part through working surface pairs -WPS (geometric interfaces between artifacts or between artifact and environment) and channel and support structures – CSS (physical component or volumes of liquids, gases or fields directly connecting two WPS). Ziv-Av and Reich (2005) develop a subjective objective system (SOS) for the generation of optimal product concepts. The SOS approach integrates information on market, organization and technology for generation product concepts. The authors state that this approach can be accommodated to add more detailed information on interactions between the components but the proposed mathematical model would not support such information. Additionally, Wyatt et al. (2012) propose to use a network structure to support product architecture generation. This network structure representation of experts' knowledge uses qualitative constraints on the arrangement of the components and the interfaces between them. Interfaces between components are represented as one of one of constraint types for architecture generation, defining which component types can be connected to which interface types. The types of interfaces that are modeled are: structural, behavioral, assignment and geometrical. These interface types are taken into account for product architecture generation but not integrated into performance estimation. A Markovian model is used to compare different product architectures with respect to time needed to stop the propagation of development delays within the network (Mane et al., 2011). The states of the Markov chain represent system components and transition probabilities as dependency strengths between systems. Component interfaces are therefore taken into global architecture estimation as probabilities that depend upon the strength between two components.

Another research field that contributes greatly in complex system modeling and simulation is qualitative physics. Automating the failure-mode-and-effect analysis (FMEA) on architectures of complex systems has also already been addressed within the Artificial Intelligence community, by the qualitative reasoning approaches. Qualitative reasoning (QR) studies dynamic systems (regulated by kinematics, kinetics, dynamics, and flow conservation laws) in a more qualitative manner than with precise numerical computations, which appropriately corresponds to design situations (see (Bredeweg B., 2003, Bourseau P., 1995)). In the QR community, efficient compositional modeling approaches have been proposed to generate dynamic system architectures. This provides the dynamic equation set for the whole system (Bredeweg B., 2003) and the system failure-modes-effects-and-criticality analysis (FMECA) (Price,

2000, Struss, 2003, Struss and Fraracci, 2011). Our objective is to focus on integrating failures related specifically to system interfaces and design parameters shared across multiple domains.

The research field that is particularly fertile ground for addressing interface definition and management in conceptual design concerns matrix-based approaches. Researchers have been developing matrix-based approaches since the 1960s (Steward, 1962, Steward, 1981). The matrices have been differently named through the years: dependency structure matrices, interaction maps, dependency maps, incidence matrices, and problem solving matrices (Browning, 2009). The information represented by graphs or matrices is essentially the same, and most problems processed by matrix depictions can be solved in an efficient manner by operations made available from graph theory (Brualdi and Ryser, 1991, Andrasfai, 1991)

Several scholars have attempted to classify matrices (Malmqvist, 2002, Li, 2009, Lindemann et al., 2009, Browning, 2009). Browning (2009) has identified two types: static and time-based. Static matrices represent elements that coexist while time-based matrices order rows and columns to indicate flows in time. According to Lindemann et al. (2009), matrices can be classified as intra-domain, inter-domain or multiple-domain. Intra-domain matrices or DSMs are square matrices representing dependencies between like elements (product components, performance attributes, engineering requirements). Several scholars (Pimmler and Eppinger, 1994, Jarrat, 2004) have provided highly valid overviews of different dependency types for these types of matrices. On the other hand, inter-domain matrices (also called Domain Mapping Matrices, or DMMs) (Danilovic and Börjesson, 2001, Gorbea et al., 2008), consider dependencies between different domains (e.g., between functional requirements and product components). One well-known DMM is a Quality Functional Deployment (QFD) matrix (Hauser and Clausing, 1988). Finally, Multiple-Domain Matrices (MDM) represents dependencies between elements across domains (Maurer, 2007) combined within one matrix. Li (2010, 2011) has studied the different matrix handling techniques to cluster or decompose design problems in the most adequate way in order to use them in design synthesis rather than analysis.

In particular, several research studies in this field have been working on interface characterization and dependency definition as to support architecture definition and generation early in design. Hellenbrand and Lindemann (2008b) use DSM to support the selection of product concepts. They present a compatibility matrix that captures possible compatibilities and their respective weights in relation to two different product components. In the proposed consistency algorithm, the existence of the interface is taken into account in order to offer all possible product concepts for selection. Bryant et al. (2005) use a set of DSMs to represent dependencies between functions and components, therefore allocating product component to function from the accordingly developed design repository. This allocation allows for generating possible product architectures that satisfy defined functions and functional flows. Wyatt et al. (2008) propose using DSMs and DMMs to capture the rules governing product architectures and use Component DSM to compare various architecture concepts. In their work they juxtapose different component and interface types. By mapping component alongside component types in parallel with component and interface types they are able to express constraints governing architecture definition. Although they take into account interface types, the data that is considered here is the possibility of mapping component types and interface types, without any data relating to the design parameters that need to be managed in the interface-type design space. Sharman and Yassine (2004) propose to integrating three levels in system representation into DSMs:

global design rules, interface rules and intramodule design rules. The interface in this product architecture is used afterward for calculating architecture visibility and dependency. Pimmler and Eppinger (1994) define four types of interaction in product development: spatial, energy, information and material. They propose a weighing scale to represent a coupling coefficient (from -2 to +2). The proposed method is of particular interest for the analysis of product design decompositions in order to propose another clustering into architecture and product chunks. This approach measures the interaction strengths by identifying different interaction types. Matrix-based approaches are particularly useful here because they visually represent the relationships between the different elements, making it both more legible and articulate, which will also be in favour of industry implementation. Scholars continue to discuss whether matrix-based or graph theory approaches are more effective, and whether certain numbers of subsystems or components are more easily represented using one approach or the other. Moreover, some researchers have emphasized the fact that both approaches yield qualitative representations of dependencies (Kreimeyer, 2009) and fail to consider data related to dependency type.

In most of cases, the majority of the developed approaches take the presence of interfaces into account in order to support concept design but only a few of them address the typology and the definition of these interfaces. We argue that interfaces define functional flows and therefore impact global system performances. Designer's support for system analysis in this phase should integrate both interface data and their impact on global architecture. In this paper, we try to address these problems by enriching matrices with product interface data. We use matrix-based approaches because of their relative ease of use and implementation potential. The model proposed in this paper is based on gathering expert knowledge on functional flows, physical interface definition and performance estimations to result in a proposal of 6 types of Design Assessment Cards (DACs) that support design space exploration and possible architecture challenges identification. Moreover, most studies focusing on these approaches attempt to address the entire product design space. Our main hypothesis is that by managing interface constraints, we will be able to manage the entire design process. Therefore, the matrices developed in this work address collaborative, multi-domain interface design spaces.

3 Research Design

An oil extraction company initiated this research study after experiencing significant project overruns in terms of both time (40 to 150%) and costs (up to 100%). This company's projects undergo many design iterations; designs may suddenly need to be redeveloped from scratch, and on other occasions they may simply be scrapped after several years of development. In addition, most products are launched with reliability problems despite several years being spent on their development and realisation. Two to three years of re-engineering are needed to achieve expected reliability having observed problems during actual use. This company manages more than 140 concurrent development projects of exploitation tools with typical durations of 7 to 15 years at an average cost of 5 to 10 million dollars per year per project.

The global research approach used in this study is Design Research Methodology (DRM) (Blessing and Chakrabarti, 2009), which has four steps: (a) defining the research goals, (b) identifying the main factors for a problem, (c) developing design support, and (d) evaluating and validating the proposed support. To define research goals, we met with the design team and managers and considered a

sample of 10% of all ongoing projects. Process data were gathered through observations and interviews with different team members. There were 14 open-ended interviews of 1,3 hours with different roles in the project design team: the project manager, the chief architect engineer, the mechanical, electrical and software engineers, the physics engineer, the after sales engineer, the maintenance engineer, the geologist, the purchase engineer and heads of all 4 engineering departments (mechanical, electrical, software and physics). Using the value stream analysis for the design process, it was determined that the majority of problems are induced by inadequately managed physical interfaces and their impacts onto the design process. Ninety percent (90%) of the reengineering loops are caused by problems in the definition of physical interfaces. Knowing that only one additional re-engineering loop adds up to 1/3 of the global project time to the scheduled time to market, we can see that proper definition and management of physical interfaces is necessary for completing the project in a timely fashion. Our initial hypothesis postulated that effectively integrating multiple domains in a high-expertise design process can be accomplished by managing design interfaces and identifying potential collaboration conflicts and tradeoffs. This was discussed and prioritized with all interviewed project team members. To evaluate and validate our hypothesis, we conducted one complete industrial case study on the Power Electric Controller (PEC). Validation results are presented and discussed in section 6 of this paper.

4 Case Study of Highly Constrained Multi-Domain Design

The Power Electric Controller (PEC) is a regulator board used to generate the +3.3V and +1.9V power supplies for motor control and main controller boards. We present partial functional analysis of the PEC in Figure 1. Functional analysis is considered to be a first step in the MDDS process and will be discussed further.

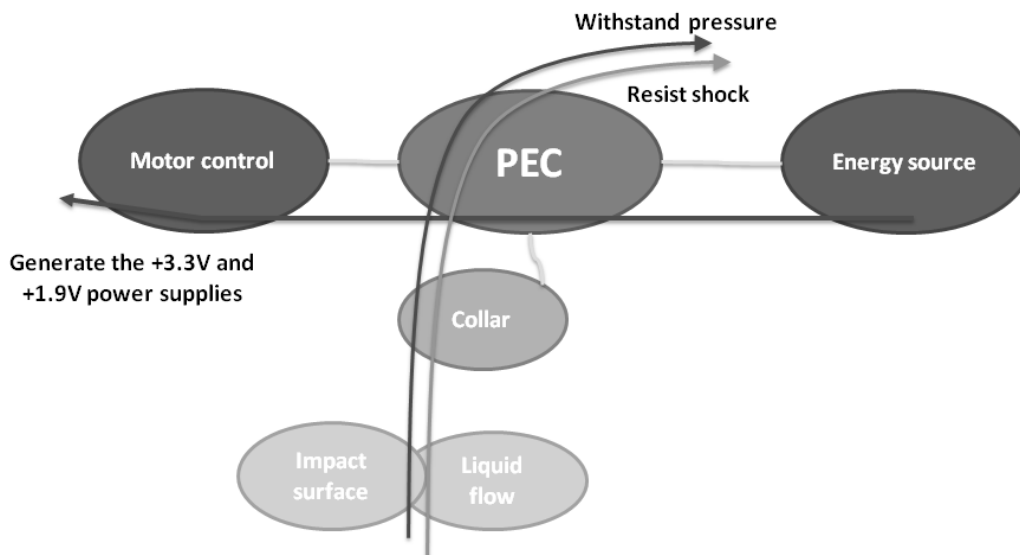


Figure 1: A partial external functional analysis of the PEC.

The “collar” is the product casing that contains the PEC. The design team identified two external environments: the impact surface (rock) against which the product experiences impact shocks and the liquid flow (the mixture of oil and mud) in which it operates. “Withstand pressure” and “Resist shock” are thus environmental functions the PEC system must comply with.

Based on the functional breakdown of the PEC, the team proposed the architecture principle shown in Figure 2. In this work, we refer to these subsystems as modules. For each module there are several design possibilities that we will henceforth refer to as alternative technical solutions.

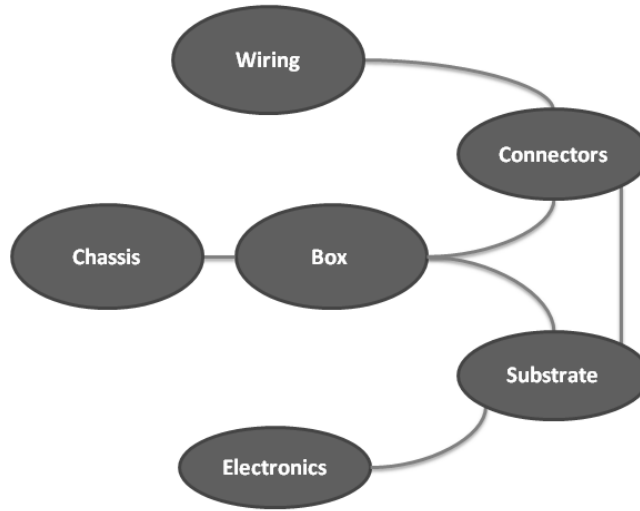


Figure 2: PEC architecture principle.

We present the overall product architectures resulting from brainstorming sessions with different design experts in Table 1, along with potential technical solutions for each module.

Table 1: Representation of a part of proposed architecture (constituted of modules) and different alternative technical solutions that were brainstormed and discussed for the project.

Department	Module	Technical solution
Mechanics	Chassis	I
		Delta
		Pivot
		Reverse Delta
Packaging	Box	HPHT
		2 faces with box
		Pivot
Electronic	Electronic (parts)	<i>One type: no specific name</i>
	Substrate	<i>One type: no specific name</i>
	Connectors	2 Hermetic Integrated
		2 non Hermetic Integrated
	Wiring	<i>One type: no specific name</i>

The PEC is composed of a “substrate” supporting “electronics” components, both of which are contained in a “box.” The PEC communicates with other systems (motor control and energy source) through “connectors” and “wiring.” The box was

mounted in a “chassis.” The overall assembly is fixed in the “collar”; as this part is outside the system’s limits, it is not taken into account in this work. The mechanics design department is in charge of designing the chassis, the packaging group designs the box, and the electronics group designs the other modules. Figure 3 illustrates the internal functional analysis of the PEC structure (i.e. functional flows throughout the architecture), thin links representing physical connections.

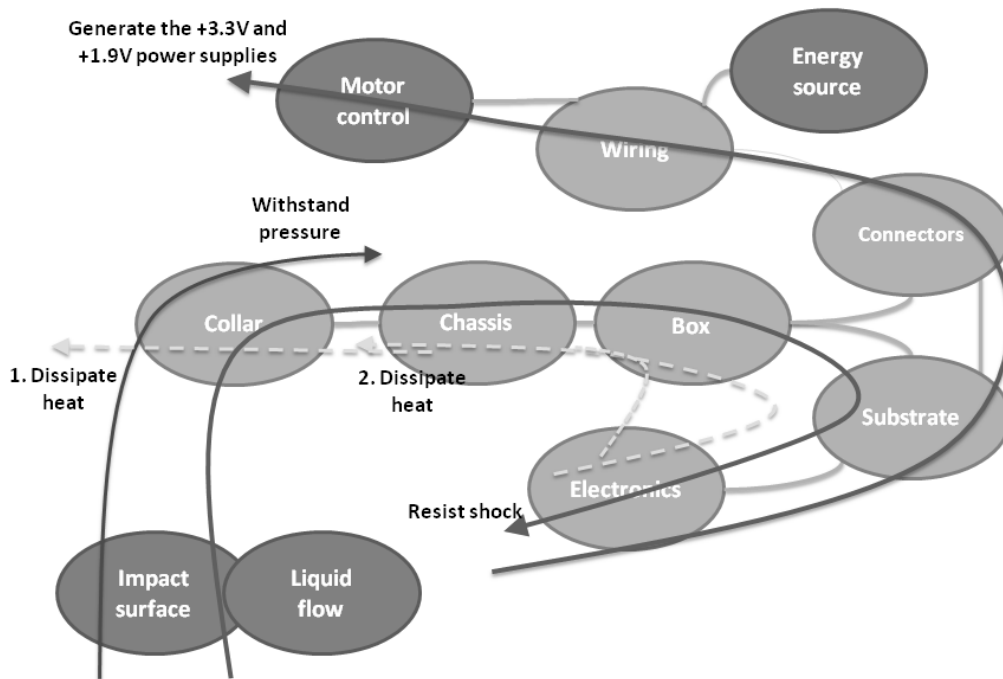


Figure 3: Internal functional analysis of the PEC.

The functional analysis follows the definitions and processes outlined by Aoussat et al. (2000). The orientation of flows stresses the way the function is applied. The flow of function “Generate the +3.3V and +1.9V power supplies” starts flowing through “electronics” component where voltage is regulated; +3.3V and +1.9V and sent, to “motor control” component via the “substrate”, “connectors” and “wiring” components (see arrow in Figure 3). As the “electronics” component generates power, it also generates heat, creating a need for “heat dissipation”; thus the “heat dissipation” function originates at the “electronics” component. Heat either passes through the “substrate”, the “box” or the “chassis” components where it is then dissipated (see discontinuous arrows in (Figure 3). Depending on the chosen solution for the “chassis”, it can pass through the “chassis” and be dissipated through the “collar”. In both modes or pathways of heat dissipation, air flow against the “box”, “chassis” or “collar” may also facilitate the process (see discontinuous arrow in Figure 3). The PEC must resist the surrounding liquid flow and, more specifically, have “withstand pressure”. If the function is achieved by leaving the liquid flowing outside the system through the “collar” (see continuous arrow in Figure 3), only the collar needs to be pressure resistant. The PEC must also “Resist the shock” caused by constant impact of the “collar” against rock. Shocks are transmitted to the “electronics” via the “chassis”, “box” and “substrate” (see arrow in Figure 3).

5 The Multi-Domain Design Scorecards Platform

5.1 Enrichment of design model representations

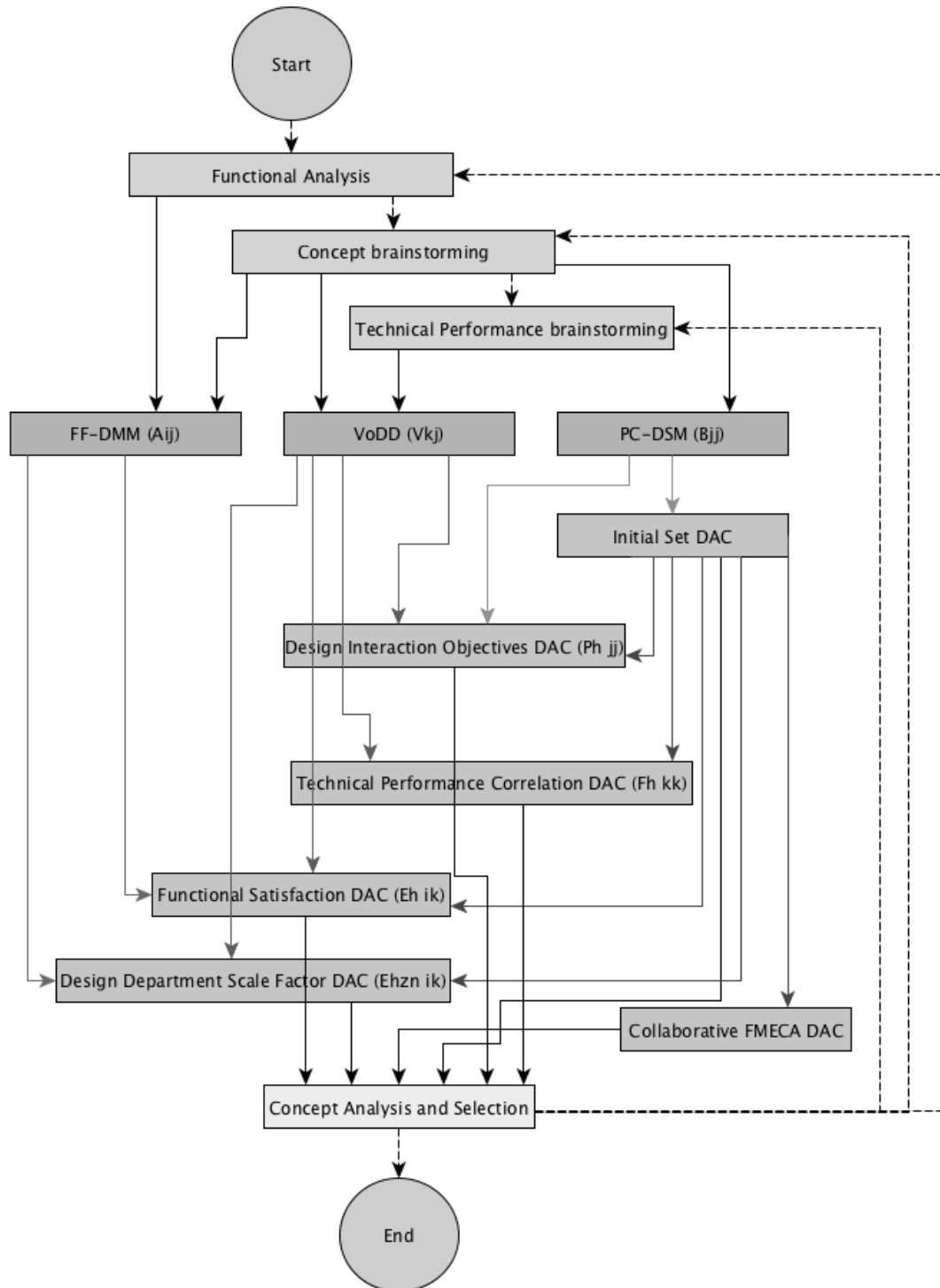


Figure 4: MDDS overall process.

The MDDS method incorporates three matrices for design concept analysis and evaluation: a Functional Flow - Design Mapping Matrix (FF-DMM), a Physical Connections - Design Structure Matrix (PC-DSM) and a Voice of Design Department Matrix (VoDD). Overall MDDS process is presented in Figure 4. After the Functional

Analysis and concept brainstorming, the three matrices are used to capture the concept design data. This data is then used to compute 6 DACs. Each of the matrices and proposed semantic enrichments are presented further in this section of the paper. The DACs are presented and illustrated in the section 5.2.

FF-DMM is a cross-functional flow and architecture mapping matrix populated with data retrieved from the functional analysis as explained in section 4 of this paper. It uses a DMM format enriched by integrating a functional flow. This matrix (see Figure 5) is a result of a functional analysis and a brainstorming workshop related to concept generation. Functions are expressed in rows, while modules and their alternative technical solutions are expressed in columns. The data contained in the matrix represent the function flows through proposed modules and technical solutions. Each function can have several functional flows (see Dissipate heat function in Figure 3). The numbers represent the order of on functional flow through the product architecture (i.e. modules). For example, the “Generate power” function passes first through electronics numbered 1, then through substrate numbered 2, through connectors numbered 3 (where we have two options/technical solutions) and in the end through wiring numbered 4.

		Modules											
		Chassis		Box		Connectors	Collar	Substrate	Electronics	Wiring			
		Alternative Technical Solutions											
		1	Delta	Pivot	Reverse Delta	HPHT	2 faces with box	Pivot	2 hermetic integrated	2 non-hermetic integrated	Basic		
Functional Flows	Generate power							3	3		2	1	4
	Withstand pressure									1			
	Dissipate Heat	4	4	4	4	3	3	3		5	2	1	
		3	3	3	3	2	2	2		4			
Resist shock (X)	2	2	2	2	3	3	3			1	4	5	

Figure 5: FF-DMM matrix (A_{ij}).

The FF-DMM matrix is defined as an $A = a_{ij}$ matrix, where i is the index of defined functions and j is the index of technical solutions.

Next, a product architect or system engineer fills in the PC-DSM matrix during the concept brainstorming session with the design team. The PC-DSM matrix is defined according to system architecture (different modules and technical alternatives) identified in the previous step (see Table 1). PC-DSM matrix is enriched with interface typology information related to the Interface Data Model (see Figure 6). The Interface Data Model consists of the following information: technical performances related to interface, shared design parameters, principle solutions related to this parameter, previous experience and projects that encountered the same design dependency problem, modules connected through the interface, design engineers that are responsible for the design, design experts that have knowledge about principle solutions and tasks that are planned for interface design. These additional data represent the design information required and defined by design engineers in order to incorporate shared design parameters as well as their potential impact with the design process.

		Alternative Technical Solutions									
		Chassis				Box			Connector		
		I	Delta	Pivot	Reverse Delta	HPHT	2 faces with box	Pivot	2 hermetic integrated	2 non-hermetic integrated	
Alternative Technical Solutions	Chassis	I					E, V, {V, S}	V	No assembly		
		Delta					V; S	V, {V, S}			
		Pivot						G			
		Reverse Delta					S	V; S			
	Box	HPHT								F	
		2 faces with box									G
		Pivot								F	
	Conn	2 hermetic integrated									
		2 non-hermetic integrated									

Figure 6: PC-DSM matrix (B_{ij}).

The notation symbols {V, S} means that two types of links are possible at this time for the considered interface (V stands for mechanical interfaces, e.g., screws, and S stands for silicon interfaces). The PC-DSM matrix is defined as $B = b_{ij}$, where j is the total number of technical solutions.

The third matrix used in MDDS is a VoDD matrix. This enriched QFD-like matrix is defined in order to capture the designer's assessment of alternative technical solutions and their potential to attain the required performance criteria. The data concerning possible interfaces between modules is captured in the head of the matrix (see Figure 7). The empty (grey) cells in the top of matrix show the "No assembly" possibility between modules and alternative technical solutions. Performance criteria and their correlations, as expressed by different design departments, are represented by data on the left side of the matrix (see Figure 7). If enhancing one performance criteria will in all likelihood enhance another then the relationship between them is labeled 1. On the contrary, when enhancing one matrix is likely to result in the degradation of another, the correlation is labeled -1. When no correlation exists between performances the relationship is 0. The empty cells in the body of the matrix show that this module is not contributing to achieving that particular technical performance criterion.

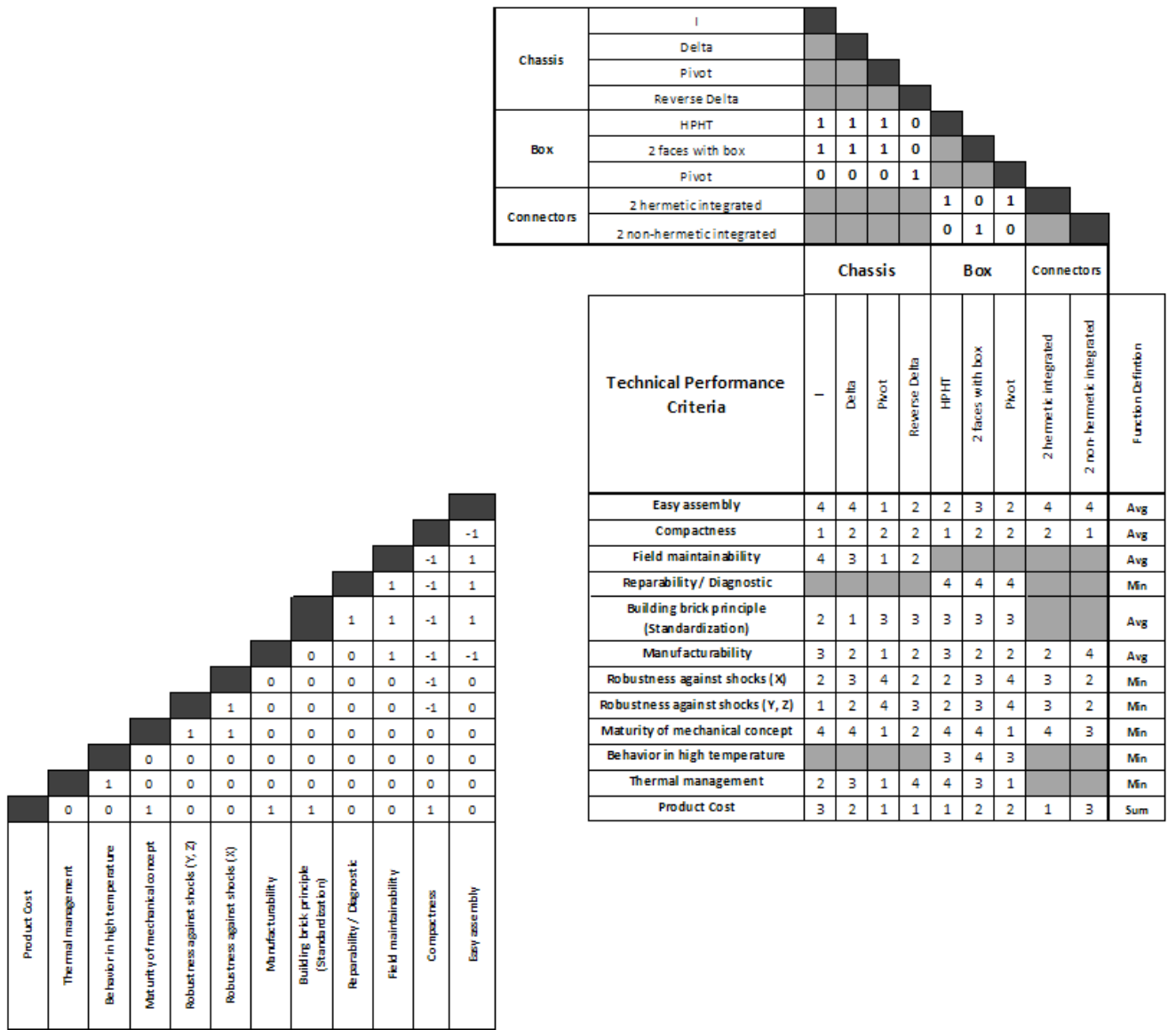


Figure 7: VoDD matrix (V_{kj}).

This matrix is defined as $V = v_{kj}$, where k represents the index of technical performance criteria and j is the index of alternative technical solutions in the design process. For algorithm clarity, we define a vector $R = r_k$ as an expression of performance target functions and $P_a = p_{kk}$ as an expression of performance correlation in the VoDD matrix. The body of the matrix contains data concerning the designers' assessment of the potential performances of different modules. Each evaluation is defined and discussed with different design departments and their experts. For each performance criteria, experts evaluate the potential of each module and technical solution to meet the required technical performances. This evaluation of the level of satisfaction is based on a four-level maturity scale (see Table 2).

Table 2: Maturity scale for the estimation of technical performance satisfaction.

Maturity Scale		
General Guide		Adaptation to “shock acceleration” performance of PEC case study
4	Never done in the world	Over 50G of shock resistance
3	Done by business competitor	Between 30G and 50G
2	Already done in our industry	Between 10G and 30G
1	Without any problem	Below 10G of shock resistance

These three enriched interface representations identify potential couplings within the system which, in turn, are used to define six types of Design Assessment Cards (DACs) to manage and support the design process.

5.2 Six types of DACs to support the two design quality vectors

DACs are matrices derived from the PC-DSM, FF-DMM and VoDD matrices. The objective of these DACs is to provide information to the design team concerning potential difficulties and conflicts that may arise in the design process. They represent special connectivity maps (Yassine et al., 2003) developed to identify potential conflicts, especially in architectural interfaces, and tradeoffs to be made. The aim of this method is to identify potential difficulties in multidisciplinary design. Therefore six types of DACs are proposed to address recurrent engineering problems:

1. **Initial Set:** In multidisciplinary design, in view of the number of possible solutions for each domain, the task of exploring the combinatorial explosion of all possible solutions and their integration into one concept is one of significant complexity. This type of DAC automatically generates all possible concepts with regards to expert knowledge of functions, interfaces and performances.
2. **Functional Satisfaction:** Evaluation of overall system performances is difficult due to the functional flow through components that are determined by different domains. Therefore, this DAC supports designers into estimating system performances in view of functional flows.
3. **Design Department Scale Factors:** In early design stages, the performances are estimated by experts. This type of DAC represents a sensitivity analysis and highlights the impact of one design domain onto overall system performances. Therefore, if one domain is over constrained there is a possibility to detect it and identify possible trade-offs.
4. **Technical Performance Design Compromise:** It is often not possible to satisfy all system requirements. Technical Performance DAC highlights possible trade-offs that need to be made in order to reach some of the required performances.
5. **Design Interaction Objectives:** Physical interfaces define functional flows and contribute to overall system performances. Therefore, this type of DAC

underlines the level of performances that is defined for one interface in view of expert knowledge in order to attain system performances.

6. Collaborative FMECA: In multidisciplinary design, risks are often separately managed for each subsystem or domain. Collaborative FMECA addressed the risks that are in particular related to physical interfaces and that are to be collaboratively managed between two or more domains.

5.2.1 Initial Set DAC

The Initial Set is a matrix defined using an algorithm that is based upon enhanced consistency algorithm defined by Hellebrand and Lindemann (2008a). Using the data gathered in the PC-DSM matrix, the proposed algorithm generates all of the possible architectural combinations (or concepts) of a technical solution by enumerating all possible physical interfaces between alternative technical solutions (see Figure 8). The number of concepts identified in the Initial Set is defined as h .

Initial Set DAC					
	Module Chassis	Interface Chassis-Box	Module Box	Interface Box-Connector	Module Connector
Concept X	I	(E)	HPHT	(F)	2 Hermetic Integrated
Concept X	I	(V)	HPHT	(F)	2 Hermetic Integrated
Concept 7 « Non Hermetic Chassis »	I	{V,S}	HPHT	(F)	2 Hermetic Integrated

Figure 8: Initial Set DAC

The proposed algorithm will create all possible concepts with regard to the data contained in the PD-DSM matrix (see Figure 6). For the module Chassis, the algorithm views alternative solution I as compatible with HPTP Box. There are 3 types of physical interfaces. For the first concept it will chose physical interface E. The HPHT Box is compatible with 2 Hermetic Integration Connector with only one type labelled F. Therefore, the first concept is I Chassis, connected with interface E to HPHT that is has interface F with 2 Hermetic Integrated Connector (see the first line in Figure 8). In the second concept the algorithm will see that the I Chassis as having a possible interface V with HPHT: and HPHT has only one possible physical interface with 2 Hermetic Integrated Connectors (see line 2 in Figure 8). The algorithm will explore the PC-DSM Matrix until all possible concepts taking into account designer's knowledge on possible physical interfaces.

5.2.2 Functional Satisfaction DAC

Functional Satisfaction, estimates the potential of reaching global system technical performances that are related to one function. This is based on expert evaluation of the potential of each alternative technical solution to reach the necessary performance criteria. The objective is to map functions and technical performance criteria onto matrices for each concept in the Initial Set. This DAC is defined as $E_h = e_{h ik}$. In order to obtain this result, both the VoDD matrix ($V = v_{kj}$) and the

binary FF-DMM ($A = a_{ij}$) matrix are duplicated into h matrices and then filtered to eliminate technical solutions that bear no relevance to the given concept in order to represent each concept independently. An excerpt of a filtered matrix for a given concept can be seen in Figure 9.

		Modules						
		Chassis	Box	Connectors	Collar	Substrate	Electronics	Wiring
		Alternative Technical Solutions						
		-	HPHT	2 hermetic integrated	Basic			
Functional Flows	Generate power			3		2	1	4
	Withstand pressure		1	1	1			
	Dissipate Heat	4	3		5	2	1	
		3	2		4			
	Resist shock (X)			3	1		5	2
		2	3			4		

Figure 9: Filtered FF-DMM matrix ($A_h = a_{h_{ij}}$) for a concept from the Initial Set.

The resulting $V_h = v_{h_{kj}}$ matrices represent the correlations between technical performance criteria and technical solutions comprising each concept; the $A_h = a_{h_{ij}}$ matrices correlate functions and technical solutions for each concept. The $V_h = v_{h_{kj}}$ matrices are then transposed to obtain $V_h^T = v_{h_{jk}}$. The $E_h = e_{h_{ik}}$ matrices (Figure 10) are obtained via a matrix product between V_h and A_h which is further normalized by the number of technical solutions involved in the evaluation (Equation 1).

$$e_{h_{ik}} = \frac{\sum_{j=1}^n a_{h_{ij}} \times V_h^T}{\sum_{j=1}^n a_{h_{ij}}} \quad (1)$$

		Technical Performance Criteria											
		Easy assembly	Compactness	Field maintainability	Reparability / Diagnostic	Building brick principle (Standardization)	Manufacturability	Robustness against shocks (X)	Robustness against shocks (Y, Z)	Maturity of mechanical concept	Behaviors in high temperature	Thermal management	Cost
Functional Flows	Satisfy electrical continuity	4	2	0	0	0	2	3	3	4	0	0	1
	Pressure Stand	3	1,5	0	2	1,5	2,5	2,5	2,5	4	1,5	2	2
	Heat dissipation	1	2	1	0	3	1	4	4	1	0	1	1
		1,5	1,5	0,5	2	3	2	3	3	2,5	1,5	2,5	2
		2,5	2	0,5	0	1,5	1,5	3,5	3,5	2,5	0	0,5	2
	2,3	1,7	0,3	1,3	2	2	3	3	3	1	1,7	3	
Shock resist (X)	2,3	1,7	0,3	1,3	2	2	3	3	3	1	1,7	3	

Figure 10: Functional Satisfaction DAC for Concept 7 ($E_{h_{ik}}$).

Furthermore, subtracting E matrices related to different concepts provides a relative comparison between different concepts (see Figure 11).

		Technical Performance Criteria											
		Easy assembly	Compactness	Field maintainability	Reparability / Diagnostic	Building brick principle (Standardization)	Manufacturability	Robustness against shocks (X)	Robustness against shocks (Y, Z)	Maturity of mechanical concept	Behaviors in high temperature	Thermal management	Cost
Functional Flows	Generate power	0	1	0	0	0	-2	1	1	1	0	0	-2
	Withstand pressure	0	-0,5	0	0	0	0,5	-0,5	-0,5	0	-1	1	0
	Dissipate heat	0	0	0	0	0	0	0	0	0	0	0	0
		-0,5	-0,5	0,5	2	0	0,5	-0,5	-0,5	0	1	0,5	0,5
		-0,2	0,3	0,7	-1,3	1	-0,8	0,5	0,5	-0,2	-1,3	-0,3	-1
Resist shock (X)	-0,2	0,2	0,5	4	1,5	-0,5	0	0	1	3	2	-0,3	
-0,3	0	0,7	2,7	1	-0,3	0	0	0,3	1,7	1,2	-0,3		

Figure 11: Comparison of functional satisfaction for two explored concepts. The maximal difference is four points between each concept. Therefore, a positive value of +0.8 shows the advantage of this concept over the second of almost a whole point. If the design requirements highlight the importance of one particular technical performance criteria, this comparison can be used as a decision support tool.

5.2.3 Design Department Scale Factors DAC

The goal of the Design Department Scale Factors DAC is to evaluate the potential influence of design departments on functional satisfaction values for each concept and each module they design. Our objective is thus to map functions and technical performance criteria onto a matrix representing each concept, each design department and each module $E_{hzn} = e_{hzn ik}$. In order to do this, the $V_h = v_{h kj}$ matrix is filtered into n matrices, each representing a single module with its associated alternative technical solutions. V_h only accounts for technical solutions comprising of the concept h ; thus, each associated design department is also represented independently given that the modules are designed by a single design department. Raising the value of technical performance criteria by one, the aim becomes to explore the sensibility analysis of change on one module and therefore the design department. The resulting $V_{hzn} = v_{hzn kj}$ matrices provide correlations between technical performance criteria and functions designed by one department. Therefore this DAC can be obtained by working out the difference between the product of the transposed V_{hzn} matrix with binary matrix A_h , normalized by the number of technical solutions involved in the evaluation, and the truncated E_h matrix for each module (Equation 2).

$$e_{hzn ik} = \frac{\sum_{j=1}^n a_{h ij} \times V_{hzn}^T}{\sum_{j=1}^n a_{h ij}} - E_h \quad (2)$$

Figure 12 shows an example of the E_{hzn} matrix for “Concept 7 – Non-Hermetic Chassis” associated with the “mechanics” design department and their “chassis” module.

		Technical Performance Criteria											
		Easy assembly	Compactness	Field maintainability	Reparability / Diagnostic	Building brick principle (Standardization)	Manufacturability	Robustness against shocks (X)	Robustness against shocks (Y, Z)	Maturity of mechanical concept	Behaviors in high temperature	Thermal management	Cost
Functional Flows	Generate power	0	0	0	0	0	0	0	0	0	0	0	0
	Withstand pressure	0	0	0	0	0	0	0	0	0	0	0	0
	Dissipate heat	1	1	1	0	1	1	1	1	1	0	1	1
		0,5	0,5	0,5	0	0,5	0,5	0,5	0,5	0,5	0	0,5	0,5
		0,5	0,5	0,5	0	0,5	0,5	0,5	0,5	0,5	0	0,5	0,5
	Resist shock (X)	0,3	0,3	0,3	0	0,3	0,3	0,3	0,3	0,3	0	0,3	0,3
0,3		0,3	0,3	0	0,3	0,3	0,3	0,3	0,3	0	0,3	0,3	

Figure 12: PEC example of $E_{7\text{mechanicalchassis}}$ scale factor ($E_{hzn ik}$).

Due to the sensitivity analysis changing the designer assessment of technical performance by one, the entries can be from 0 to 1. This matrix shows the impact of one

design department on to the correlation between functions and technical performance criteria. Therefore if the value is one, the satisfaction of both the function and technical performance criteria depend entirely upon one design department, and if it is 0.3 the impact is shared and minor in comparison to other functions and technical performance criteria.

Since these values vary by concept, they highlight the contribution of each design department to the overall technical performance of the system. The information is useful for constraint tradeoff decisions. If we know that the mechanical design guarantees technical performance, the design team can negotiate constraints in other parts of the system.

5.2.4 Technical Performance Design Compromise DAC

The Technical Performance Design Compromise DAC identifies potential tradeoffs when it comes to satisfying defined performance criteria. As the body of the VoDD matrix represents an evaluation of the degree of potential performance satisfaction for each technical solution, the product of $V = v_{kj}$ and its transposed matrix V^T indicates the degree of correlation between technical performance criteria with regard to different existing technical solutions and their interfaces. Therefore, the Technical Performance Design Compromise DAC (Equation 3) can be calculated by multiplying expert evaluations of the correlations (positive or negative) between performances $P_a = p_{kk}$ and their degree of correlation.

$$f_{h \quad kk} = \sum_{k=1}^z p_{a \quad kk} \times (v_{h \quad kj} \times [v_h^T]_{jk}) \quad (3)$$

One example of this DAC for Concept 7 is provided in Figure 13. The values of this matrix are from -48 to +48. Negative values show relative degree of conflicts between different performance criteria and potential trade-offs to be made.

Positive entries are synonymous for the fact that there is no necessary compromise between technical performance criteria (i.e., simultaneous fulfilment of both). Negative entries indicate the need for a design compromise between technical performance criteria (i.e., impossibility of simultaneous fulfilment). Empty entries indicate that no correlations exist between technical performance criteria.

		Technical Performance Criteria											
		Easy assembly	Compactness	Field maintainability	Reparability / Diagnostic	Building brick principle (Standardization)	Manufacturability	Robustness against shocks (X)	Robustness against shocks (Y, Z)	Maturity of mechanical concept	Behaviors in high temperature	Thermal management	Cost
Technical Performance Criteria	Easy assembly												
	Compactness	-14											
	Field maintainability	4	8										
	Reparability / Diagnostic	0	6	4									
	Building brick principle (Standardization)	12	0	0	0								
	Manufacturability	-18	0	0	0	0							
	Robustness against shocks (X)	0	0	0	0	0	0						
	Robustness against shocks (Y, Z)	0	0	0	0	0	0	0					
	Maturity of mechanical concept	0	0	0	0	0	0	0	0				
	Behaviors in high temperature	0	0	0	0	0	0	0	0	0			
	Thermal management	0	0	0	0	0	0	0	0	0	0		
	Cost	0	5	0	0	6	7	0	0	0	0	0	0

Figure 13: The Technical Performance Design Compromise DAC for Concept 7 in the PEC study ($F_{h\ kk}$).

5.2.5 Design Interaction Objectives DAC

The aim of the Design Interaction Objectives DAC is to establish design interface goals for physical interfaces by mapping technical performance criteria to an interface for each concept. To this end, the $B = b_{ji}$ matrix is used as a basis for interface identification. For each concept h , a VoDD matrix is generated by evaluating the potential satisfaction of technical performances $V_h = v_{h\ kj}$. Based on these evaluations, a target is set for each technical performance with regard to the information given in the vector $R = r_k$ as a part of the initial VoDD matrix. This DAC shows a degree of difficulty that each module has in reaching the performances under the impact of the interface. An excerpt of this matrix is provided in Figure 14 for Concept 7.

Technical Performance Criteria	Reverse Delta	Target for the design of the interface	HPHT
Easy assembly	2	2	2
Compactness	2	1,5	1
Field maintainability	2	0	
Reparability / Diagnostic		0	4
Building brick principle (Standardization)	3	3	3
Manufacturability	2	2,5	3
Robustness against shocks (X)	2	2	2
Robustness against shocks (Y, Z)	3	2	2
Maturity of mechanical concept	2	2	4
Behaviors in high temperature	0	0	3
Thermal management	4	4	4
Product Cost	1	1	1

Figure 14: Design Interaction Objectives DAC for Concept 7 $P_{h_{jj}}$.

The Design Interaction Objectives DAC (Figure 14) establishes collaborative design targets for the chassis-box interface of Concept 7. The $V_{7\text{Chassis-Box}}$ matrix shows the associated technical solutions, an evaluation of their technical performance criteria, and the associated design departments collaborating on the design of the interface. The $P_{7\text{Chassis-Box}}$ matrix represents technical performance targets in the chassis-box design.

5.2.6 Collaborative Failure Mode, Effects and Criticality Analysis (C-FMECA)

The last DAC is a Collaborative Failure Mode, Effects and Criticality Analysis (C-FMECA). Its objective is to manage potential collaborative conflicts properly. FMECA (IEC60812, 2006) is a reliability engineering method describing possible failures of system elements, their causes, effects and criticality. The difference is that C-FMECA is a reliability analysis that is related to physical interfaces in one system. All generated data is gathered in this DAC. An excerpt of the proposed C-FMECA is provided in Figure 15.

Discipline	Physical Connection	Design Parameter	Risk Description	Likelihood	Severity	Original Exposure	Mitigation Plan	Resource	Duration
Packaging	Glue	Surface area	In a glue physical connection, shearing strength and pull-off strength vary a lot following the management of these design parameters: Surface area between technical solution, Roughness of surface, viscosity coefficient and thickness of glue. See design guideline number: 100236239	5	5	25	Look equations and target values extracted from empirical approach on the design guideline number: 100236239	Contact the Houston center and the glue specialist	1 week
Electronics		Viscosity coefficient							
		Thickness							

Figure 15: An excerpt of the Collaborative FMECA .

FMECA was selected because it is easy to use and has been recognized for its widespread use in industry (Birolini, 2007). The C-FMECA database addresses only the collaborative risk, and is therefore focused on interfaces. The shared design parameters that might represent a potential risk of project failure for each of these interfaces are also included in this database. The first column in Figure 15 represents design domains impacted by the physical interface; in this case it is packaging and electronics. The Physical Connection Column refers to the interface type. In the column referring to Design Parameters, the parameters that are critical for the interface design and that are taken from the Interface Data Model. The next four columns (Risk description, Likelihood, Severity and Original exposure) represent expert evaluation of the risk severity. The last three columns address preventive measures to avoid the risk. They represent some of the best practices available in the company or that experts recommend for the given type of risk.

6 Case Study Results and Discussion

To evaluate the proposed method, we conducted a case study on the PEC. PEC is one of the crucial sub-systems of a development project of an innovative oil exploitation tool. Typically, oil exploitation tools consist of tubes that go deep in the subsoil and that are subject to high pressures (up to 300 bars) and high temperatures (up to 200 degrees Celsius). The tubing in the use phase is a mix of water, mud, oil, gas and acid. The power supplies for transmitters including nuclear power supplies, corresponding receivers and their electronic processors and encoding must be integrated into a rather small diameter (5 to 15 centimetres). Therefore, in PEC, the space is a parameter that is significantly constrained. For example, the electronic domain needs to design the electronic card related to passive components and the space is a critical parameter. On the other hand, the mechanical domain needs to design a chassis that will resist previously described conditions, and the preference would be to use a full steel tube. This is only one example of constraints and there are many for each domain. Therefore, it is needed to support the design team in the identification of interfaces and shared parameters so as to structure the communication and the design process.

PEC design takes about 2 year project development time and involves 3 engineering departments. The project team is made of 11 engineers (from design team and suppliers) and development costs are up to 1/5 of the global project budget. Since our aim is to support the projects during the concept generation and design process management phases, in the instance of this case study, it started with functional analysis and concept generation brainstorming sessions. We gathered initial data during a 3-day workshop. We present a sample outline of the workshop structure in Figure 16.

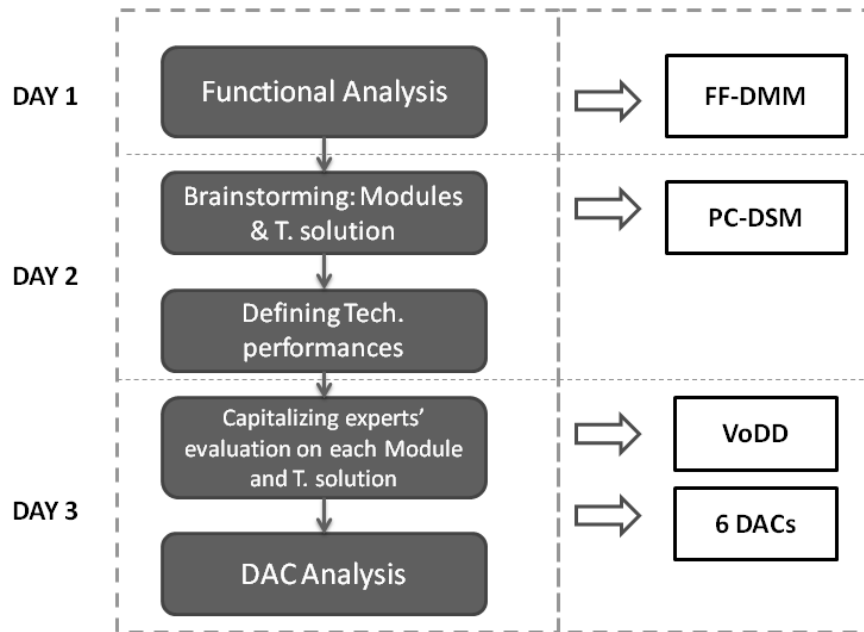


Figure 16: Initial data gathering workshop for the PEC case study.

During these three days the entire MDDS approach is deployed and discussed with engineers. Prior to the workshop itself, the discussion was held with the project manager regarding the initial scope, the objectives of the workshop and the best approach in organizing these three days. If a design team is not familiar with these approaches (as was true in our case), it is important for a mediator to collect the data and discuss the results. In prior discussions in the preparation phase, it is pointed out that the engineers are not familiar with Design Structure Matrices and related design methodologies, like MDDS that is proposed. Therefore, for the workshop, one of the authors plays the role of data collection and result interpretation. The benefits are that the project team is entirely focused on the concept generation and discussion with the benefits of analysis and insights on concept advantages and the necessary trade-offs during the third day. However, because of the 2-year duration of the project, appropriate project follow-up was not organized despite the importance of evaluating the relevance of the solutions screened and selected by MDDS methodology. Initially the design team considered 19 design concepts based upon the product architecture, constituted of modules. Various technical solutions were presented as potential solutions for each module. After collecting the data on these 19 concepts and proposed product architecture, specifically relating to possible physical interfaces and related design parameters. During the first two days, this data is integrated into the 3 matrices (PC-DSM, FF-DMM and VoDD). Using the “Initial Set” DAC, 630 design concepts are identified. Integrating the evaluation of experts on performances, these design concepts are ranked. In order to rank concepts, designers estimate a weight to each performance criterion. For each performance a target function (see VoDD matrix) is defined. The overall rank of a concept represents a weighted sum of all performance criteria estimations for one concept. In the final concept evaluation, six of the 10 best concepts are generated by MDDS. We present two of the nine concepts generated during brainstorming sessions in Figure 17.



Figure 17: The PEC "HPHT" Concept. Left side: Open casing with the substrate and electronic components inside. Right side: The integrated box (I) technical solution for the chassis.

At the end of the 3-day workshops, the design team is asked to give their opinion about the relevance of gathered data and the MDDS method. The feedback is organized as a group feedback from all 11 engineers. They were asked a set of questions concerning the MDDS method: will they use it in the future, what are the advantages they can see in using the MDDS method, what the difficulties are. Engineers are also encouraged to volunteer feedback on the method. The PEC design team states that the approach provided structure and helped them collect data from their previous work. They also find it of particular use to be able to map the parameters of the functional analysis, concept, and technical performance representations. Three of them feel that data collection and exploration alone represent an improvement in the design process. Although six engineers feared that the evaluation step could be time consuming, they are pleasantly surprised to discover that time spent filling out the VoDD is “pretty short for the capitalization made possible”. Exploring and evaluating different concepts brings clarity and produces a general sense of commitment to the project’s success.

The project manager states, “It is clear now that some concepts that we imagined are really better than the others; we did not see that before this workshop.” This project manager was initially skeptical about the approach due to the fact that no industry standard is involved. Moreover, the concept (amongst 19 initially proposed concepts) that was preferred is not ranked highly with regards to system performances.

The proposed MDDS method is based on enriched matrices that are relatively well-known in the industry. The method itself permits identifying trade-offs that are often taken into account too late in the design process. One example is the condenser height that impacts the choice of product architecture on the system level. Initially, shared parameter is in part subjected to constraints imposed by other parts and not taken into account into the global product architecture evaluation. Nonetheless, manipulating the Design Assessment Cards can be difficult. We propose an Excel platform that can help automate the process. The process of filling in matrices influences the results of the approach. In addition, as this method is dependent upon the chosen panel of experts, its relevance is entirely dependent upon a company’s resource management process. In our experience, design team decision adherence is difficult when applying such an approach in an industrial context, as engineers and architects continue to advocate for their preferred solutions. We have no simple solution to propose for tackling this issue; apart from the fact that engineering management imposes a strict and systematic use of the methodology, which can also leave a trace of the design process and the design rationale for architectural concept generation and selection.

7 Discussion and Conclusion

Highly constrained mechatronic systems involve multiple functional flows that share common pathways through the product architecture. These systems operate under severe conditions (high temperature, high pressure, shock and vibration in limited dimensions), and the design process involves engineers from various departments (mechanics, electronics, physics of sensors, etc.); both factors pose challenges to managing system interfaces or dependencies. One major concern is that higher level technical performance may be too constrained by interfaces, thereby interfering with the design process. Moreover, in the design process, different design departments are more or less constrained depending upon the target technical performance criteria and the given concept architecture.

In order to address these issues, we proposed using the MDDS method and to semantically enrich the conventional representation models; PC-DSMs to represent admissible architecture interfaces and dependency configurations, FF-DMMs to link functions and architectures, and VoDD to assess technical performance criteria. We enriched PC-DSM with interface typologies, thereby creating a range of choices early in the design stage and revealing related data about the nature of probable difficulties. In our approach, we proposed six types of DACs to support design space exploration and identify possible trade-offs in order to ensure system feasibility. In this paper, we documented the application of the MDDS method in one product case study. We discussed some benefits, challenges and difficulties to implementing this method in industrial settings. The MDDS allowed us to generate and analyze concepts as well as immediately gather pertinent data during the design process. In this case, the MDDS method helped generate at least 53% of the best-rated concepts. It also helped predict some design challenges and identify necessary tradeoffs. Implementation feedback highlighted how non-experts find it difficult to manipulate these types of methods; therefore enlisting the help of a mediator with detailed knowledge of the process is recommended.

8 References

- IEC Albers, A., Braun, A., Sadowski, E., Wynn, D. C., Wyatt, D. F. & Clarkson, P. J., 2011. System Architecture Modeling in a Software Tool Based on the Contact and Channel Approach (C&C-A). *Journal of Mechanical Design*, 133, 101006-8.
- Andrasfai, B., 1991. *Graph Theory: Flows, Matrices.*, Bristol: Adam Hilger 1991.
- Aoussat, A., Christofol, H. & Le Coq, M., 2000. The New Product Design - A Transverse Approach. *Journal of Engineering Design*, 11, 399-417.
- Birolini, A., 2007. *Reliability Engineering: Theory and Practice*, BerlinHeidelberg: Springer-Verlag.
- Blessing, L. T. M. & Chakrabarti, A., 2009. *DRM, a Design Research Methodology*: Springer.
- Bourseau P., B. K., Dague P., Dormoy J.-L., Evrard J.-M., Guerrin F., Leyval L., Lhomme O., Lucas B., Missier A., Montmain J., Piera N., Rakoto-Ravalontsalama N., Steyer J.P., Tomasena M., Travé-Massuyès L., Vescovi M., Xanthakis S., Yannou B., 1995. Qualitative Reasoning: A Survey of Techniques and Applications. *AI-COM: Artificial Intelligence Communications*, 8, 119-192.

- Bredeweg B., S. P., 2003. Current Topics in Qualitative Reasoning (Editorial Introduction). *AI Magazine*, Winter, 13-16.
- Browning, T. R., 2009. Applying the design structure matrix to system decomposition and integration problems: a review and new directions. *Engineering Management, IEEE Transactions on*, 48, 292-306.
- Brualdi, R. & Ryser, H. J., 1991. *Combinatorial Matrix Theory*: Cambridge University Press.
- Bryant, C., Mcadams, D. A. & Stone, R. B., 2005. A computational technique for concept generation. *ASME International Design Engineering Technical Conference IDETC 2005*. Long Beach, USA.
- Danilovic, M. & Börjesson, H., Managing the Multiproject Environment. 3rd Dependence Structure Matrix (DSM) International Workshop, 2001 Cambridge, MA. Massachusetts Institute of Technology.
- Eppinger, S. D. & Salmien, V., Patterns of Product Development Interactions. International Conference on Engineering Design, August 21-23 2001 Glasgow, UK.
- Gorbea, C., Spielmannleitner, T., Linderman, U. & Fricke, E., 2008. Analysis of Hybrid Vehicle Architectures using Multiple Domain Matrices. *10th International Design Structure Matrix Conference, DSM 08*. Stockholm, Sweden.
- Hauser, J. R. & Clausing, D., 1988. The House of Quality. *Harvard Business Review*, 66, 63-73.
- Hellenbrand, D. & Lindemann, U., 2008a. Using the DSM to support the selection of product concepts. *10th International Design Structure Matrix Conference, DSM 08*. Stockholm, Sweden.
- Hellenbrand, D. & Lindemann, U., Using the DSM to Support the Selection of Product Concepts. 10th International Design Structure Matrix Conference, November 11-12 2008b Stockholm, Sweden.
- IEC60812, 2006. IEC 60812.
- Jarrat, T. 2004. *A Model-Based Approach to Support the Management of Engineering Change*. Cambridge University.
- Kreimeyer, M. F. 2009. *A Structural Measurement System for Engineering Design Processes*. Ph.D., Technical University of Munich.
- Li, S., 2009. Matrix-Based Decomposition Algorithms for Engineering Applications: Survey and Generic Framework. *International Journal of Product Development*, 9, 78-110.
- Li, S., 2010. Extensions of the Two-Phase Method for Decomposition of Matrix-based Design Systems. *Journal of Mechanical Design*, 132.
- Li, S., 2011. A Matrix-based Clustering Approach for the Decomposition of Design Problems. *Research in Engineering Design*, Online first.
- Lindemann, U., Maurer, M. S. & Braun, T., 2009. *Structural Complexity Management*, Berlin Heidelberg: Springer Verlag.
- Malmqvist, J., A Classification of Matrix-Based Methods for Product Modeling. Design conference, 2002 Dubrovnik, Croatia.
- Mane, M., DeLaurentis, D. & Frazho, A., 2011. A Markov Perspective on Development Interdependencies in Networks of Systems. *Journal of Mechanical Design*, 133, 101009-9.
- Maurer, M. S. 2007. *Structural Awareness in Complex Product Design*. Ph.D., Technical University of Munich.

- Moullec, M.-L., Bouissou, M., Jankovic, M. & Bocquet, J.-C., Product Architecture Generation and Exploration using Bayesian Networks. Design 2012, 2012 Cavtat, Croatia.
- Pimmler, T. U. & Eppinger, S. D., Integration Analysis of Product Decompositions. ASME Design Theory and Methodology Conference,, 1994 Minneapolis, USA:.
- Price, C., Autosteve: automated electrical design analysis. ECAI-2000, August 20-25 2000 Berlin, Germany. 721-725.
- Sharman, D. M. & Yassine, A. A., 2004. Characterizing Complex Product Architectures. *Systems Engineering*, 7, 35-60.
- Sinha, R., Paredis, C. J. J. & Khosla, P. K., Interaction Modeling in Systems Design. DETC 2001, Computers in Engineering Conference, September 9-12 2001 Pittsburgh, PA.
- Steward, D., 1962. On an Approach to the Analysis of the Structure of Large Systems of Equations. *SIAM Review*, 5, 321-342.
- Steward, D., 1981. The Design Structure System: A Method for Managing the Design of Complex Systems. *IEEE Transaction on Engineering Management*, 28, 79-83.
- Struss, P. & Fraracci, A., 2011. FMEA of a Braking System - A Kingdom for a Qualitative Valve Model. *25th International Workshop on Qualitative Reasoning*. Barcelona, Spain.
- Struss, P., Price, C., 2003. Model-based systems in the automotive industry. *AI Magazine*, 24, 17-34.
- Suh, N., 1990. *The Principles of Design*: Oxford University Press.
- Suh, N., 2001. *Axiomatic Design: Advances and Applications*, Oxford: Oxford University Press.
- Suh, N. P., 1995. Designing-in of quality through axiomatic design. *Reliability, IEEE Transactions on*, 44, 256-264.
- Wyatt, D., Wynn, D. & Clarkson, J., Synthesis of Product Architecture using a DSM/DMM Based Approach. 10th International Design Structure Matrix Conference, November 11-12 2008 Stockholm, Sweden.
- Wyatt, D., Wynn, D., Jarrett, J. & Clarkson, P., 2012. Supporting product architecture design using computational design synthesis with network structure constraints. *Research in Engineering Design*, 23, 17-52.
- Yassine, A., Whitney, D., Daleiden, S. & Lavine, J., 2003. Connectivity Maps: Modeling and Analysing Relationships in Product Development Processes. *Journal of Engineering Design*, 14, 377-394.
- Yassine, A. A. & Braha, D., 2003. Complex Concurrent Engineering and the Design Structure Matrix Method. *Concurrent Engineering: Research and Applications*, 11, 165-176.
- Ziv-Av, A. & Reich, Y., 2005. SOS - subjective objective system for generating optimal product concepts. *Design Studies*, 26, 509_533.