INTERNATIONAL DESIGN CONFERENCE - DESIGN 2010
Dubrovnik - Croatia, May 17 - 20, 2010.

THE ROLE AND APPLICATION OF ACTIVITIES IN THE INTEGRATED PRODUCT ENGINEERING MODEL (IPEM)

A. Albers and S. Muschik

Keywords: iPeM, modelling, activity

1. Introduction

The knowledge of technologic systems and their interdependencies has evolved in past decades and the degree of multidisciplinarity in the development of products has grown immensely. Companies face continuously rising pressure for better products in shorter development times and higher quality.
Research tried to address these issues by projecting real product engineering processes onto product engineering process models to support analysis, planning and execution of the processes. Until now there is no single approach commonly accepted as concurrently capturing all sorts of possible design situations while being specific enough to provide support on operative work level [Clarkson, Eckert, 2005]. Models are in the one case abstract models, built rather prescriptively and do not contain precise content description on how to derive a process model for practical application. In the other case they are concrete procedural models, being too detailed to be applied to multiple design situations. The transition from one view on engineering processes to the other has not been sufficiently described yet.
Insights from a study of the development of systems of objectives in early activities of product engineering processes in practise [Albers, Muschik, 2010] have shown a need of management for methodological support in structuring and planning engineering activities and matching appropriate tools and methods. Since existing process models mainly base on stage gate approaches they are rigid in sequence and not decomposable to operative work level. Existing process descriptions on this level are usually the result of organisational change measures and not flexible for precise situation specific planning as necessary for the unpredictable progress of early activities with changing constraints. The actual agents of the process often refrain from using process models because the benefit in daily work is not given on a direct, intuitive basis, e.g. providing procedures on how to execute certain activities or supporting knowledge management by matching information for reuse in future projects.
An approach to this topic is the Integrated Product Engineering Model (iPeM) [Albers, 2010]. Its objective is to address the issues described above in providing support for product engineering, based on a systemic comprehension of engineering processes from an abstract meta model to application in practice. The model has been defined as a meta model based on five hypotheses about product engineering. Extending the model's applicability for practical purposes is focus of current research.
As part of this research, this paper investigates the course of the transformation from a meta model to an implementation model and elaborates ideas for an approach on how to improve the applicability of the iPeM. The development process of systems of objectives in product engineering is used as example to discuss the ideas presented in this paper. The development of systems of objectives is a current research field of the authors and is investigated by the implementation of the research approaches in practice at a German car manufacturer.

To provide a common understanding of process modelling, the basic elements and requirements are discussed and compared to current approaches on engineering models in chapter 2. The findings are used for developing ideas for an approach to increase the applicability of the iPeM in chapter 3 on a theoretic level. The ideas are then deployed onto an example from the development process of systems of objectives at the car manufacturer in chapter 4. The potential of this initial approach is discussed in chapter 5 and supplemental future work is listed.

2. Literature Review

Many approaches have been developed to describe product engineering processes. So far the approaches have not fully succeeded in capturing all design aspects sufficiently while being able to give detailed aid on how to conduct design activities in practice [Clarckson, Eckert, 2005]. Contrariwise most approaches are narrow in focus. The following paragraphs discuss the state of the art and difficulties of process models with focus on the role of activities in modelling processes.

2.1 Process modelling

Characteristics of models

There are multiple classification schemes which are used to characterise different types of models.

Stage- versus activity-based models. Stage-based models are characterised by a serial and chronological sequence of process steps. An activity-based model on the other hand represents problem-solving activities, being cyclic due to rework iterations. Other models regard the stage-based view lying orthogonal to the activity-based description. This idea is extended in models where the solution space narrows while traversing the sequential stages of the process [Clarkson, Eckert, 2005].

Solution- versus problem-oriented literature. Solution-oriented approaches start off with an initial solution, which is then iteratively adapted as the product engineering process progresses and the requirements become clearer. Accessing a design issue with a problem-oriented approach firstly abstracts and analyses the problem before solutions are developed. Stage-based models rather use problem-oriented strategies while activity-based models might use either one [Clarkson, Eckert, 2005].

Abstract versus analytical versus procedural approaches. Abstract models contain a generic description of product engineering processes separating them into few activities to capture a broad bandwidth of design situations. Analytical approaches are project-oriented, using a representation form to describe the specific project part and methods or tools using this representation for application during execution. Procedural models are more specific, use more stages, usually address a certain type of user but are one-dimensional in modelling process sequences [Clarkson, Eckert, 2005].

Descriptive versus prescriptive modelling approaches. The transformation of implicit knowledge about how activities are conducted in reality into a model representation is illustrated in a descriptive model approach, which is built inductively [Browning, et al., 2006]. Prescriptive models are built deductively and are meant to improve distinct parts of design projects by giving advice on how to proceed in certain situations. Combining descriptive and prescriptive modelling supports consistency [Browning, et al., 2006]

Classification of models

There are different abstraction levels to be differentiated when regarding process models. Rupprecht [Rupprecht, 2002] distinguishes model levels and levels of individuality. The model levels describe different grades of abstraction of the formal structure of a model. A meta model defines formal guidelines, a kind of common language. On the model level of type, this language is used for representing abstracted classes of processes with similar characteristics (“Ausprägungen”). Models on the level of characteristics represent real characteristics of a process model. The specification of the content of a model is depicted with different levels of individuality. Constraints to processes rise when their content gets more detailed, which needs to be taken into account when defining the model. Models capture more application cases when their content is more abstract, e.g. when several detailed process steps are summarised to a more abstract one. The correlations of characteristics of models (cp. previous paragraph), depending on the model’s abstraction level, are visualised in Figure 1.

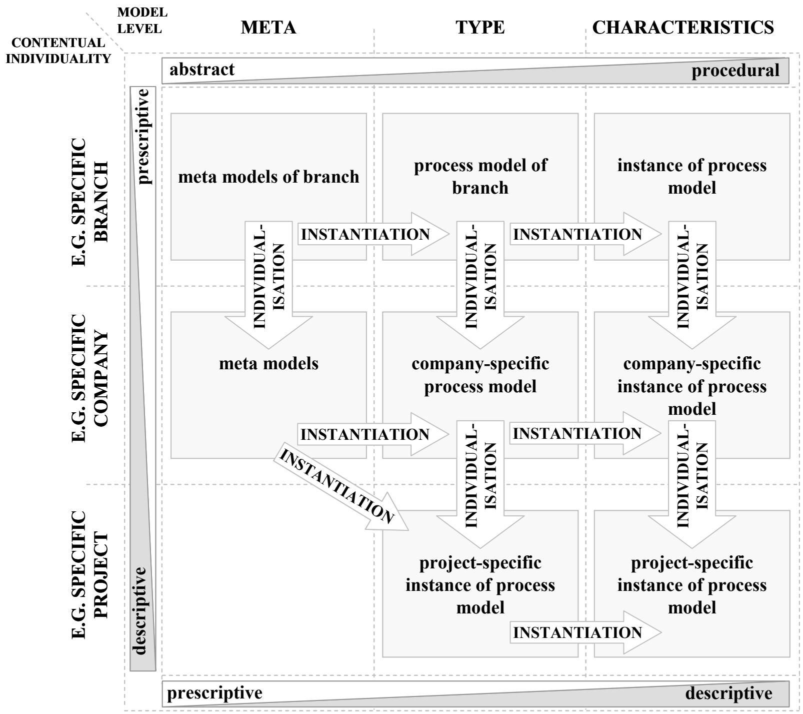

Figure 1. Abstraction levels and model characteristics [Rupprecht, 2002]

Requirements on product engineering process models

In contrast to models of conventional business processes, product engineering processes impose difficulties in modelling because of their complexity, multidisciplinarity and interdependencies of activities and respective agents and tools. Ambiguity is higher, conflicting constraints exist [Clarkson, Eckert, 2005] and especially early activities are characterised by a high level of creativity, processing information with little knowledge about interconnectedness. Parallelism of processes is high with many iterations [Browning, et al., 2006]. Uniqueness in sequence and content of product engineering processes [Albers, 2010], involving varying activities, demands flexibility of a model and makes standardisation more difficult.

2.2 Activities in the context of modelling product engineering processes

Systemic aspects for process modelling

A system can be described from different points of view. Structurally, by the relations between its elements and functionally by changes of its state caused by in- and output. Each system consists of subsystems and represents itself a subsystem of a supersystem [Ropohl 2009]. Several approaches using these premises for modelling multidisciplinary engineering processes agree that these can be described as systems [Rupprecht 2002], consisting of a system of objectives, which is transformed into a system of objects. The transforming system is an operation system [Ropohl 2009], which can be divided in operation system (agents system) and process system [Browning, et al., 2006]

Components of a model

To understand complex process behaviours, Browning et al. recommend to look at the constituent parts and their relationships (*decomposition paradigm*) [Browning, et al., 2006]. These are *activities*, being "packages of work to be done" for achieving results. Attributes (*resources*) are amongst others time, money but also personal motivation. Attributes need certain inputs (*deliverables*) which are transformed into outputs (*deliverables*). The flow of deliverables generates the dependencies between activities. Activities are often named processes, process elements or also subprocesses or tasks. This is due to the decomposability, since each process is a set of activities, but at the same time represents an activity as part of a superior process on the next aggregation level. However, the number of hierarchy levels cannot be defined generally, due to the differing granularities [Rupprecht, 2002].

Modelling activities in product engineering processes

The understanding of activities in the context of modelling product engineering processes of Hubka and Eder [Hubka, Eder, 1996] is based on the hypothesis that designing is a rational cognitive activity, which is decomposable into smaller steps (cp. to previous paragraph), i.e. into different levels of abstraction to match the relevant design problem. They propose a hierarchy of design activities, from design stages and according operations to basic operations, representing the problem-solving cycle down to the lowest level of elementary activities and operations. Every activity on one level comprises the activities on the following lower level and is itself part of all of its superior activities.

Sim and Duffy share the assumption that design activities can be depicted on different layers of abstraction when regarding activities on a knowledge level [Sim, Duffy, 2003]. In contrast, they relate activities to the knowledge of the agent in the developing design and not to the relevant state of information about the design. They classify the main activities in *design definition*, *evaluation* and *management*, which proceed in sequence or in parallel, depending on the endeavoured design goal. They conclude that the main problem to detect generic design activities is the level of abstraction.

Modelling different activities of product engineering processes

Arguing from an operational level of activities, Browning et al. [Browning, et al., 2006] find that product engineering processes show recurring structures due to the nature of design, containing many consistent patterns. It can be summarised that when activities are abstracted to a higher level, they resemble in structure. Being more specific they differ in structure and attributes, as the early activities and the detailed design activities differ in the degree of available information and knowledge about design goals. Browning et al. recommend modelling each activity separately but integratably. Subsequently the different models can be unified on a higher abstraction level. With this procedure best practices from different activities can be evaluated and exchanged.

An example for a procedural process model is the FORFLOW-process model of Krehmer et al. [Krehmer et al. 2009], which is rather stage oriented and has the objective to provide descriptive and prescriptive support. The model arranges process steps on three different levels, whereas the first level functions as a guide for the agent through the process listing main development stages. The second level contains multiple steps detailing one of the processes from level one, level three respectively details level two with nearly ninety partial steps. First and second level are to be conducted in certain order, steps on level three can be applied specifically for a particular situation.

An industrial study at a German car manufacturer was used to identify recurring activities in the development process of systems of objectives [Albers, Muschik, 2010]. Four main activities were identified, the *generation*, *concretisation*, *monitoring* and *reconciliation* of objectives. An analysis revealed that during each main activity the activities of the problem-solving cycle were traversed.

Conclusions from literature review

Existing models do not succeed to depict the difference between process description i.e. prescription and do not support customising and scaling of processes. Descriptive models usually not entail all important dependencies between activities [Browning, et al., 2006]. A process model needs not to be fully defined by experts, but shall be complemented by the knowledge of the agents resulting from past processes [Rupprecht, 2002]. Most models are built of distinct stages or activities, but there is no common understanding what makes up the distinction between them. Rather design stages are described than design activities [Sim, Duffy, 2003] An objective of a process model should be to align the mental models of the users and to ease communication [Browning, et al., 2006].

3. The iPeM in the context of process modelling

3.1 The model

The integrated Product engineering Model (iPeM) as presented by Albers [Albers, 2010] was developed to describe any specific engineering process from a meta model to application. For modelling uniqueness of product engineering processes and to improve interaction between objectives, operations and created objects, the systemic approach of the ZHO-model ([Ropohl, 2009], section 2.2) represents the model framework. The meta model (see Figure 2) shows the operation system in the centre flanked by the system of objectives and the system of objects. Main elements of the operation

system are activities. They are presented in a combination of a stage-based and activity-based approach. On the vertical axis, the product life cycle is described by the activities of product engineering. The activities of the problem-solving cycle lie along the horizontal axis, transform objectives into objects and together with the vertical axis span a matrix of activities representing the structural concept of the engineering process. Activities are not fixed concerning points of start and end. Each activity is seen as specific type of information necessary for the achievement of design goals. The sequence in which they occur in the actual process is dependent on the specific project. They run in parallel or iteratively.
The *meta model* contains the main building blocks for the derivation of more specific *reference models* which describe patterns of successful processes. They constitute the basis for formulating *implementation models* for specific projects. *Application models* monitor the actual course of the process for deriving information about improvement potentials for reference models. The visualisation of processes is shown in the *phase model* of the operation system.

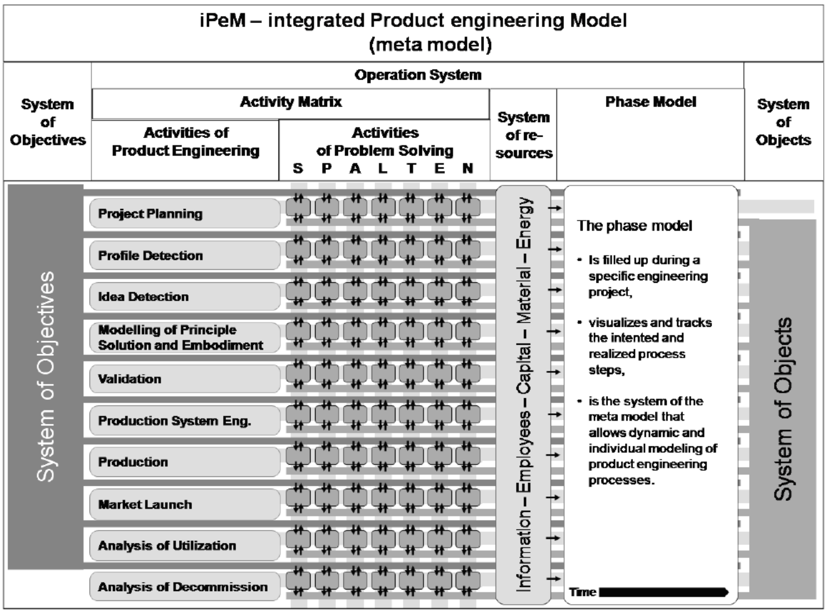

Figure 2. The meta model of the integrated Product engineering Model [Albers, 2010]

3.2 Abstraction levels of the iPeM

The following paragraphs discuss the classification of the iPeM in its abstraction levels to investigate the course of the transformation from the meta model to an implementation model and the interfaces between the model levels.
The different model levels described in the section above form the abstraction levels concerning the formal structure of the model. To describe the abstraction levels concerning the models addressed content, the contentual individuality is introduced as referring to [Rupprecht, 2002]. To define the distinct levels the scope of the iPeM is reviewed, i.e. the application cases in which it is used.

1. *Application in research* to support the investigation of product engineering processes in different domains or for different products, i.e. to be taken as basis for scientific reasoning.
2. *Application in industry* for modelling the processes in a special branch, company and of a special product development.

In the first case (see Figure 3) three contentual abstraction levels are useful. First level represents the *general* view on the iPeM without contentual concretisation. Referencing the iPeM to a specific domain, as for example to the micro specific engineering process as shown in [Albers, 2010] spans the second abstraction level, the *domain-specific* level. The respective processes for a product development can be depicted on the *product-specific* level. For the second application case, the second contentual level could be represented by a *company-specific* view, the third would be applied to a *product-specific* view. This makes it possible to define common company-specific processes on one level and to individualise them on the same basis for different products developed in one company.

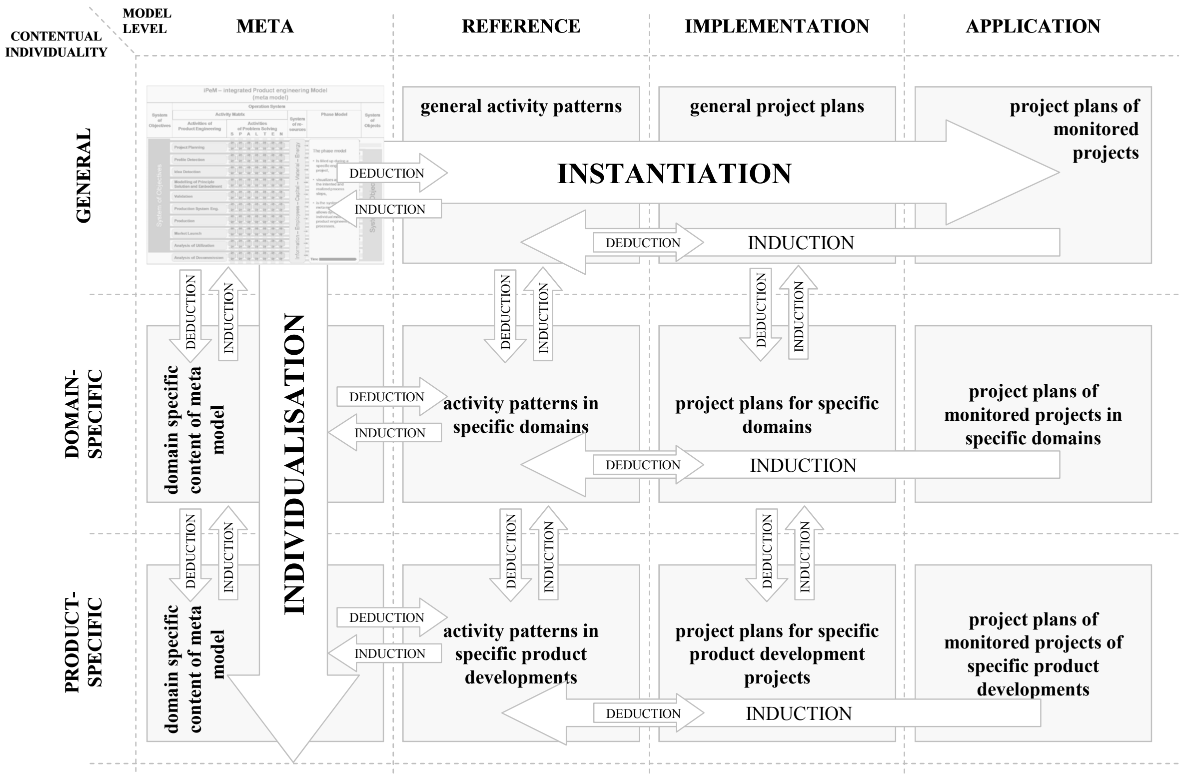

Figure 3. Abstraction levels of the iPeM

Models for different abstraction levels are derived through two main procedures, *induction* and *deduction*. Deduction, the conclusion from the general to the specific is used for the concretisation of formal structure in instantiation, i.e. for the specialisation of the content in individualisation. Induction represents the use of knowledge from specific levels to build generalisations on superior abstraction levels. The exception is the interface between reference, implementation and application level. Application models reproduce the course of an actual process on the same abstraction level as implementation models. Their insights are conveyed inductively to reference models for an improvement of patterns.

3.3 The understanding of activities

To find out how the actual derivation of models on the different abstraction levels is in fact done, it is necessary to go into more detail on how the iPeM is actually built. The fundamental mental models are systems thinking, macro- and micro-logic representing the framework for modelling processes.

Activities as systems

Extending the systemic thought of the iPeM, activities are a part of the operation system, which itself is part of the system of product engineering. Consequently, an activity must be a subelement to the operation system. As such, it is part of the hierarchical structure of the operation system. As could be seen in section 2.2, each activity transforms an input into an output, according to the functional concept of a system. This output is itself input to another activity, relating the activities and forming a structure of activities. As an activity can be depicted with each different system concept, an activity of the iPeM can be modelled as system, i.e. subsystem to the operation system.

Decomposition of activities

The stage-based activities of the meta model of the iPeM are derived from the product life cycle, the macro-logic, linked to the micro-logic, the problem-solving cycle. In the previous chapter was stated that an instantiation of the meta model is a concretisation of its formal structure. As presented in the previous paragraph, this structure is built of related subsystems.

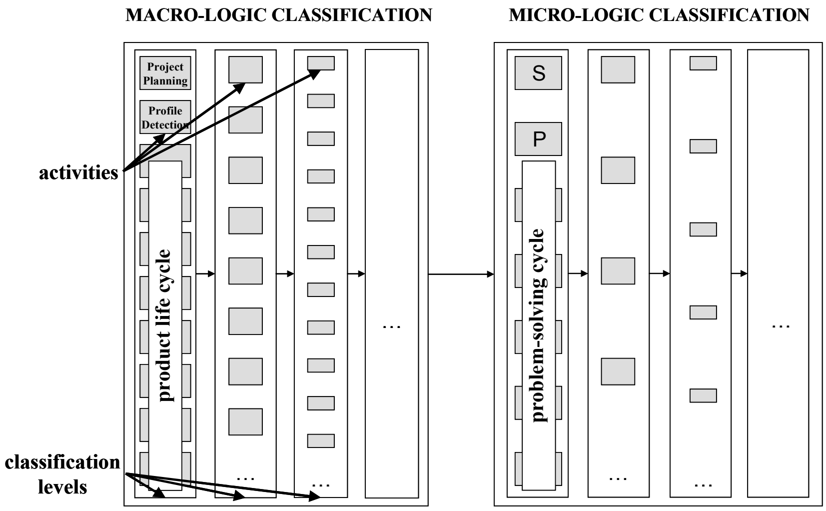

Figure 4. Classification of activity abstraction levels

As a result, instantiating the meta model would mean a concretisation of the abstraction levels of its subsystems, i.e. the activities. According to Hubka and Eder (cp. section 2.2) an activity is decomposable into different abstraction levels. The abstraction extent needs to be suitable to the prominent problem. It has to be possible to derive models from the meta model with an abstraction level according to the relevant problem, i.e. a meta model has to provide the formal structure for modelling different abstraction levels of activities.

Assuming that each activity on the macro-logic level, i.e. the micro-logic level is a system, then each activity consists of subsystems, which exist of subsystems etc. For the macro-logic, these subactivities concretise the description of the product life cycle while subactivities of micro-logic activities concretise the problem-solving steps. Different classification levels result (see Figure 4), whereas each activity on a higher abstraction level contains all subactivites in lower levels and is itself entailed in all activities on superior levels. Every macro-logic subactivity also contains the activities in the problem-solving activity hierarchy. Classification levels are not always clearly distinguishable due to different granularities in abstraction and depend on the specific application case.

Abstraction of activities

The classification levels build the framework for modelling the iPeM on different abstraction levels. The meta model implements them as a set of activities (see Figure 5). These provide the formal structure for deriving each kind of reference model. The activity patterns are constituted by the activities from the meta model, which are selected, linked to each other and brought into temporal context in the phase model. Also attributes are assigned. The respective activities are chosen from the abstraction (classification) level which is most suitable for describing the pattern. With this flexibility it is possible to model the specifity of the different activities in the product life cycle, e.g. early vs. detailed design activities. Activity patterns represent homogenous activity classes, also understood as processes (cp. to section 2.2) since activities form the smallest building blocks of processes.

Specific implementation models can be derived from reference models by the adaptation of the actual occurrence of certain activities in a specific project, their sequence and the temporal relations between activities as well as their attributes. Inductively, information from for example the application of a certain implementation model can be transferred to alter the activity patterns in the reference model or to complement the set of activities (subactivities) of the meta model. The same procedure applies for the change in contentual abstraction levels for the individualisation of a model.

Advantages for modelling product engineering processes

The set of activities in the meta model is dynamic, i.e. it develops according to the knowledge of product engineering processes. Therefore it is never completed and an information transfer about occurring activities, their attributes and relations can be ensured to keep the model complying with changing constraints in continuously developing product engineering processes.

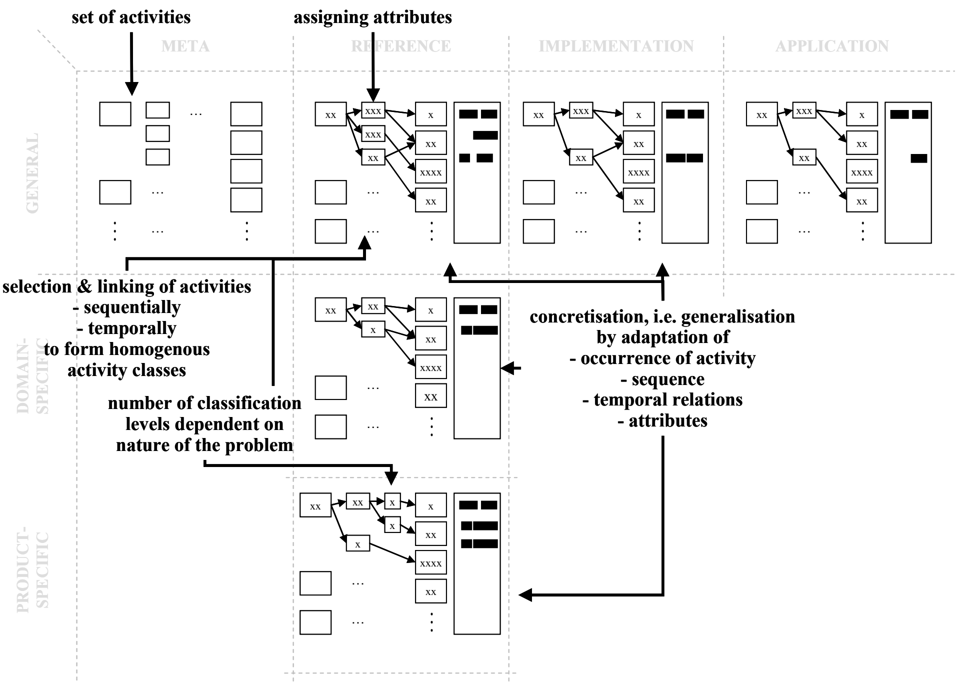

Figure 5. Using activities for modelling different levels of abstraction

By linking the abstraction levels, changes in activities are always provided for each modelling level, which helps standardising the modelling procedures. With the set of activities being adaptable to different abstraction levels the iPeM is able to model all kinds of design situations. Further advantages lie in the possibility to assign knowledge directly to the design situation in which it was generated for example for reuse in other product developments. Tools and methods can be easily matched to activities in the respective context and best practices can be assigned and exchanged when being inductively generalised to a higher abstraction level.

4. Application on the development process of systems of objectives

The findings discussed in chapter 3 were exemplarily applied to results of the process analysis conducted in the study of systems of objectives in early activities of vehicle engineering processes [Albers, Muschik, 2010]. The aim was to substantiate the ideas developed above by deriving a depiction of an activity pattern for the activity *derivation of an objective frame.* This is a subactivity to the activity *generation of objectives.* External boundary conditions are processed to be integrated in the derivation of technical objectives. In this implementation case the abilities of the extended iPeM approach to define and structure activities and the actual support for deriving activity patterns and process models was to be investigated using the project planning tool MS Project.

Layout of classification levels

At first the activities conducted during the derivation of an objective frame as identified in the process analysis were classified in respect to the vehicle life cycle activities (see Figure 6, cp. to Figure 4). The definition of the classification levels follows the scope of the problem to be depicted. Therefore the main product life cycle activities are concretised into the main activities describing the development of systems of objectives (cp. to [Albers, Muschik, 2010]). These are further specified to be able to model the activities necessary for deriving an objective frame (highlighted in Figure 6).

Derivation of an activity pattern for an objective frame derivation

The classification of activities was used to build an activity pattern for the derivation of an objective frame (Figure 7). The structure given by the classification levels was implemented in MS Project. Product life cycle activities were linked depending on their temporal sequence and the respective problem solving steps were matched. Resources were assigned according to former experiences in deriving objective frames. The actual project plan was derived on the basis of this activity pattern.

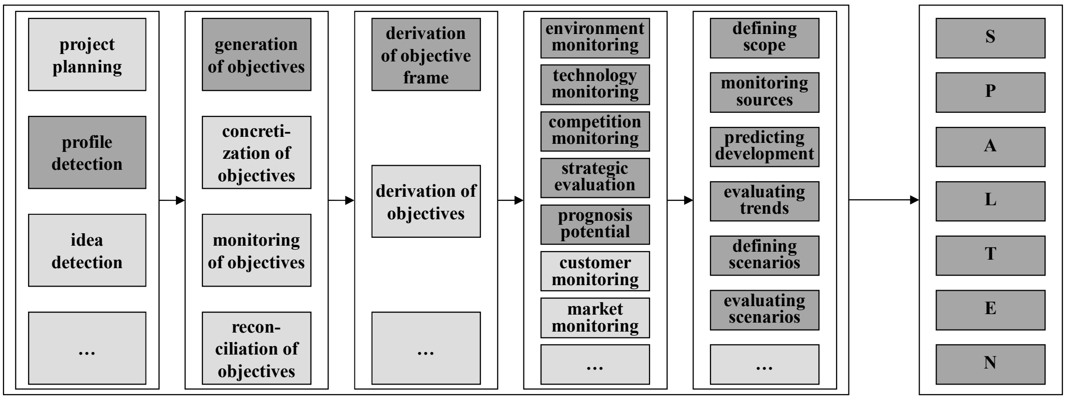

Figure 6. Possible classification of activities for the derivation of an objective frame

The project timeline and activities were specified according to the project boundary conditions, e.g. the activity *monitoring of sources* for environmental aspects was not incorporated in the plan since it had already been done before project start. Subsequently the course of the project was monitored, results were displayed in the application model. One insight gained for improving the activity pattern was that more steps of the problem-solving cycle as originally planned were necessary.

The application showed that the iPeM approach with the possibility to define activities on problem-specific classification levels is suitable to describe and structure real world processes on operative work level. It helps to avoid missing out on certain work steps during problem solving, which are often not included in project plans despite their influence on development cost and time. Planning of processes is facilitated since an activity pattern needs only to be defined once and little effort is necessary to specify it for a certain project. Improvement of process conductions can be ensured by conveying insights gained inductively into the basic reference model. However the building of the models using MS Project still requires effort and a uniform definition and naming of activities turned out to be difficult. This implementation case is one of several studies in which the approach is being explored regarding its potentials and limits. The requirements for providing highest possible applicability are collected and evaluated to be integrated into the approach.

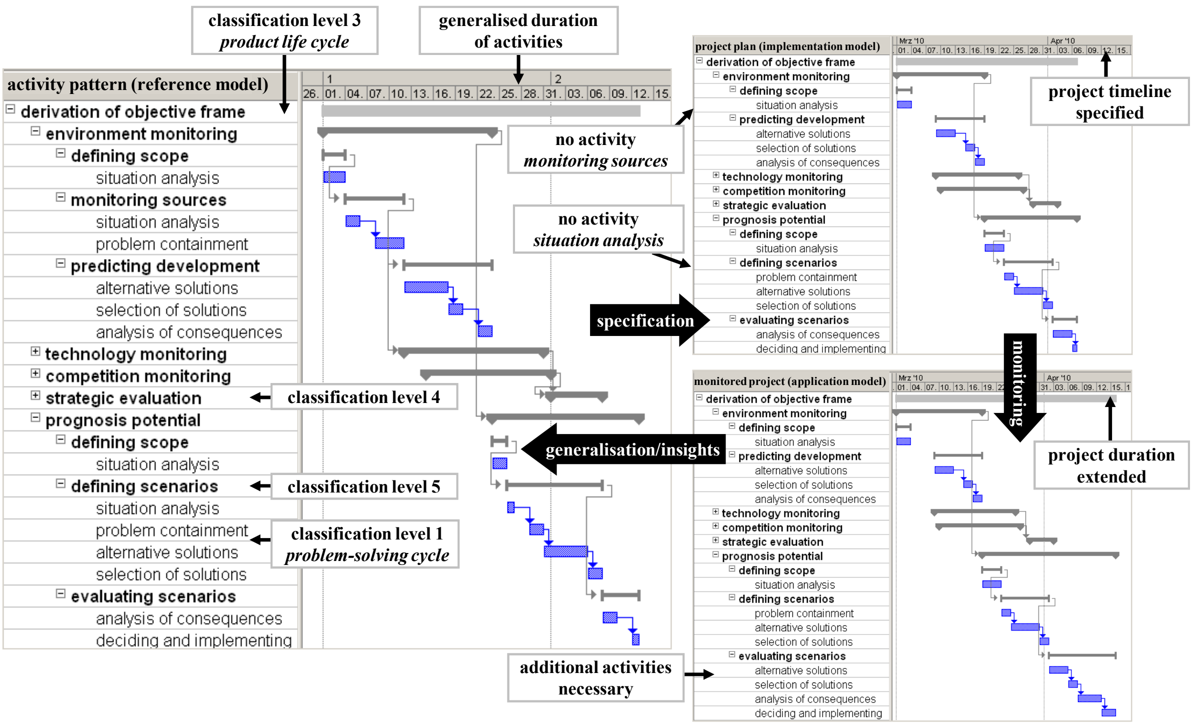

Figure 7. Example for the generation of an activity pattern for an objective frame derivation

5. Conclusion and future work

This paper has discussed the idea of using activities on different abstraction levels for modelling the transformation of meta models to implementation models applicable for practical use. On the basis of a thorough analysis of the art of modelling and current approaches the classification of the iPeM in its different abstraction levels has been shown. This formed the frame for an extended understanding and definition of activities on the basis of the existing systemic approach of the meta model. We have shown that it is possible to describe the gap between main product life cycle activities and detailed process steps by different abstraction levels of activities. Without these levels the prominent problem cannot be sufficiently addressed. With this knowledge, we proposed the definition of a classification of activities to depict the different degrees of abstraction. We find that the detection and formulation of activities results either deductively or by experience from operational implementation and evaluation inductively. This enables not only researchers or management to use this model for investigation or planning, but also the agents to integrate their knowledge gained from activity executions. The set of activities provided for the derivation of models is subject to continuous review, adaptable to changing constraints and supports reduction of redundancies in the formulation of activities, i.e. supports unification in the modelling approach. A first application supported the potentials of the approach and proved the ability to model the interface to the system of objectives.

Subject to further investigation are the content of the product life cycle and problem-solving activities and description with the help of respective activity abstraction levels. The relations between activities are to be looked at closely. The use of the approach idea for relating the product life cycle activities to a special context (e.g. system of objectives) shall be evaluated. The depiction and link to conventional business processes is to be examined. One main research field is the development of an ontology for activities, which shall support the matching of knowledge to activities. A unification of activity patterns by the use of tools and methods which are directly assigned to their relevant activity and exchange of best practice experiences on higher abstraction levels is to be investigated, also in respect to the use with activities of development of systems of objectives. The approach will be fully validated, which will be achieved by the integration in current projects. The presented ideas contribute to the overall objective of the iPeM to support the product engineering process as a model from a meta level to application in practice.

References

Albers, A., "Five Hypotheses about Engineering Processes and their Consequences", International Symposium Series on Tools and Methods of Competitive Engineering - TMCE 2010, Ancona, Italy, 2010

Albers, A., Muschik, S., "Development of Systems of Objectives in Early Activities of Product Development Processes", International Symposium Series on Tools and Methods of Competitive Engineering - TMCE 2010, Ancona, Italy, 2010

Browning, T. R., Fricke, E., Negele, H., "Key Concepts in Modeling Product Development Processes", Systems Engineering, 9, 2006, pp. 104-128

Clarkson, J., Eckert, C., "Design Process Improvement", Springer-Verlag, London, 2005

Hubka, V., Eder, W. E., "Design Science: Introduction to needs, scope and organization of engineering design knowledge", Springer-Verlag, Berlin, 1996

Krehmer, H., et al., "Coping with Multidisciplinary Product Development - A Process Model Approach", International Conference on Engineering Design, ICED ´09, Stanford, USA, 2009

Ropohl, G., "Allgemeine Technologie - eine Systemtheorie der Technik", 2009

Rupprecht, C., "Ein Konzept zur projektspezifischen Individualisierung von Prozessmodellen", Dissertation, Karlsruhe, 2002

Sim, S. K., Duffy, A. H. B., "Towards an Ontology of Generic Engineering Activities", Research in Engineering Design, 14, 2003, pp. 200-223

Prof. Dr.-Ing. Dr. h.c. Albert Albers
Karlsruhe Institute of Technology (KIT), IPEK –
Institute of Product EngineeringKaiserstr. 10, D-76131 Karlsruhe, Telephone:: +49-(0)721-6082371
Email: albers@ipek.uni-karlsruhe.de
URL: http://www.ipek.uni-karlsruhe.de