System Driven Product Development (SDPD) by Means of Development of a Mechatronic Systems in an Industrial Context

Vahid Salehi, Lukas Burseg

An integrated approach for Sytem Driven Product Development (SDPD) by means of development of a mechatronic systems in an industrial context

Vahid Salehi[1], Chris McMahon[2],

[1]Munich University of Applied Sciences, Munich, Germany
Salehi-d@hm.edu
[2]University of Bristol, Bristol, UK
chris.mcmahon@bristol.ac.uk

Abstract: The engineering process is characterized by a long and complex design activity which starts with requirements formulation and the first sketches in the preliminary design phase and extends to the final detailed functional and physical models. Every design phase includes different process steps and tasks which are closely interconnected with each other. The different design stages demand capable Product Lifecycle Management which are able to handle the different kinds of design information. Currently one of the possibilities is the application of Systems Engineering tools and systems like PLM System (Teamcenter) in product development processes. To achieve the full potential of such PLM in Systems Engineering tools especially in view of the complexity of the parts and assemblies in industries such as the aeroplane industry it is important to have a clear understanding of how best to use such systems. Systems Engineering is an interdisciplinary field of engineering that focuses on how to design and manage complex engineering systems over their life cycles. Issues such as reliability, logistics, coordination of different teams (requirements management), evaluation measurements, and other disciplines become more difficult when dealing with large or complex projects. Systems Engineering deals with work-processes, optimization methods and tools in such projects. It overlaps technical and human-centered disciplines such as control engineering, industrial engineering, organizational studies, and project management. Systems Engineering ensures that all likely aspects of a project or system are considered, and integrated into a whole. After a short introduction, this paper, which is based on the results of the accomplished descriptive study and literature survey, presents a generic integrated approach of System Driven Product Development (SDPD) and demonstrates the general requirements of a generic integrated approach during the Engineering Design of Systems. The second section presents a new approach of Systems Engineering, which is based on SDPD and will explain the different phases and sub-phases of the developed approach. By means of designing a Quadrocopter the different phases of the developed generic integrated approach will be demonstrated and presented. Section three will discuss the results of the prescriptive study and address the most important issues. In general this paper presents the prescriptive phase of the design research methodology according to Blessing and Chakrabarti.

1 Introduction

Today the application and development of methods in Systems Engineering design is something natural. There are many definitions of what System Engineering in the field of product development is. For example, according to ANSI/EIA-632-1999 Systems Engineering is "An aggregation of end products and enabling products to achieve a given purpose."[19]. DAU defined Systems Engineering as "an integrated composite of people, products, and processes that provide a capability to satisfy a stated need or objective." According to INCOSE Systems Engineering is an interdisciplinary approach and means to enable the realization of successful systems. It focuses on defining customer needs and required functionality early in the development cycle, documenting requirements, then proceeding with design synthesis and system validation considering the complete problem. Systems Engineering integrates all the disciplines and specialty groups into a team effort forming a structured development process that proceeds from concept to production to operation. Systems Engineering considers both the business and the technical needs of all customers with the goal of providing a quality product that meets the user needs. IEEE Std 1220-1998 said Systems Engineering is "A set or arrangement of elements and processes that are related and whose behavior satisfies customer/operational needs and provides for life cycle sustainment of the products."[19]. ISO/IEC 15288:2008: "A combination of interacting elements organized to achieve one or more stated purposes."[19]. In general, it can be said that all this definitions consider Systems Engineering from different aspect and perspective, which are valid in their field of application. It can be said that Systems Engineering consider aspects like requirement definitions, functional-, logical-, System- (Subsystems-) and components-Level. That means for the successful accomplishment of Systems Engineering needs a systemically and methodologically connection of the different disciplines. One of the important aspect of Systems Engineering is the methodological process of engineering things. The history of design method development is very long and therefore there are many relevant books, theses and research papers in this area. Some of these conventional and general design methods are described by Roth, 1979 [1]; Ehrlenspiel, 1974 [2]; Hubka, 1976 [3]; Rodenacker, 1976 [4]; Pahl and Beitz, 1977 [5]; Koller, 1985 [6]; VDI 2222 [7] and Suh, 1985 [8]. The application and development of the generic integrated approach presented in this paper is defined for Systems Engineering process in an integrated product development environment. According to Ehrlenspiel integrated product development is a “holistic approach to overcome the problems that arise in product development due to the division of manpower” [2]. Furthermore, products become more complex and because of that the development can not be accomplished by a single designer. Integrated product development is an approach that includes different methods of problem solving, organizational methods of optimizing interpersonal processes and technical methods for the direct improvement of products [2]. The current situation of integrated product development is based on a stronger interaction and integration of different design activities, groups and departments. These new boundaries and approaches aim to provide stronger support for the individuals (designers and other participants) in the design process during their working process and tasks. Pahl and Beitz [5] defined the term “method” in engineering design as analyzing the structure of technical systems and their relationships with the environment. Furthermore, the aims of methods are to drive principles for the development of these systems from the system elements and their relationships [5]. They also used the term “methodology” and defined it as a “concrete course of action for the design of technical systems that derives its knowledge from design science and cognitive psychology and from practical experience in different domains”. This includes the planning of actions to connect working steps and design phases according to content and organization. Furthermore, methods are prescriptive, goal and solution oriented. Methods in the product development process present a kind of guide and advice to reduce the complexity of something [5]. By means of methods, complex problems are divided in smaller sub-problems which can be solved more easily. In addition, methods help achieve better cooperation and communication between the participants in the product development process. The handling and administration of design information and knowledge can also be supported through method application. They also promote the comprehensible documentation of design information in the development process [5]. Especially during the design process of mechatronic and Systems Engineering there is a huge need of methodologies to handle all the information during the development process of Systems Engineering. That means that specially related to Systems Engineering there are methods necessary to be able to enhance all the information, which are from different discipline of the product development process like, mechanical, electrical, and information technology area. In this case without a systemically product development it will be quite difficult to handle all the information. The next section will present the developed system driven product development approach

2 Development of a generic integrated System driven product development (SDPD) approach

The following paper will explain the System Driven Product Development (SDPD) approach based on four different main phases which comprise the top level of the developed approach. These phases are: 1) Business Level 2) Functional Level, 3) Technical Level and 4) Component Level. The example, which is chosen to explain the different stages of the SDPD, is a Quadrocopter based on the SDPD approach at the University of Applied Sciences. Figure 1 shows the different phases of SPDP.

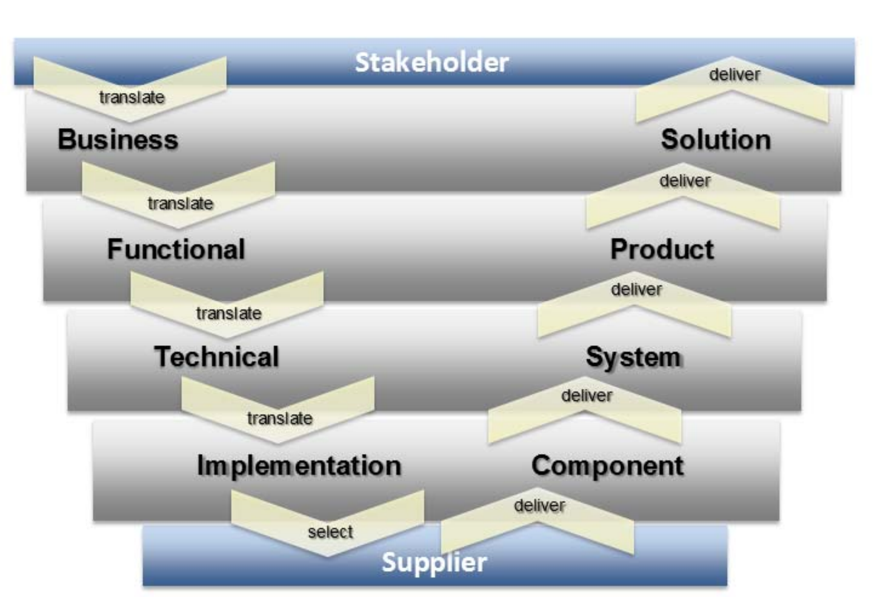


Fig 1: The different phases of SDPS

2.1 Business Level:

The Business Level of the product development contains all the costumers need and wishes related to the product and its requirements. That means that the challenge during the first stages of the product development is to understand and to translate the costumers need into technical requirements. In product development and process optimization, a requirement is a singular documented physical and functional need that a particular design, product or process must be able to perform. It is most commonly used in a formal sense in Systems Engineering, software engineering, or enterprise engineering. It is a statement that identifies a necessary attribute, capability, characteristic, or quality of a system for it to have value and utility to a customer, organization, internal user, or other stakeholder. A specification (often abbreviated as spec) may refer to an explicit set of requirements to be satisfied by a material, design, product, or service.

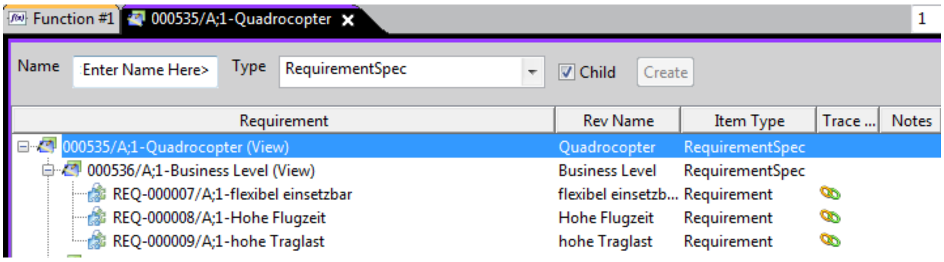
In the classical engineering approach, sets of requirements are used as inputs into the design stages of product development. Requirements are also an important input into the verification process, since tests should trace back to specific requirements. Requirements show what elements and functions are necessary for the particular project. This is reflected in the waterfall model of the software life-cycle. However, when iterative methods of software development or agile methods are used, the system requirements are incrementally developed in parallel with design and implementation. Requirements are typically classified into types produced at different stages in a development progression, with the taxonomy depending on the overall model being used. Furthermore, there are different kinds of requirements, which are:

- Architectural requirements: Architectural requirements explain what has to be done by identifying the necessary systems structure and systems behavior, i.e., systems architecture of a system.
- Business requirements: High-level statements of the goals, objectives, or needs of an organization. They usually describe opportunities that an organization wants to realise or problems that they want to solve. Often stated in a business case.
- User (stakeholder) requirements: Mid-level statements of the needs of a particular stakeholder or group of stakeholders. They usually describe how someone wants to interact with the intended solution. Often acting as a mid-point between the high-level business requirements and more detailed solution requirements.
- Functional (solution) requirements: Usually detailed statements of capabilities, behaviour, and information that the solution will need. Examples include formatting text, calculating a number, modulating a signal. They are also known as capabilities.
- Quality-of-service (non-functional) requirements: Usually detailed statements of the conditions under which the solution must remain effective, qualities that the

solution must have, or constraints within which it must operate.[4] Examples include: reliability, testability, maintainability, availability.

- Implementation (transition) requirements: Usually detailed statements of capabilities or behaviour required only to enable transition from the current state of the enterprise to the desired future state, but that will thereafter no longer be required. Examples include: recruitment, role changes, education, migration of data from one system to another.

The Quadrocopter is used here as a very good mechatronic example to define and demonstrate the different steps of SPDP approach. A Quadrocopter is an aero-vehicle and has four vertically acting rotors or propellers that give their power regularly downwards so that the aero-vehicle kept stable in the air. It is a mechatronic product and includes not only mechanical but also electronic, and software technical components. Now the SPDP approach The „Requirement Management“-module in the PLM System offers (Teamcenter) excellent opportunities to sync and manage the relevant requirements in Office Word or Doors. A requirement specification in Teamcenter Systems Engineering is a container for request and paragraph properties. These objects can be structured within a requirements specification in a hierarchy of parent and child relationships. Furthermore the PLM System Teamcenter enables traceability which contains defining relationships. With traceability Teamcenter provides the opportunity to follow the whole lifecycle of a request in forward and backward. Related to the quadrocopter the requirements can be divided in different level of the systems. In the overall view of the system the requirements are a long flight time, flexible maneuver and a high carry weight. In the next section the functional level will be explained. Figure 2 shows a small example of the requirements of the Quadrocopter on the top level.



Name: Enter Name Here> Type: RequirementSpec ☑ Child

Requirement	Rev Name	Item Type	Trace ...	Notes
000535/A;1-Quadrocopter (View)	Quadrocopter	RequirementSpec		
000536/A;1-Business Level (View)	Business Level	RequirementSpec		
REQ-000007/A;1-flexibel einsetzbar	flexibel einsetzb...	Requirement	🔗	
REQ-000008/A;1-Hohe Flugzeit	Hohe Flugzeit	Requirement	🔗	
REQ-000009/A;1-hohe Traglast	hohe Traglast	Requirement	🔗	

Fig. 2: Extract at the requirements definition

2.2 Functional Level:

The functional level is a method for the understanding of the total product and to define defines the structure and behavior of a system. A system is structurally represented by the functional or structural relationships between the individual components or sub-functions with regard to the technical requirements. With the creation of a functional system architecture, it is possible to identify the different functions in the entire system. For the outdoor use of Quadrocopter it ist necessary to defined the functions consisting of sinking, rising, landing, yaw, pitch or roll on the basis of defined customer requirements. Moreover, the Quadrocopter construction should be very solid for the outdoor use and the protection of the Quadrocopter with the so called bumpers. Furthermore, a high flight time should be ensured. For this purpose, relevant parameters such as weight, type of battery, engine / propeller combination and the losses occurring due to wind or driving style are taken into account in the system development. The functions of the Quadrocopter can be connected in the Visio diagram via ports. Ports are used to connect the individual functions and to nest together. Between these links information or objects are transported. Ports can be displayed as input, output, input and output as well as undirected. The following figure shows the ports are undirected symbolizes what can be seen on the rectangular endpoints on Connector. After the request structure is linked to the function structure, they can be represented visually in PLM System (Teamcenter) with main and sub-functions in the form of functional blocks. Figure 3 shows the functional level based on the Systems Visio.

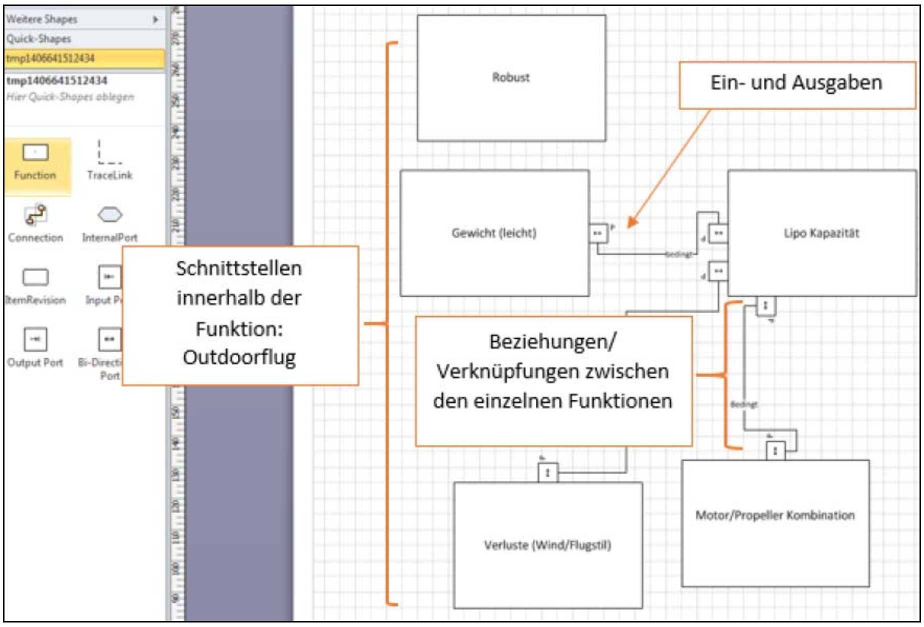


Fig 3: Create a function set in Siemens PLM System (Teamcenter)

In order to assign the defined in Figure 3 functional structure subsequent or preceding structures are used in PLM System (Teamcenter) trace links. A Trace Link enables the traceability of customer requirements; define the functional and logical view of the sample submitted in outdoor flight. Furthermore, the trace link is used to display other functions in the same structure or function in different views. These functions must be linked together, so that the functionality of the system can be ensured through their interaction.

2.3 Technical-Level

Once the functional structure is illustrated, the technical system architecture is created with consideration to the functional structure. The technical system architecture of SDPD approach describes the logical behavior of a system. To simulate the logic models and modeling, the simulation tool Matlab & Simulink in 2013 used for the application example. It enables the design engineers to consider the dynamic behavior of the components. The system modeling and simulation can be ultimately verifies that the modeled system meets the requirements defined in the business level or not. Unexpected interactions between the individual components or subsystems can be detected at an early stage validation. In the simulation of the Quadrocopter control technology plays a very important role. It helps to analyze and to ensure the stability during the flight as well as to link the sensor data and the motor outputs to each other. For this, the PID controller is used for the application example. In addition, can be guarantees that by means of the integration module for Mathlab and Simulink in PLM System (Teamcenter) significant issues of the work between the systems works. The interactions in Mathlab / Simulink can be performed directly from within PLM System (Teamcenter). Thus, the continuity between the systems can be represented very well. Figure 4 shows the mathlab and simulink model of the quadrocopter.

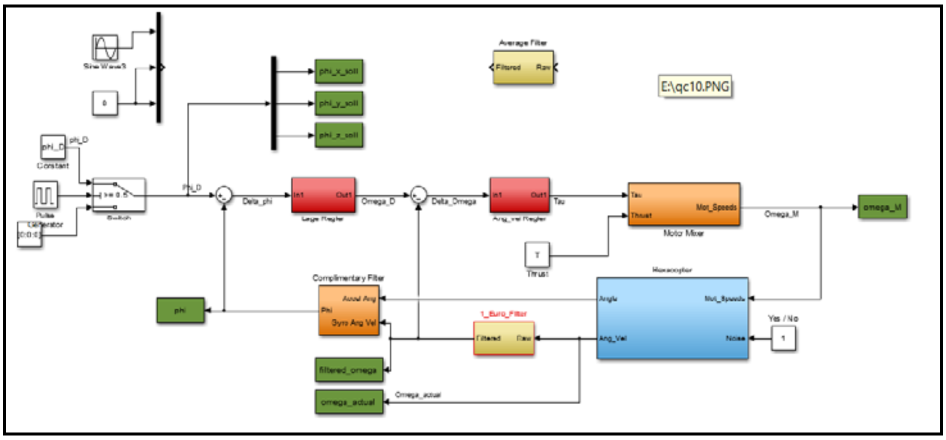


Fig. 4: Matlab & Simulink model of the Quadrocopter

2.4 Component-Level

The final step of the SPDP approach of Siemens describes the Component Level. In this context, physical design and the 3D geometric model of the product to be developed should be designed as an overall design. First, the physical properties of the system are shown and described in a hierarchical structure in Systems Engineering. The figure below illustrates the structure of the component levels. After the behavioural description of the individual components has been extensively described in the previous section, so that a large part of the definition of the technical system architecture is covered. In conclusion to this chapter, the physical system architecture is now still displayed, which describes the real view of the overall system. Since the Quadrocopter not only contains mechanical components, but also consists of electrical and mechatronic components, design engineers need a 3D modeling solution that takes into account the different views of 3D models of this whole system.

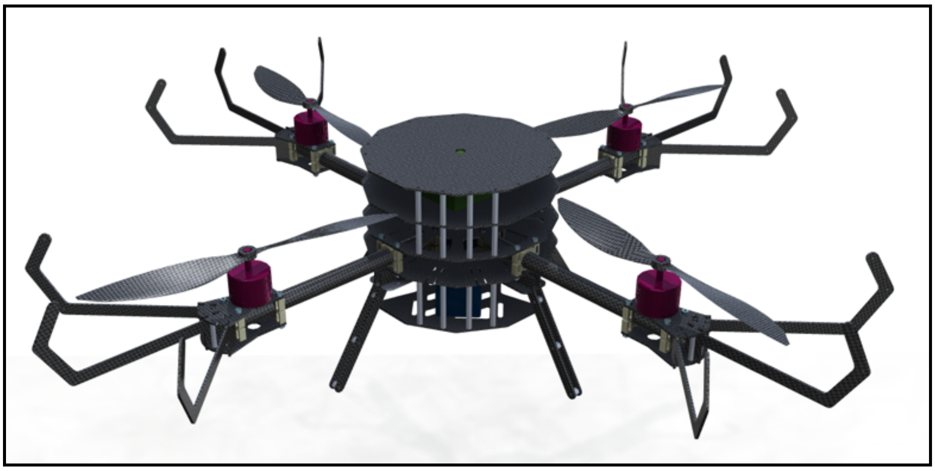


Fig 5: The 3D-View of the Quadrocopter

3. Benefits for the team during the use of PLM System (Teamcenter) Systems Engineering

When applying the Systems Engineering Module and SPDP-approach for the development team of the Quadrocopter is was possible to achieve all the goals during system development. The mechatronic system Quadrocopter was in this way completely and consistently mapped and integrated in PLM System (Teamcenter) with all processes. This is one of the main strengths of the system engineering approach with PLM System (Teamcenter). The following additional benefits were identified from the perspective of the product development team:

- The Systems Engineering Module provides a very good integration of the customer requests or demands in order to be transparent for each stage of the product development requirements.
- There is a classification of requirements and whose relations (Trace Links) possible. Thus, the complex relationships of a system can be represented comprehensible.
- The synchronous communication between Systems Engineering Module with Office programs.
- Based on the requirements solution blocks are represented consistently and transparently.
- The decision-making processes can be displayed and traceable throughout.
- The product development process can be integrated and presented detailed.
- Adjustments and changes may be permitted pursued and implemented.

4. Summary

The complexity of the development as part of the product-development-process has increased steadily with this interdisciplinary character. In order to control these complexity and to be able to develop efficiently mechatronic systems during product development process, a "system Driven Product Development" (SPDP) is suggested. "System Driven Product Developement" integrates all engineering disciplines referred in a single process to design a solution. This methodological approach followed mechatronic systems on requirements, capabilities, technical components to the physical description of system. The SPDP approach is developed from abstract to specifically for a particular procedure. The initial phase respectively the business level of the product development process plays a important role in subsequent development steps. For this, the module offers Requirement Manager excellent benefits in the definition of requirements and customer wishes. A complete and early request definition that supports customer needs extensive supports the development of high-quality mechatronic system. By integrating Visio and Mathlab / Simulink in PLM System (Teamcenter) it is possible to produce functional and logical point of view, so that the decision paths can be easily traced during the design process and plausibly. In conclusion, that was very successfully displayed in the project of the Quadrocopter using Systems Engineering in PLM (Teamcenter).

References

[1] Roth K. Konstruieren mit Konstruktionskatalogen, Springer Verlag, Berlin, 1994.

[2] Ehrenspiel K. Integrierte Produktentwicklung - Methoden für Prozessorganisation, Produkterstellung und Konstruktion. Carl Hanser Verlag, München, 1995.

[3] Hubka V. Theorie der Konstruktionsprozesse - Analyse der Konstuktionstätigkeit, Springer Verlag, Berlin, 1976.

[4] Rodenacker W. Methodisches Konstruieren – Grundlage, Methoden, Praktische Beispiele, Springer Verlag, Berlin, 1991.

[5] Pahl and, G., Beitz. Grundlagen erfolgreicher Produktenwicklung. Methoden und Anwendung.
Berlin: Springer, 2003, pp. 30-45

[6] Koller R. Konstruktionslehre für den Maschinenbau, Grundlage zur neu und Weiterentwicklung der technischen Produkte, Springer Verlag, 1994.

[7] VDI Richtlinie 2222, Verein Deutscher Ingenieure, Mehodik zum Entwickeln und Konstruieren, technischer Systeme und Produkte. Düsseldorf, VDI-Verlag, 1993. pp. 10-15

[8] Suh N. The Principles of Design (Oxford Series on Advanced Manufacturing), Oxford, 1990.

[9] Shah J. Assembly Modeling as an Extension of Design by Features, special Issue on Advances in CAD, Research in Engineering Design, 1993, pp. 218-237.

[10] Blessing L. DRM: A Design Research Methodology, in: Proceedings of the Conférence Internationale Les Sciences de la Conception, INSA-Lyon, 2004, pp 1-8.

[11] Author 1, Author 2. Action Research into the use of parametric associative CAD systems in an industrial context. In International Conference on Engineering Design, ICED'09, Stanford, August 2009.

[12] VDI Richtlinie 2209, Verein Deutscher Ingenieure. 3D Produktmodellierung, Düsseldorf, VDI-Verlag, 2006, pp. 44.

[13] Homepage of the Kolbenschmidt AG: http://www.ms-motor-service.com/content2.asp?area=hauptmenue&site=produkte&cls=02&pcat=8&pID=24&upID=1

[14] Hernandez C. Design Procedures: A Computational Framework for Parametric Design and Complex Shapes in Architecture, 1993, pp. 13-70.

[15] Vajna S. Einsatz der Parametrik in der Produktentwicklung, VDI-Z spezial C-Techniken, 1998, pp. 40-60

[16] Bartolomei J. et al. Analysis and application of design structure matrix, Domain mapping matrix, and engineering system matrix frameworks, Massachusetts Institute of Technology Engineering Systems Division, 2007, pp. 1-15.

[17] Parametric Technology Cooperation (PTC), Complex Assemblies Made Simple—By Design

[18] Pro/ENGINEER® Tools Deliver Faster Design, Improved Quality, 2005, pp 2.

Equations (Equation Number).