# INTEGRATED PROCESS AND PRODUCT MODEL FOR THE EVLAUATION OF PRODUCT PROPERTIES

**Christoph Westphal[1], Sandro Wartzack[1]**
(1) Universitiy of Erlangen-Nuremberg, Germany

## ABSTRACT

Providing the right product information at the right time is an important factor for an efficient product development process and especially for the evaluation of product properties. But due to the mass of different documents and types of data representing product information and the highly dynamic and iterative processes a rigidly or statically linkage between all product and process information is almost impossible. Therefore a dynamic and generic linkage between product and process information is necessary to efficiently support the data logistic within the product development process and to safeguard the steady evaluation of product properties. This approach links product and process information as models within a semantic web. By the additional consideration of the employed tools and methods a quantifiable representation of the information flow within companies can be accomplished. The analysis of this information flow should support the project manager in planning which properties can be evaluated during the product development process with which tools or methods. Moreover a prediction of the quality of this evaluation and of the dependent product information is possible.

*Keywords: Property based product development, process model, product model, semantic networks, property evaluation*

## 1 INTRODUCTION:

Automotive manufacturers are forced to provide vehicles with a multiplicity of diverse properties and functions in order to differentiate from the competitors [14]. To realize this it is necessary to offer a wide spectrum of different variants and extra equipment (figure 1).

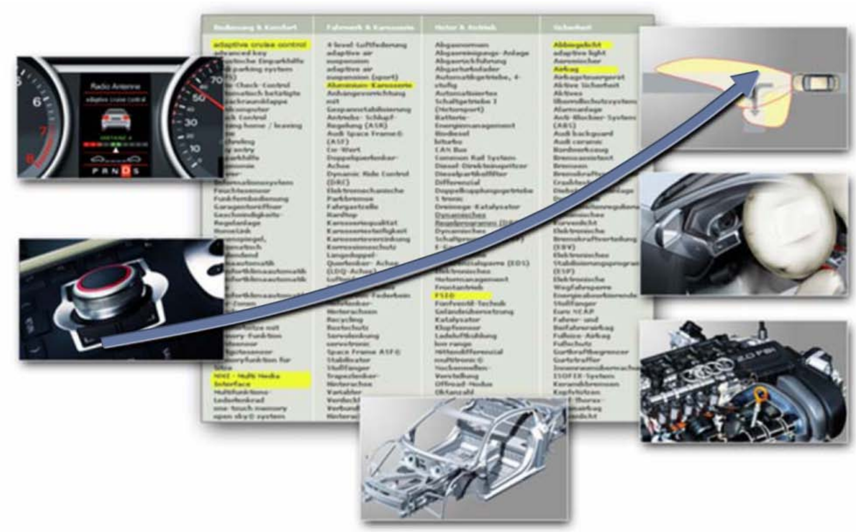

**Figure1:** Increasing variability of functions in the automotive industry [14]

The challenge is hereby to realize all requirements concerning properties and functions of these variants with a high quality standard within short development periods and with low costs. This can only be guarantied if a quality deficit of the product's properties and functions (for example a defective accelerator pedal) is detected early in the product development process (PDP). To realize this, the actual status of achievement of the product's properties and functions has to be continuously monitored during the PDP.

### 1.1 Research question and objective

But one of the most serious problems of monitoring product properties over the PDP is that the product representations – from early sketches over CAD- and CAE files to hardware prototypes – are very different and heterogeneous [4]. Thus the tools and methods, which are used to analyse the product properties on the basis of the existing product representations, are very heterogeneous, too. Consequently it is hardly possible for a project manager to decide, which properties and functions can be analysed at which period of the PDP with which tools and with which quality.
Therefore a methodology will be presented in this paper how to link a process and a product model in order to relate the product validation process to product properties or functions. This linkage enables the chronological arrangement of product validation processes within the PDP according to their dependent input information.

### 1.2 Procedural method and benefit

To reach this aim it is firstly necessary to model the development and especially the product validation process. On the basis of this process model it is possible to identify all information, which are needed to analyse and to evaluate product properties. In a next step a product model has to be created, which is able to classify and to group these information. By linking the product and the process within a semantic web it is possible to create a network which represents the information flow of the product validation process.
This so called directed graph can now be used to determine and calculate the chronological order of the product validation processes during the PDP on the basis of their dependent input information. Moreover the total quality of the product validation process can be calculated and therefore the used tools and methods can be optimized. Finally this semantic web enables the conception of property oriented data management.

## 2 FUNDAMENTALS AND RELATED APPROACHES

In the following chapter some fundamentals and related approaches should be discussed, which are the scientific background for this paper.

### 2.1 Micro- and macro cycle of the PDP

To represent the information flow of the development process it is necessary to have a look at the fundamental tasks of a product developer during the design process. According to WEBER'S CPM/PDD approach the product developer defines a solution concept during synthesis, which has to meet the requirements. This is done by influencing or determining the so called characteristics like structure, shape or material [19]. During analysis the properties, which describe the behaviour like safety or reliability of the created solution concept are identified. If the actual properties don't match the target properties either the solution concept has to be optimized or the requirements have to be modified. The benefit of Weber's approach for this work is that it focuses on the two core types of product information – the characteristics and the properties – and embeds them into a control loop like the TOTE-scheme described by MILLER, which ends, if the actual properties of a product match the target properties (requirements) [11]. Due to the fact that this procedure describes a problem solution cycle and is done many times during product development it is called a micro cycle [17].
But development processes like the development of a vehicle, which lasts up to 4 or 5 years can't only be described with the fundamental steps of the problem solution circle. A lot of tasks have to be done in a certain chronology. Therefore so called process models have to be created to model the main steps of product development and to support the developer in each of these steps. Different process models have been developed during the last fifty years to describe this macro cycle of the PDP. HUBKA and EDER *("Theory of technical systems: A total concept theory for engineering design")*, PAHL and BEITZ ("*Planning, cenceptualizing, designing and elaborating*"), EHRLENSPIEL ("*Procedure-cycle*"), and SUH ("*Theory of axiomatic design*") as well as ULRICH and EPPINGER *("Produkt Design and Development")* and many more developed different models to define the important procedure steps as well as the fundamental procedure during development. Some of them are the basis for the VDI 2221 guideline - *Systematic approach to the development and design of technical systems and products* –

which has been introduced in 1986. Other disciplines like software engineering have developed other models like the waterfall model, the fountain model or the whirlpool model.
A mechatronical approach is the VDI 2206 - *Design methodology for mechatronic systems*. It brings different domain specific guidelines together and deals with the development of a modern mechatronic product in its entirety. It describes the development process as the detailing of requirements (system design), the design process within the single domains (domain specific design) and the aggregation of analysis and their results (system integration) [17]. These three phases are embedded into a modelling and model analysis phase. Next to this reason and the fact that this guideline supports the macro cycle of mechatronic products it is very important for this paper, because it is one way to describe the information flow though the product structure and hierarchy (from the whole product to the domain specific component).

### 2.2 Product modelling

To create a generic information flow it is necessary to classify and model the product information during the development process. Product information occur within documents or data, which should be designated as product representations within this paper (e.g. specification lists, or test results). According to ANDERL the product model is the result of the PDP and describes the product in a formal way and documents all relevant properties and characteristics during the PDP [2]. GRABOWSKI et al. specifies the term product model and defines the so called integrated product model, which consists of different partial models representing different views (lifecycle stages, disciplines etc.) [1]. The most important approach to describe these partial models is the DIN ISO 10303, which has different application protocols (APs) for different views or domains (for example the AP 214 for the automotive industry), and which is based on the modelling language EXPRESS. Moreover some application protocols describing properties and requirements– the DIN ISO 10303 AP 233 *Systems Engineering Data Representation* and the DIN ISO AP 235 *Engineering Properties for Product Design and Verification* are state of the art. These application protocols are the basis for the clustering of the partial models described in chapter 3.2. Another important differentiation of product models is done by PONN and LINDEMANN. The product models are distinguished to due to their level of concretion (requirement-model, functional-model, working-model, and building-model). Therefore this approach gives an important instruction about the different components of a solution concept [12].

### 2.3 Integration of the product and process model

The integration of product and process model can be defined as the conjunction between product and process information. This conjunction can be achieved in different ways. On the one hand product representations can be statically assigned to process steps (e.g. classical workflow management systems (WMS)). On the other hand product representations can be described with attributes and parameters. This enables the dynamic linkage between those product representations and the PDP [7]. To get a generic linkage between product and process information it is necessary to classify these information within a process and a product model. The linkage between these two models can be done directly be done by linking parts of the product model as design objects to the process model [18]. An other way of linking process and product models is to build up a meta model, which creates a generically conjunction between the product model and the process model [4]. This kind of meta modelling of the product and the process model is an important precondition for the creation of an semantic web in order to represent the information flow during the PDP.

### 2.4 Semantic webs and ontologies

Ontologies as a subtype of semantic webs become more and more important for the modelling of product information, because they are able to handle and to interchange knowledge [11;15]. Next to the attempt of describing and handling information of different domains, ontologies are used to retrace information and to analyse information along the product lifecycle [3]. In contrast to classical data or product models ontologies are able to model classes, relations, attributes as well as concrete product representations like CAE-models as individuals. Therefore it is possible to group product representations together in the product model and to link them to a process model, which can also be modelled in ontology. Due to these benefits an ontology should be used in this paper to represent the information flow, because it is possible to create a meta model as well as to model the product representations in one software environment.

## 3 METHODOLOGY FOR THE DESCRIPTION OF THE INFORMATION FLOW

In the following chapter a methodology should be described how a linkage between the product and the process model can describe the information flow, which is necessary to evaluate product information.

### 3.1 Process model

To describe the product development process focused on properties it is necessary to combine the micro and the macro process. Thus it is necessary to consider the correlation between characteristics and properties (micro process) as well as the correlation of properties along the product structure or hierarchy (macro process). Due to the fact that the control loop describing the micro cycle was only fundamentally described it has to be enlarged on the steps requirements synthesis, analysis model creation and the evaluation. During the requirements analysis the requirements are described with objective or subjective measurable quantifiable target properties [8]. Moreover the boundary conditions of the product validation like the test scenario or the tested product configuration have to be specified. During synthesis the product characteristics are defined by the designer (see chapter 2) and are filtered, pre-processed and assembled to an analysis model during the model creation step. In the following the product properties are analysed (for the calculation of the *system pressure* of an air supply aggregate a simple model of the *piston force* and the *piston area* is needed. This *piston area* can for instance be pre processed out of the *geometry* of a CAD-model). And finally these actual properties are compared with the target properties. This evaluation decides if the next step of the macro process can be done or if an iteration loop begins.

According to the V-model the PDP can be separated into three parts. During the first phase the requirements are specified and detailed. Considering HERFELD this has to be done along the product hierarchy, whereas the solution concept of a top layer defines the requirements for the next deeper layer (for example the property *driving comfort* of a vehicle (product) requires a special *suspension* (assembly) as solution concept. This *suspension* requires a distinct *damping ratio*, which should be realized with an *air damper* (subassembly). This *air damper* requires again a distinct *system pressure* of the *air supply aggregate* (component)) [5]. Therefore the micro cycle has to be processed on every layer of the product hierarchy until the solution concepts fulfils the requirements, which are dedicated from the solution concept one layer higher in the product hierarchy (figure 2).

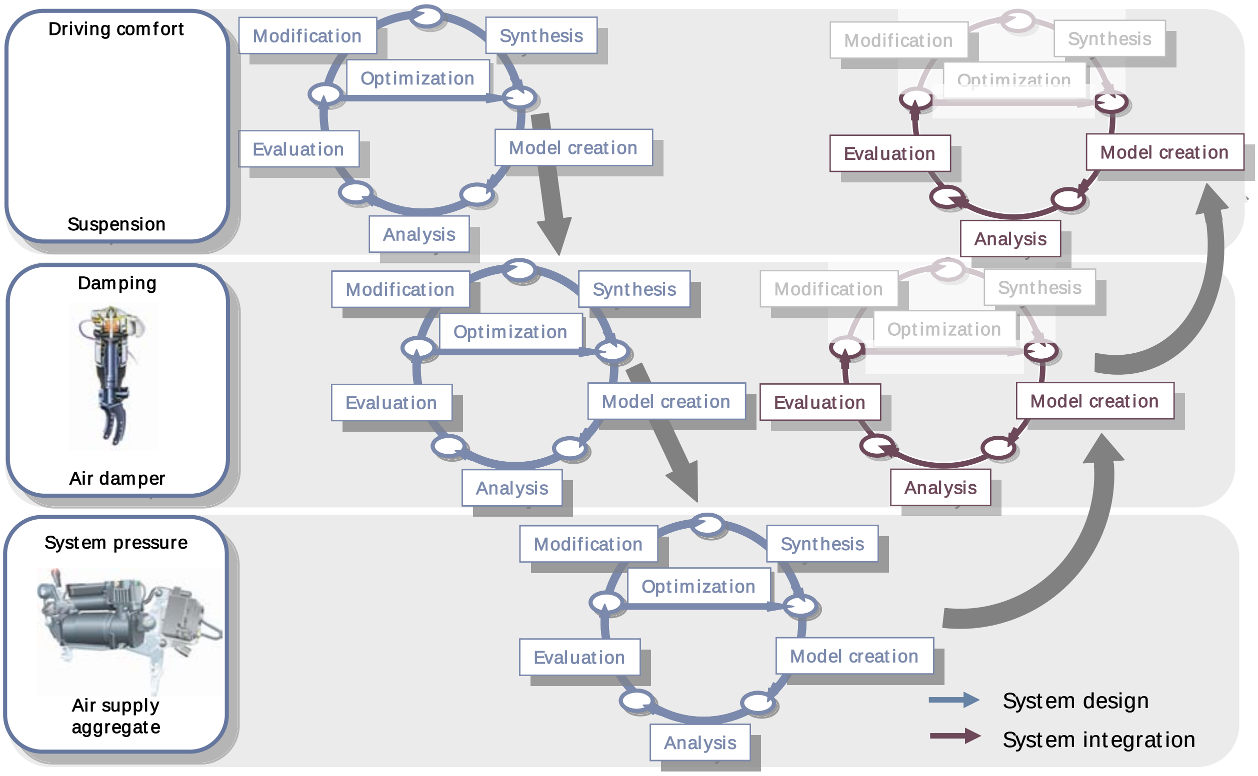


**Figure 2: Process model of a property based product development**

During detailed design each discipline has to create solution concepts for their components. And finally those solution concepts are tested and assembled to sub assemblies, assemblies and to the product in order to evaluate the properties und functions during system integration. But unlike the system design phase the system integration phase has no synthesis, modification or optimization steps, because if a property isn't fulfilled a drawback to the system design phase is necessary. Consequently it can be postulated that the macro process of the PDP is a distinct sequence of the micro cycles of different layers of the product hierarchy.
In order to detail the six main process steps (synthesis, model creation, analysis, evaluation, modification and optimization) and to investigate the product representations occurring during the micro cycle these steps were discussed and described within workshops with process developers of an industrial partner.
The ARIS scheme (a graphical modelling language for the representation of business processes) was used to model these steps because it allows a hierarchical process modelling and the description of the input and output documents, data, or information in context of each process step (figure 3).

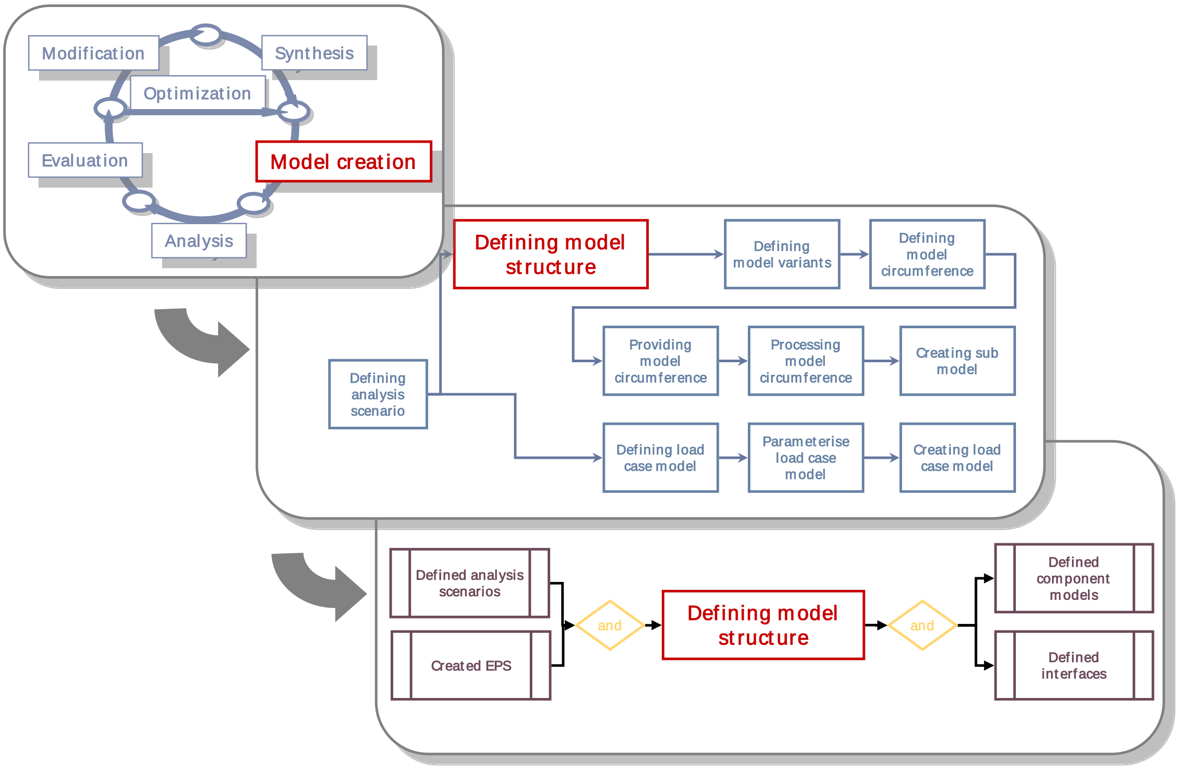


**Figure 3: Detailing of the process model of a property based product development**

This enables the description of all product representations occurring during the product development process, which are necessary to validate the product properties. Moreover each product representation is linked at least to one process step as an input- or an output object.

## 3.2 Product model

As mentioned before a product model is a formal specification of the product information and should support a standardized description of product information as well as the management of product representations. To capture all product information it is necessary to investigate the product representations about their containing information. Consecutively these pieces of information have to be classified in objects and related to each other in order to create a product model (for example as an entity relationship diagram).
This creates a static mapping between the components of the product model and the process model, whereas the pieces of information are the linkage because they occur within the product representations in the process model as well as within the objects in the product model (for example the maximum speed of a car is a piece of information in the specification list which is the input for a synthesis step. Moreover this maximum speed is a target property which is an object of the product

model). Additionally this enables a so called m-n relation which means that a product representation contains one or more objects of the product model and vice versa (for example a specification list contains the requirement maximum speed on a dry highway. The maximum speed is a target property object whereas the dry highway is a test scenario object).
In a last step the objects of the product model can be grouped to so-called partial models. Figure 4 shows these objects as well as their arrangement. In this product model product properties are arranged within a property structure. Assessment criteria are used to describe those properties on a measurable level (for example the property driving comfort can be described with a maximum yaw angle, which can be measured during an ISO lane change manoeuvre (also called moose accident test)). Three kinds of assessment criteria exist for the description of the target properties, of the actual properties and for the current status of the properties (comparison between target and actual properties). The target properties are requirements for the solution concept. This solution concept consists of a chosen principle functionality of his product or component and characteristics, which has to be defined by the designer. The designer has to define the fundamental parameters (for example the geometry or the material) the structural architecture as well as the configuration of his product. The analysis model is an assembly of distinct characteristics or distinct components out of the solution concept and has to be analysed within a test scenario (also called load cases). These test scenarios can be grouped, too (for example NCAP crash scenarios). The results of an analysis are the actual assessment criteria, which have to be compared against the target assessment criteria in order to evaluate the status the solution concept.

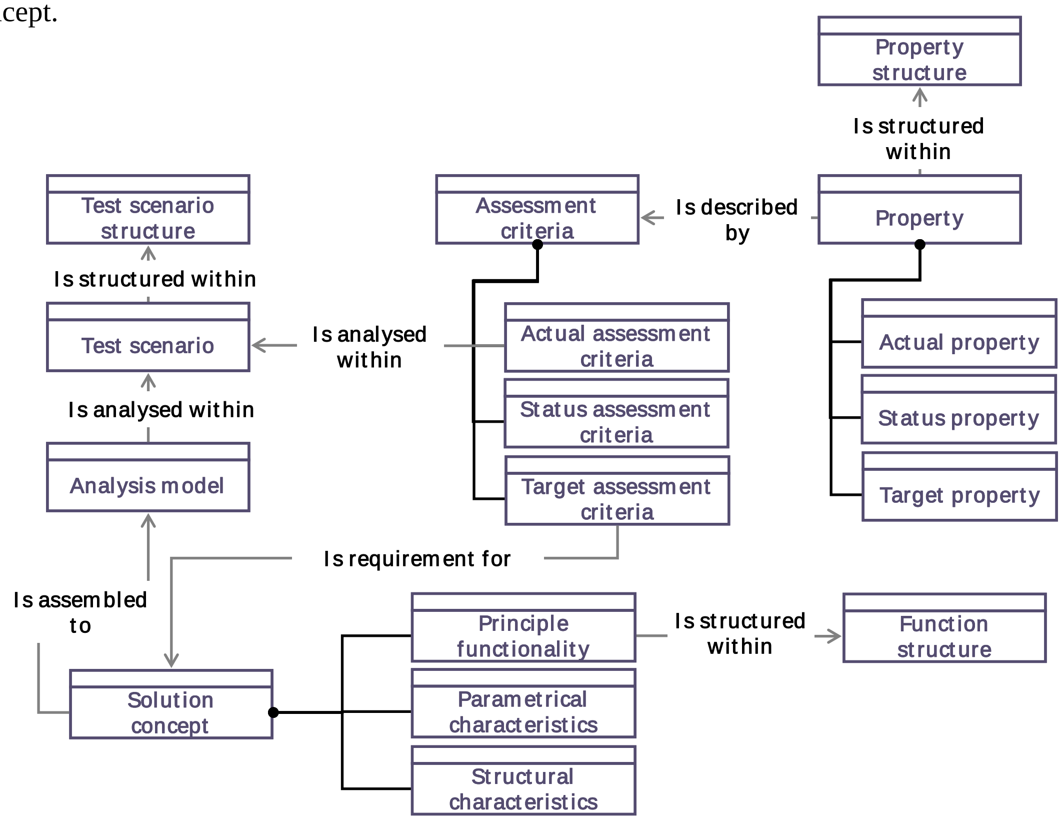


**Figure 4: Product model of a property based product development**

Though this is just a very high level view on the classification and relation of product information, this is basis to group theses objects into partial models. The properties, their structure and the describing target assessment criteria as well as the description of the test scenarios and their grouping display the partial model “requirements”. The partial model “solution concept” consists of the meta description of the component as well as the characteristics and the description of the functionality. The partial model “analysis model” consists of the different models, which where build up for testing the properties. The partial model “analysis result” displays the actual assessment criteria and therefore the actual properties. And finally the partial model “evaluation” consists of the status of the assessment criteria and therefore of the status of the properties. The last two partial models can be structured like the requirements within a property structure. This grouping allows on the one hand a target oriented data management support and on the other hand a generic linkage can be accomplished between the partial models and the process steps. Consequently the requirements elicitation step results in the partial model “requirements”, the synthesis step results in the “solution concept” the model creation step results in the analysis model, the analysis step determines the “analysis results” and the evaluation step determines in the partial model “evaluation”. Considering the product hierarchy the “solution concept” as described in chapter 3.1 can be defined as “requirements” for one level below within the product hierarchy during system design or it can be assembled to a solution concept one level higher within the product hierarchy during system integration. This enables a generic mapping between the process model of the micro cycles and the partial models, because each of the six main process steps transform information of a distinct partial model to an other distinct partial model (for example a crash analysis uses a FE-model (analysis model) as input and calculates the maximum deformation (analysis result)). Now it is possible to generically map each product information of an object within a partial model to a distinct process step (for example the maximum speed as a target property is always used as input for a synthesis step as well as an input for an evaluation step). Without this classification of the product information within the product model no a generic conjunction between these product information the process wouldn’t be possible, because the product representations can only be statically linked to process steps. Therefore it is possible to describe the information flow, which is necessary to validate the product properties during the development process with this sequence of process steps and objects of the product model (figure 5).

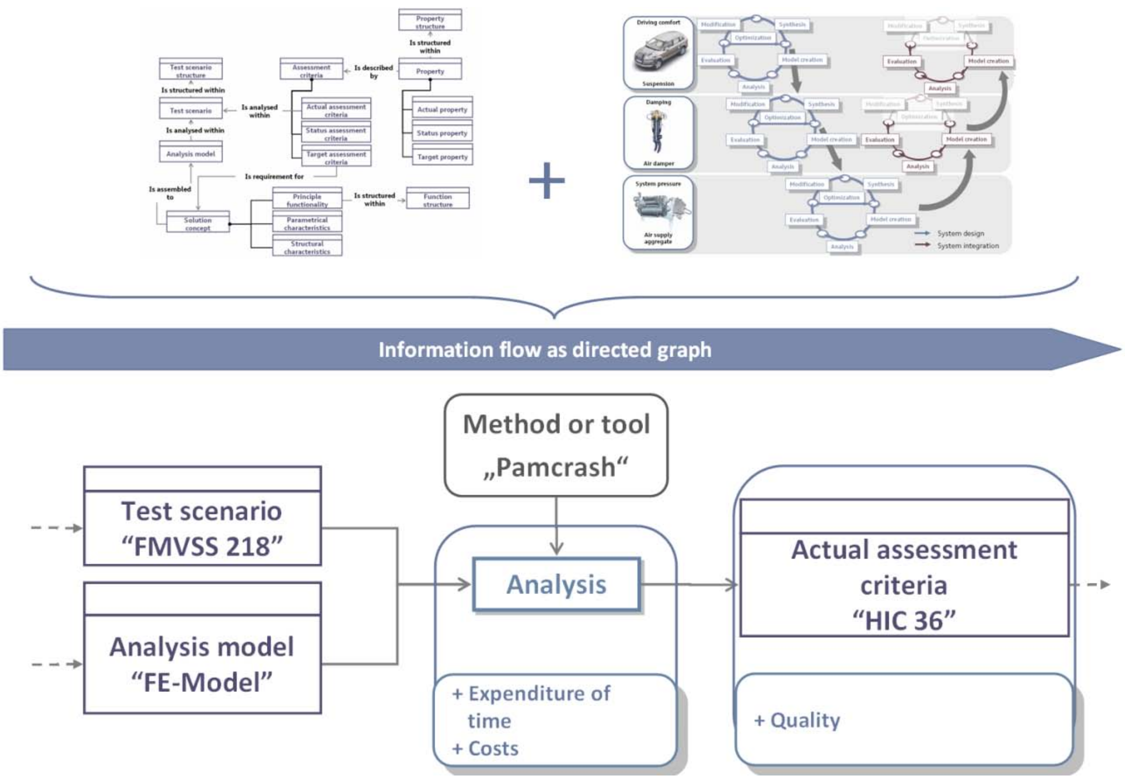


**Figure 5:** Information flow as directed graph

But to model the real information flow within a company the consideration of the employed tools and methods during the PDP is necessary. For each process step a tool or method has to be used for the transformation of the incoming into outgoing objects of the product model. This transformation affects the quality of the outgoing product representations. Moreover the necessary expenditure of time and the costs of each process step depend on the employed methods or tools. Figure 5 shows the example of an analysis step (crash analysis) which transforms the ingoing test scenario and the analysis model into outgoing assessment criteria (HIC 36 – criteria for the heaviness of an injury after a vehicle crash)). By using the tool Pamcrash for this analysis the process step has a distinct expenditure of time and costs and the outgoing assessment criteria have a distinct quality.

Consequently the consideration of the used methods and tools allows an allocation of quality to each product representation and an allocation of an expenditure of time and costs to each process step. One boundary condition has to be defined to analyze the information flow of the property based product development process. Due to the fact that most companies have a component based product development processes the point of time in the PDP when different stages of solution concepts have to be released are fixed to milestones. This means that the information flow has the solution concept with a fixed start point in the PDP as boundary condition [16]. This enables four types of analysis of this information flow to answer the described problems in chapter one:

- By summing up the expenditure of time of all process steps of one information chain it is possible to calculate the point of time during the PDP at which a property can be evaluated
- The necessary tools and methods can be displayed for each information chain
- The quality of the property evaluation can be displayed by calculating the quality propagation on the basis of the quality
- The components of the solution concept can be identified which are necessary for the evaluation of a property. This is extremely worth fully for a change process if a target property isn't reached.

## 4 PROTOTYPICAL IMPLEMENTATION

A prototypical implementation of this concept was built up in the ontology editor Stanford Protégé. As mentioned in chapter 2 ontologies are a special type of semantic webs and have pivotal benefits of connecting and relating information. To implement the methodology for the description of the information flow it is necessary to create three stand alone ontologies in the first step:

- Process model ontology
- Product model ontology
- Tools & Methods ontology

The process model ontology is used to describe the micro cycle of the PDP. The product model ontology is used to describe the product model with its five partial models (requirements, solution concept, analysis model, analysis result, evaluation). The tools& methods ontology has to be created to model the tools and methods which are used during the development process. The relations between these three ontologies are described within a top layer ontology. Moreover it includes the rating of each product representation and process step in context of the used tools.

The next step is to feed the ontology with test data to create a data basis. Therefore product representations were modelled as instances within the ontology. To analyze the information flow, which is necessary to evaluate product properties, some queries have to be verbalized. These queries should enable to sum up the expenditure of time of all process steps of one information chain, to display the necessary tools and methods, to calculate the quality of the whole information chain and to identify the components out of the solution concept which are necessary to evaluate the product property (figure 6).

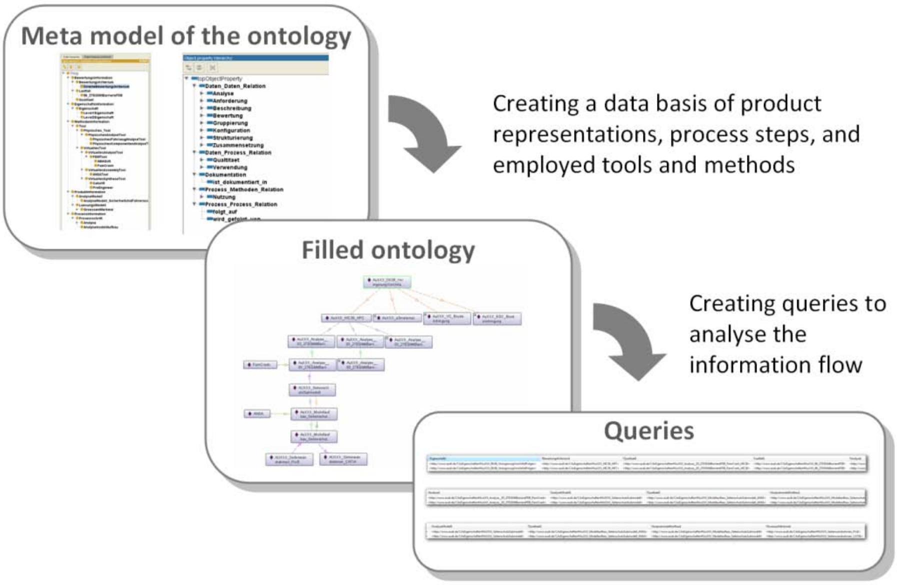


**Figure 6:** Prototypical implementation of the described methodology

## 5 CONCLUSION AND DISCUSSION

In the same dimension as products and their requirements become more and more complex the PDP becomes more and more complex, too. Therefore the management and the planning of the PDP is an important task to ensure the fulfilment of all product requirements. For this reason the question arises how the responsible project manager can be supported in monitoring the product properties. This paper presents a methodology how to model the information flow within the PDP by integrating a process and a product model. The process model is fundamentally based on the problem solution cycle, which is repeatedly processed during the different phases of the V-model. A mapping of the product representations onto the objects of a product model enables a generic conjunction between those two models. With the integration of the used methods and tools and the resulting quality, costs and expenditure of time the process can be described as directed graphs with qualitative relations, which can be analysed in order to answer the question how to monitor product properties. A prototypical implementation of this methodology and examples for prototypical queries were shown in chapter 4. This example proofs that the methodology is able to describe and to manage an interpretable and traceable information flow during the system integration phase. But it also points out two main problems of this methodology: Firstly it is enormous extensive to collect and especially to connect the input and output product representations for each process step. Consequently it is surly impossible to manage the information flow only within an ontology as shown in the prototypical implementation. To solve this problem the methodology and ontology must be able to monitor the data flow of different product data management systems (PDMS). In this case the ontology is able to access the data within the PDMS and to connect different PDMS during product lifecycle. This integration is a current topic of research [9]. The other problem is that especially the early phases of the PDP are highly iterative. The methodology is able to describe these iteration cycles but it is almost impossible to forecast how many iteration cycles have to be processed until the solution concept matches the requirements. Therefore an analysis of the expenditure of time and of the costs is extremely difficult in the early phases (especially at developing new products). But nevertheless the cross linking can be described and supported between data deliverers and data receivers. Moreover the quality of the product validation process can be described in dependence of the used tools impartial of the duration of the micro cycle.

## 6 FUTURE WORK

Future work will include further validation and optimization of the meta model of the semantic web in order to test if it can really model a generic information flow during all phases of the PDP. Therefore a case-by-case analysis has to be integrated to analyse if the current phase of the design process is either the system design or the system integration. The enlargement of the data basis will finally allow the utilization of the information flow in order to develop a property testing map. By comparing and analyzing the different information chains depending on the respectively used tools it can be calculated when a property can be evaluate during the PDP with which quality and with which tools or methods. These results can be displayed in such a map, which plots the possible validation methods and tools against the milestones of the PDP. This map should give the project manager an overview which properties can be evaluated during the PDP with which methods and with which quality. This will be an important step for the frontloading and raise of efficiency of product property evaluation.

## REFERENCES


[1] Anderl, R., Grabowski, H., Polly, H., Integriertes Produktmodell – Entwicklungen zur Normung von CIM, Berlin, Beuth, 1993.

[2] Anderl, R., Trippner, D., *STEP Standard for the Exchange of Product Model Data - Eine Einführung in die Entwicklung, Implementierung und Nutzung der Normenreihe ISO 10303 (STEP)*, B.G. Teubner Verlag. Stuttgart, Leipzig, 2000.

[3] Assouroko, I., Ducellier, G., Belkadim, F., Eynard, B., Boutinaud, P., *Improvement of engineering design and numerical simulation data exchange based on requirements deployment: a conceptual framework*, in Proceedings of the 7th International Product Lifecycle Management Conference, Bremen, 2010.

[4] Grebici, K., Wynn, D.C., Clarkson, P.J., *Describing information use in engineering design processes using a diagrammatic model*, in Proceedings of the 17th International Conference on Engineering Design ICED’09, Stanford, 2009.

[5] Herfeld, U., *Matrix-basierte Verknüpfung von Komponenten und Funktionen zur Integration von Konstruktion und numerischer Simulation*, PHD Thesis, München, 2007.
[6] Horridge, M., *A practical Guide to building OWL ontologies using Protégé 4 and CO-ODE* Tools, Edition 1.2, Handbook, Manchester, 2009.
[7] Lauer, W., Lindemann, U., *Dynamic linking and retrieval of product models to product development processes*, in Proceedings of the 17th International Conference on Engineering Design, ICED'09, Stanford, 2009.
[8] Maletz, M., *Integrated Requirements Modeling*, PHD Thesis, Graz, 2008.
[9] McMahon, C., Li, C., *Ontology-Based Annotation in PLM Systems*, in Proceedings of the 7th International Product Lifecycle Management Conference, Bremen, 2010.
[10] Miller, G.A., Galanter, E., Pribram, K., *Plans and the structure of behaviour*, New York, 1960.
[11] Ovtcharova, J., Marinov M., Mahl A., Schubert, P., Bittel, V., *Ontology based, function oriented support for cross-domain and cross-enterprise engineering*, in Proceedings of International Conference on Collaborative Mechatronic Engineering, Salzburg, 2009.
[12] Ponn, J., C., Lindemann, U., *Konzeptentwicklung und Gestaltung technischer Produkte,* Springer-Verlag Berlin Heidelberg, München, 2008.
[13] Scheer, A. W., *Architektur integrierter Informationssystem,*. Springer, Berlin 1992.
[14] Seiffert, U., *Virtuelle Produktentstehung für Fahrzeug und Antrieb im Kfz*, Vieweg und Teuber Verlag, Wiesbaden, 2008.
[15] Suh S-H., Shin S-J., Yoon J-S., Um J-M., *A new paradigm for product design and manufacturing via ubiquitous computing technology*, International Journal of Computer Integrated Manufacturing, 2008.
[16] VDA, *Automotive VDA-Standardstruktur Komponentenlastenheft,* Frankfurt, 2007.
[17] VDI Richtlinien 2206, *Design methodology for mechatronic systems*, VDI Handbuch Konstruktion, Düsseldorf, 2004.
[18] Wartzack, S., *Predictive engineering – Assistenzsystem zur multikriteriellen Analyse alternativer Produktkonzepte,* PHD Thesis, Erlangen, 2000.
[19] Weber, C., *CPM/PDD - An extended theoretical approach to modelling products and product development process*, in Proceedings of 2nd German-Israeli Symposium on Advances in Methods and Systems for Development of Products and Processes, Fraunhofer-IRB-Verlag Berlin, 2005.

Contact: Christoph Westphal
University of Erlangen-Nuremberg
Chair of Engineering Design (Department of Mechanical Engineering)
Martensstrasse 9
91058 Erlangen, Germany
Tel: +49 151 58402062
EMail: westphal@mfk.uni-erlangen.de
URL: www.mfk.uni-erlangen.de

Dipl.-Ing. Christoph Westphal is research associate at the Chair of Engineering Design at the university of Erlangen-Nuremberg since 2008. He studied mechanical engineering at the university of Erlangen-Nuremberg; his main focus is the integration of design and simulation.

Prof. Dr.-Ing. Sandro Wartzack is head of the Chair of Engineering Design at the Friedrich-Alexander-University of Erlangen-Nuremberg, where he also received his PhD in 2000 and studied mechanical engineering some years before. After finalization of his PhD Study he was employed in an international automotive supplier company as director of virtual product development. Currently, he is member of the Design Society, honorary member of TechNet Alliance, member of Berliner Kreis, member of VDI, member of the scientific committee of ICED, DESIGN Conference and CIRP Conference on Computer Aided Tolerancing, expert advisory board member of the conference 'plastics+simulation', Editor of the Symposium DFX, as well as SIG Leader of the SIG 'Decision Making' in the Design Society.