INTERNATIONAL DESIGN CONFERENCE - DESIGN 2008
Dubrovnik - Croatia, May 19 - 22, 2008.

TRANSPARENT 3D VISUALIZATION OF MECHATRONIC SYSTEM STRUCTURES

H. Diehl, D. Hellenbrand and U. Lindemann

Keywords: mechatronic, process design, complexity management

1. Motivation

Increasing number of comfort features as well as rising requirement standards regarding safety and environmental issues require a broad application of mechatronic systems in present-day cars. Mechatronic systems generate additional functionality by integrating electronic and software elements to a mechanical structure. This integration of three different domains causes a significantly higher complexity level for the complete system and the belonging processes [Bullinger 2003].
In order to handle a robust development of complex mechatronic systems, improved system understanding as well as an effective and on schedule integration of the three domains is essential especially regarding measures of function validation.
The following figure (Figure 1) shows the main interdependencies of a system that should be made transparent to gain a better understanding of the mechatronic system. Domains of interest are functions, system elements and persons. Concerning the future costumer, the product development process can be seen as the realization of functional requirements by means of the development of systemelements. To enable an efficient development of a product the engineers should have a transparent visualization of the interdependencies between and among functions and systemelements as their realizing counterpart. In this context the term “system element” is used as a neutral synonym for mechatronic product elements like software, electronic or mechanic parts.
The visualization of the interdisciplinary interdependencies between system elements is especially important when changes have to be made. Closely related is the question about responsibilities for functions and systemelements. So a typical question is “Who do I have to talk to when I have to make a change?”
The shown questions in Figure 1 are closely related to the design of a robust development process. The main questions in this context concern the seamless integration of the discipline specific development activities.

- Which subfunctions have to be developed to realize a customer function?
- Which discipline (Software, Electrics, Mechanics) is used to implement which function?
- Which functions are tested when and what degree of maturity is necessary to do so?

These questions may seem trivial, but when the high time and cost pressure as well as the high amount of call backs because of low quality is taken into account the importance becomes obvious.
To cope better with product complexity and the triggered process complexity an approach for a function orientated modeling of the development process of mechatronic products has been developed. The project has performed in the scope of the CAR@TUM cooperation between the BMW-Group and the TU-München. The approach integrates the structure of the system with the costumer function on the highest level of hierarchy. A main goal of this approach is to make a serial development process more transparent. Further on visualization techniques which make the dependencies of the before

mentioned domains transparent have been developed and will be further optimized. These visualization techniques are designed to support the engineer in his daily work within the development process (see Figure 1). In the following the vision of a function orientated process model is introduced. Afterwards the necessary steps to model the interdependencies of the system are explained more detailed. This includes the introduction of an approach for a 3D visualization of the interdependencies. The paper closes with an outlook about how the systemstructure can be used to structure the process.

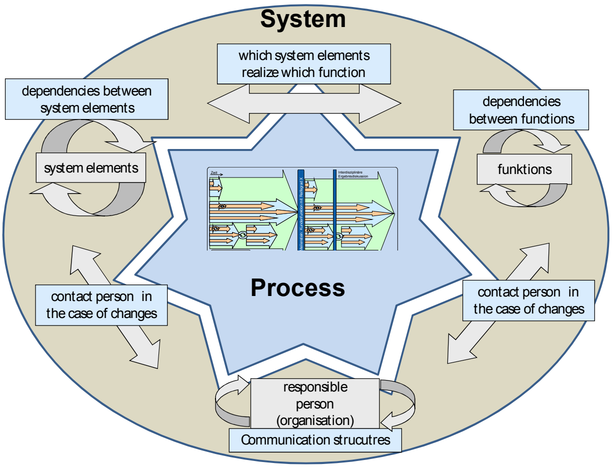

Figure 1. Importance of better system understanding

2. Vision of a function driven process model

Regarding the steps of function validation there is the necessity of an effective and on schedule integration of the three disciplines' deliverables, even in early stages of product development. Therefore discipline-specific development processes will have to be more effectively interlinked. Typical ways of thinking in electronic/software engineering and mechanical engineering – e.g. function driven vs. component driven understanding of mechatronic systems – further complicate this issue.

Process modeling techniques currently applied in industry do not enough enhance overall system understanding and generate too little awareness for the importance of discipline-integrating milestones. This is due to a distinct decoupling of the representations of the development and production processes on the one hand and the system under development on the other.

The simplified chart depicted below demonstrates the main idea behind functional-driven process modeling.

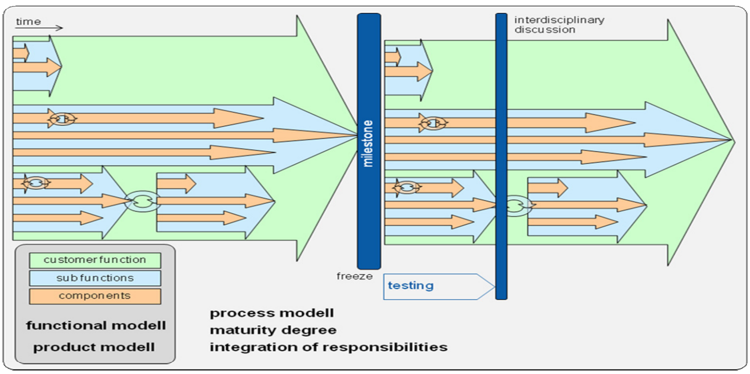

Figure 2. Function driven process modeling

The abstract process chart contains different levels of hierarchy. The top level represents functions demanded by the customer (big arrows), which are realized by various sub functions and their interplay (mid sized arrows). On bottom level, the single components of the mechatronic system (small arrows) can be found, which are essential for realizing the sub functions. Those component system elements accomplish one or several bottom level functions (not depicted in chart) each. Exemplary bottom level functions are “detect force” or “generate actuating force”.
Arrows in the chart represent development phases on the presented levels of hierarchy, concluded by milestone deadlines. Completion of customer function milestones (big arrows) requires all development efforts on lower hierarchy levels to be available in according stages of development. Essential milestones on customer function level are typically defined by vehicle tests integrating domain specific components in serial development. Subsequent to those domain integrated testing activities, cross-disciplinary and prompt discussion of obtained results is of fundamental importance.
Such discussions generate or increase common problem and system understanding within the development team and across domain borders and keeps communication channels short. Long communication channels are a main driver of system failures later on in the field. Additional to those inter-domain testing activities on customer function level, domain specific testing is still existent on lower levels of hierarchy, and is represented as circle arrows in the chart.

3. Procedure for modeling a function orientated development process

The shown vision of process modeling requires three main steps:

- Generation of an integrated product model: linking of functions with systemelements and definition of responsibilities
- Rough process planning: definition of milestones by the use of the system structure
- Detailed process planning: definition of work packages needed to achieve milestones

3.1 Generation of an integrated product model

Figure 3 shows the main steps for building an integrated product model. In the following these steps are explained in more detail.

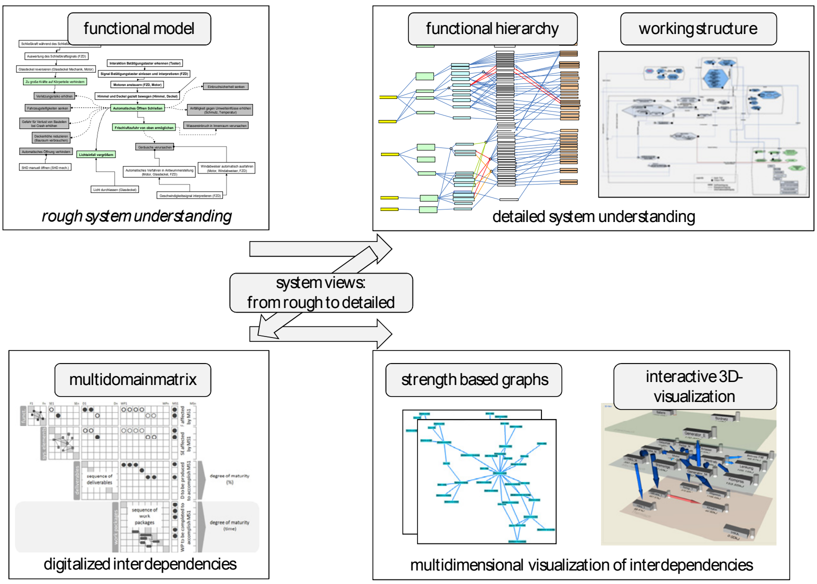

Figure 3. Integrated product model

As already mentioned the creation of an integrated product model by means of a number of sub models is necessary for the modeling of a function oriented process as well as for the generation of an interdisciplinary system understanding. To get a first impression of the system structure a rough functional model is created. After that the definition of the linkage between systemelements and functions by means of a functional hierarchy is of special importance. In our case the functional hierarchy contains the following levels of abstraction:

- cluster of functions
- modules of functions
- customer functions
- sub functions
- elementary functions

On the highest level of abstraction the product has to be separated into functional clusters e.g. „comfort access" or "ergonomic ingress and egress" when looking at automotive industry. A functional cluster should be defined as a block of functions which has strong marketing character. The clusters are subdivided into modules which represent the main use cases. An example for the cluster "ergonomic ingress and egress" are the modules "door opening" and "door closing".

Every module is subdivided into customer functions, sub functions and elementary functions. Customer functions can be interpreted as subuscases. This becomes clear when looking at the subdivision of the module "door opening", which can be divided into: "full automatic opening", "partly manual opening" and "manual opening".

To realize these customer functions technical sub functions are needed. A clear definition of these sub functions is not possible because their formulation strongly depends on the individual context. Sub functions are typically realized by the interaction of a number of systemelements. On the level of elementary functions these systemelements can be well defined. One elementary function should not have more than one belonging system element but one systemelement can realize more than one elementary function. If this is not the case the formulation of the sub functions and the belonging elementary functions have to be checked and adapted. Therefore the process of building this functional hierarchy is an iterative process. It is very important to carry this task out in an interdisciplinary team with the main representatives. All participants should be able to identify themselves with the generated model. Only that way a collective understanding of the system can be achieved. It is not possible to derive all functions from the customer view. For that reason overall functions have to be defined. These are functions like "serve energy" which are the basis for the right behavior of the system.

Besides the functional hierarchy a model is needed which reflects the interdependencies between the system elements. For that reason a working structure is generated. In our case a modeling technique for mechatronic systems developed by Kallmeyer [Kallmeyer 1998] (figurative representation in Figure 3 top right) has been used.

Following the creation of the functional hierarchy and the working structure the dependencies contained in these models have to be transferred in a computer processable form. Therefore a multidomainmatrix approach is used. This multidomain approach has been developed by Maik Maurer within his PHD Thesis "Structural Awareness in Complex Product Design". [Maurer 2007]

A Multidomainmatrix (MDM) serves to systematically link different domains of interest with each other. Domains of interest in this context are functions, systemelements and persons. (In a second step these will be linked with milestones and working packages.) A MDM consists of Design Structure Matrices (DSMs) [Steward 1981] and Domain Mapping matrices (DMMs) [Danilovic 2004]. DSMs show the dependencies within one domain of interest. DMMs define the dependencies between two domains of interest e.g. systemelements and functions. The MDM allows automatic calculation of DSMs out of one DMM and another DSM. For example it is possible to derive the dependencies within in the domain "function" from the combination of the systemelements-DSM and the DMM which defines the dependencies between systemelements and functions (Which systemelement realizes which function).

DSM-based dependencies can be visualized by means of strength based graphs (see Figure 3). [Maurer 2007] Thereto and for the building of the matrices the software tool LOOMEO by the TESEON GmbH [TESEON 2008] has been used. At the moment it is not possible to visualize cross domain dependencies with LOOMEO. To increase transparency in our view it is essential to give an intuitive clearly arranged representation of these cross domain interdependencies. That is why an approach for the 3D-Vizualization of interdomain and intradomain dependencies has been developed.

3.2. Process planning by the use of the system structure

The generated model of the system structure can be used to structure the corresponding development process. In this approach we focus on the serial development process within the automotive industry. Therefore milestones represent points in time where functional requirements are being tested. The structuring of the development process can be subdivided into a rough planning and a detailed planning.

Within the first part the test based milestones are defined. Therefore the functional structure as well as the structure of the system elements are used. First it has to be defined which functions are being tested in which test. Therefore the functional structure can be used to identify logic contents of these tests. For the reason that functions are realized by system elements it has to be defined which system elements are needed in which degree of maturity to carry out the test. The system elements can be derived from the belonging DMM. The necessary degree of maturity has to be defined by the planners of the tests. In this context degree of maturity means in which form a system element is needed:

- Simulation model
- Rapid Prototyping part
- Virtual Reality
- etc.

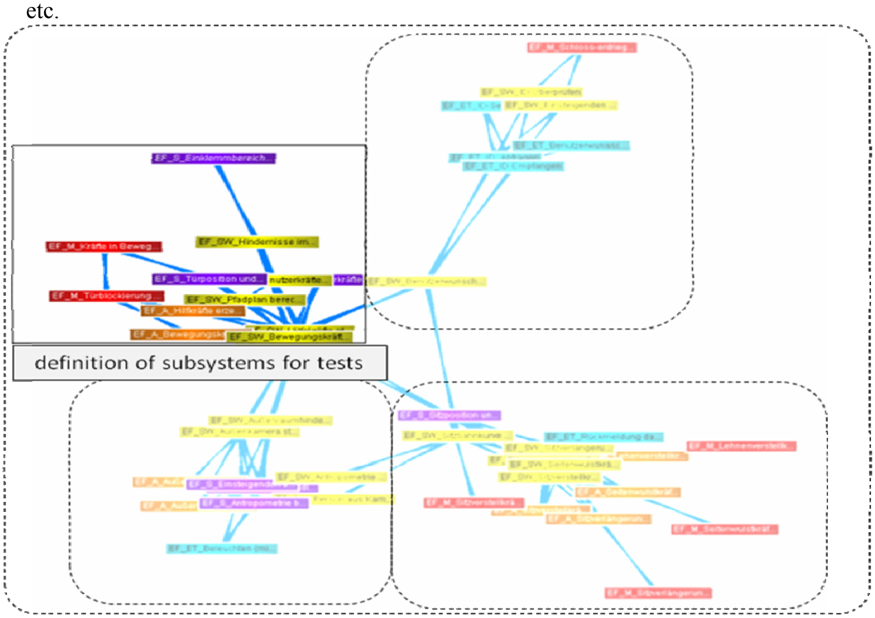

Figure 4. Definition of functional subsystems for tests based on DSMs and strength based graphs

Within the second step of detailed planning working packages which are needed to develop the system elements with the necessary degree of maturity have to be defined and assigned. This can also be done by using the already mentioned MDM-approach [Braun et. al. 2007].

4. Interactive 3D-visulization of system structures

The main motivation for the development of an interactive 3D visualization is the generation of a transparent view on the dependencies between different domains of interest. In our case these are

functions, system elements and responsible persons. Strength based graphs are a good technique to visualize structural characteristics within one domain but are not capable to visualize inter domain dependencies.
In our case we want to provide mechatronic engineers within the operational business with the following information:

- Which system elements are directly linked with system elements I am responsible for?
- Which functions are directly linked with functions I am responsible for?
- Which systemelements serve to realize which function?
- Who are the responsible persons concerning certain system elements or functions?

Further on it should be possible to visualize element related key figures in a way that is easily to interpret. These key figures can be structure based like active sum or passive sum or process related like the degree of maturity or costs.
To meet these requirements a scenegraph based tool has been developed. The tool is called 3D-MDM and uses the open source scenegraph library Openscenegraph. 3D-MDM is linked with LOOMEO via a XML interface. 3D-MDM receives from LOOMEO the following data:

- X/Y position of the elements in the strength based graph view
- information about which elements are linked with each other
- additional structure based key figures

Within 3D-MDM this data is used to place the elements on different semitransparent layers. Every domain of interest is represented by a single layer. The elements themselves are visualized in the form of boxes with variable height. This enables the visualization of key figures. It is possible to visualize crossdomain dependencies as well as dependencies within a single domain. In Figure 5 the top layer represents functions and the lower layer system elements. Responsibilities can be integrated by a third layer. The elements on this layer can then be linked to system elements and functions.
It is possible to interactively navigate through the structure. By clicking an element only the neighboring elements are visualized. The area of responsibility of a person can be simply highlighted by means of this function. Another interactive function that is served by 3D-MDM is the deactivation and activation of all elements of a layer and corresponding linkages by clicking on that layer.
Because the xml-file allows an easy definition of layers and belonging elements it is possible to visualize different aspects of a system. Each relevant aspect can be represented by a single layer. Examples are:

- levels represent different levels of hierarchy
- levels make it possible to differ between elements of a system and of its surrounding
- levels can represent different types of elements: electric components, mechanic components, software components

Further possibilities for visualizing data are served by a head up display on the right side and tons placed next to each elements box. The head up display can be used to interactively show element related data:

- person in charge
- related milestone
- related working packages
- neighboring elements
- etc.

The tons can be used to graphically visualize a second key figure like ABC-analysis or numbers of variants.
The presented functions of the 3D-MDM tool make it very suitable for the use within the development process of mechatronic systems. An integrated product model which makes interdisciplinary dependencies visible and links functions with system elements can be built and used for improved interdisciplinary system understanding.

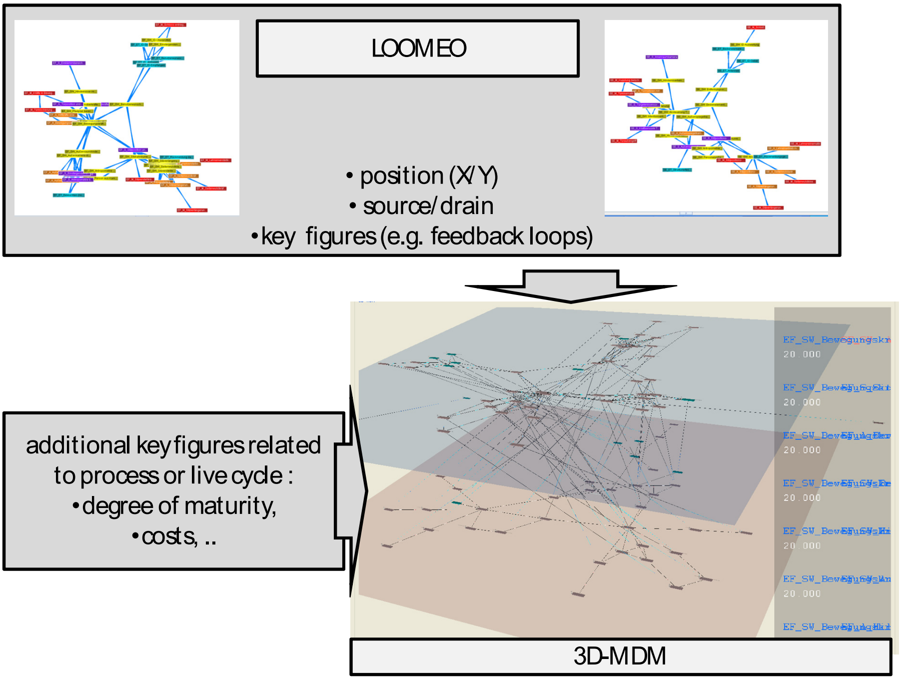

Figure 5. Crosslinking of LOOMEO and 3D-MDM

5. Conclusion and Future Work

The shown approach of 3D Visualisation of MDMs allows for an intuitive and transparent view on complex mechatronic products. By increasing the transparency and with it the understanding of the system this representation assists engineers within the mechatronic development process. A basic prototype has been realized and the results are very promising. In a second step we will use 3D-MDM to visualize how functions and systemelements are linked to the structure of workpackages and milestones [Braun et. al. 2007].
Eppinger and Salminen [Eppinger & Salminen 2001] made the assumption that an alignment of the areas of organisational structure, product architecture and processes structure results in a better competiveness. The 3D-MDM tool gives a first idea about how an analysing method for such an approach may look like. But to really use the tool for analysing the alignment of different domains a mechanism for strength based graphs in 3D has to be developed. We think that this is a very interesting field of research.

References

Maurer, M., "Structural Awareness in Complex Product Design", Produktentwicklung München Band 56, Dr.Hut Verlag, München, 2007

Steward, D.: "The Design Structure System: A Method for Managing the Design of Complex Systems", IEEE Transaction on Engineering Management 28 (1981) 3, pp 79-83., 1981.

Browning, T.: "Applying the Design Structure Matrix to System Decomposition and Integration Problems: A Review and New Directions", IEEE Transactions on Engineering Management 48 (2001) 3, pp 292-306, 2001.

Kallmeyer, F.: „Eine Methode zur Modellierung prinzipieller Lösungen mechatronischer Systeme", Dissertation, Universität Paderborn, ISBN 3-931466-41-8, HNI - Verlagsschriftenreihe, 1998

VEREIN DEUTSCHER INGENIEURE (VDI): „Entwicklungsmethodik fürmechatronische Systeme“, VDI-Richtlinie 2206, Beuth Verlag, Berlin, 2004

Danilovic, M., Browing, T.: “A formal approach for Domain Mapping Matrices (DMM) to complement Design Structure Matrices (DSM)”, In The Sixth Design Structure Matrix (DSM) International Workshop, September 12th-14th, 2004, Trinity Hall College, University of Cambridge, 2004

Bullinger, H., Kiss-Preußinger, E.; Spath, D. (EDS.): „Automobilentwicklung in Deutschland – wie sicher in die Zukunft? Chancen, Potenziale und Handlungsempfehlungen für 30 Prozent mehr Effizienz“, Stuttgart: Fraunhofer-IRB 2003

Teseon: url:[http://www.teseon.com/] 2008

Braun, S.; Diehl, H.; Petermann, M.; Hellenbrand, D.: “Function driven process design for the development of mechatronic systems”, 9th International Design STRUCTURE MAtrix Conference, DSM’07. pp 161-173., 2007

Eppinger, S. D.; Salminen, V.: “Patterns of Product Development Interactions”, In: Proceedings of the 13th International Conference on Engineering Design (ICED 01), Glasgow. Bury St. Edmunds: IMechE 2001

Dipl. Ing. Holger Diehl
PhD Student
Technische Universität München, Institute for Product Development
Boltzmannstr. 15, 85748 Garching, Germany
Tel.: +49 89 28915135
Fax.: +49 89 28915144
Email: diehl@pe.mw.tum.de
URL: http://www.pe.mw.tum.de