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

DESIGN 2010

CLASSIFICATION AND SYNTHESIS OF DESIGN THEORIES

C. Geis and H. Birkhofer

Keywords: design, theories, design models, methods

1. Motivation

Models and methodologies for designing are numerous nowadays and most universities, their institutes or design schools prefer – or feel committed to – certain kinds of theories, models, and methodologies respectively. No holistic approach which combines all models exists so far and so interactions and interfaces are not (or at least rarely) defined between them. Nonetheless, all of these models were developed with specific intentions which justify their existence, as they portray design from different perspectives and focus on varying elements or actions in designing. Some models were developed to research designing itself and thus improve the understanding of designing (descriptive models). Other models evolved to support designers in methodical work, giving them exact – or at least as exact as possible – instructions on how to proceed in the design process (prescriptive models).
Nevertheless, users cannot cope very easily with this huge variety of models and paradigms which is even hardly manageable by researchers that aren't well-versed in the subject matter. As a result, methods are often deployed in a kind of routine application without questioning the benefits and effects of the execution of methods. Therefore, we analyzed various established and well-known models of designing ("design models", "methodologies" and "design theories" are used synonymously in the following) to see what differences and potentials these models contain and how they all could possibly be combined.

2. Potentials and shortcomings of methods

Grabowski and Geiger found out that in an industrial daily job routine, a lot of methods are used only occasionally or not at all [Grabowski and Geiger, 1997]. The most popular methods are creativity techniques, methods to analyze markets and costs as well as methods to compare product solutions. Additionally, inter-departmental cooperation within the companies and/or cooperation with other companies or universities are often lacking. Customer orientation and social competence were shown to bear great potentials in – particularly the German – industry as well. Methods should support processes and help to reduce iterative loops. Visualization and access to existing knowledge should also be supported by documentation of knowledge and information. Decision making should be improved and methods should assist in reaching targeted cost and deadlines and thus lead to savings of time and cost.
Supporting the findings of Grabowski and Geiger, Reinicke [Reinicke, 2004] claimed that methods are too complex and lead to a theoretical overload. Users may only see the great effort required for deployment and not the advantages and benefits of the methods. Another problem refers to the implementation of methods in companies: they are often chosen or conducted the wrong way or the designers don't have sufficient capacity or support for carrying out these methods.
Geis et al [Geis, et al., 2008]) conducted an interview study in 2007 looking for successful behavior and actions in product design. In these interviews the participants were asked to formulate their

demands and wishes in regard to methods. Most participants stated that one big issue is that methods should support interaction (communication, presentation, documentation of results, etc.) and information management. Additionally, methods should assist in planning, analyzing and organizing the design process, granting appropriate and sustainable actions and measures. Usage of methods was another important item: methods should be flexible and simple with a focus on the output. They should be better integrated in the process and be adapted according to the wishes of the users.
So, beside the potentials and advantages methods have in information and process management, interaction and sustainability, users demand easy-to-use methods adapted to their wishes. Ironically, this is already possible with most methods – but not known to many designers who lack methodical competence. Users often know their processes and required methods, but are not fully aware of their benefits and potentials and/or other methods, which could be used comparatively easy instead. Various existing process models and design theories don't even simplify the understanding of designing and methodical work. So in the following, existing design theories will be briefly presented and consequentially, a synthesis of all models will be attempted.

3. Designing - theories and models

Prescriptive design models and design theories respectively have been developed in particular in Germany for many decades. These models represent guidelines for potential users to design in a methodical fashion.
One of the best-known guidelines for designing is the "VDI 2221" (Systematic Approach to the Design of Technical Systems and Products) as shown in Figure 1 [VDI2221, 1987]. This guideline classifies the design process into four main phases (clarification of the task, conceptual design, embodiment design and detail design). It gives an overview of results and tasks of these phases and what must be done to complete these phases. In matrices the VDI 2221 also gives an overview of relevant methods, their aim and the stage at which they can be applied.

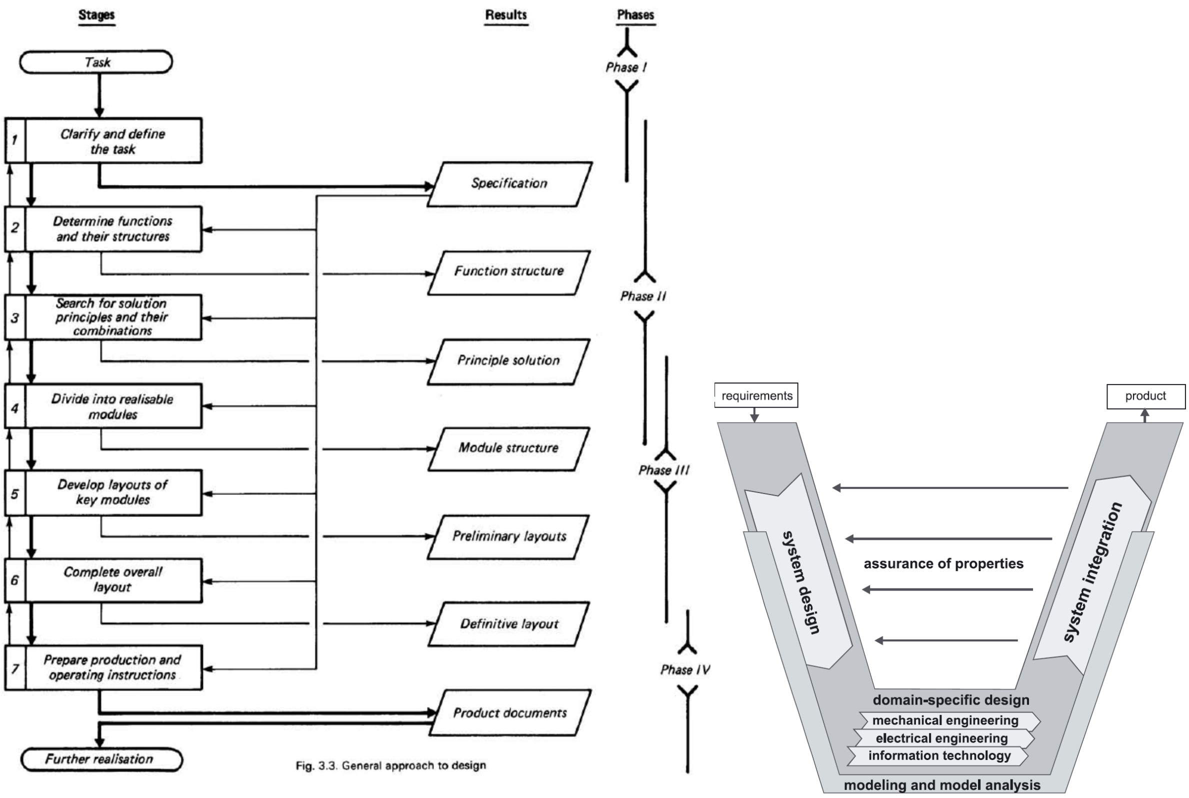

Figure 1. Procedure according to VDI 2221 (left) and VDI 2206 (right)

Similar to the VDI 2221, the VDI 2206 (design methodology for mechatronic systems [VDI2206, 2004]) gives an overview of problem solving as a kind of micro-process, but offers a V-model as an integrated macro-process, in which the different disciplines mechanical engineering, electrical engineering and information technology are combined. On the left wing the system design is parsed, while on the right wing the system integration takes place. Between the wings, the respective attributes/properties are validated. Both models have a highly iterative character.

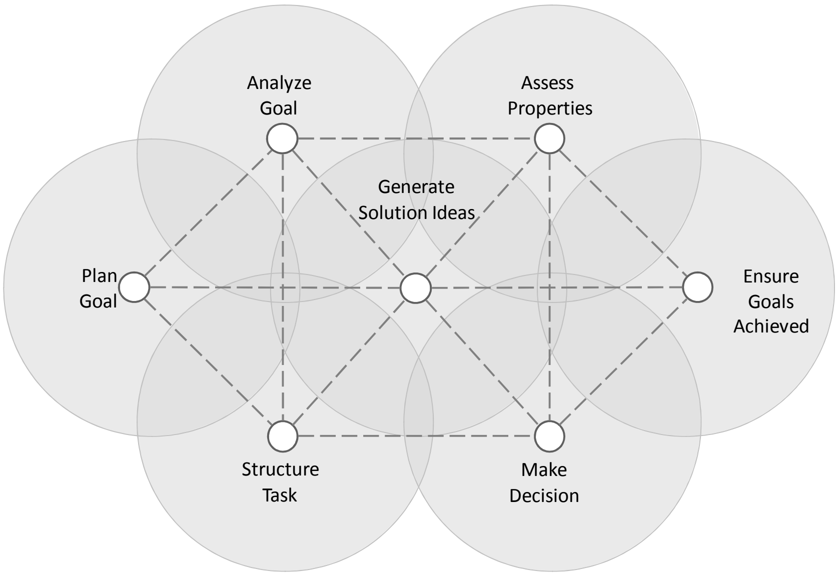

Figure 2. Procedure according to the MVM (Münchener Vorgehensmodell)

As another prescriptive model to plan design processes as an orientation tool within the process and to ensure analysis in/of projects, Lindemann et al developed the MVM (Münchener Vorgehensmodell, depicted in Figure 2) [Lindemann, 2007]. The MVM is based on three basic steps: goal clarification, generation of solutions and making decisions. Furthermore, the seven elements goal planning, goal analysis, structuring of problems, determination of solutions, determination of properties, decision making and assurance of target achievement can be addressed in a suggested standard procedure or in a flexible mode.

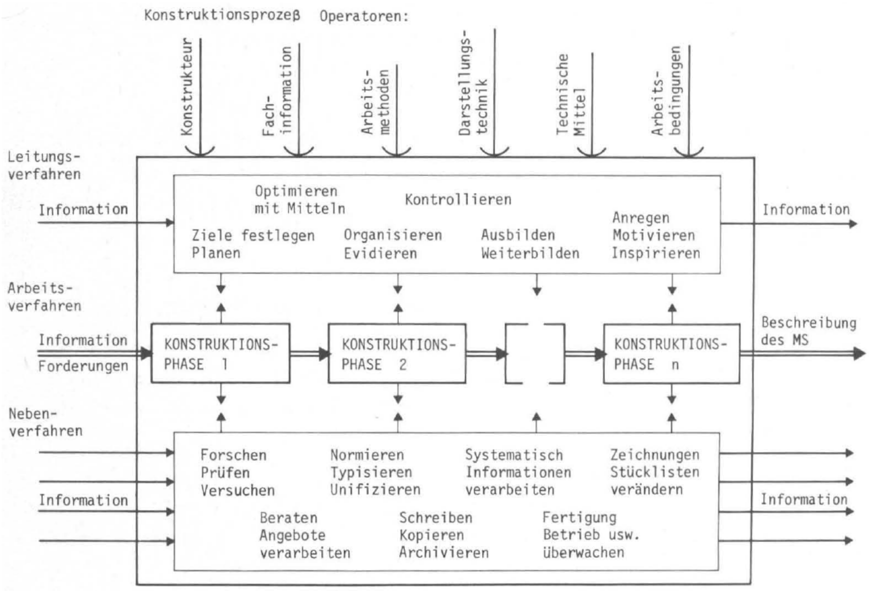

Figure 3. Procedure according to Hubka [1976]

Hubka [1976] states that the design process is generally structured into three stages that move from abstractness to concreteness (conceptual design→embodiment design→detail design). In this process the information »requirements« is processed into descriptions of producible systems according to

Hubka. He describes the construction process in a very abstract non-sequential way without any precise standard operating procedures. Instead, the main route includes the most important tasks and phases of the process, whilst other operations (such as "develop", "detail", "search for solutions", "assess", etc.) support additional important tasks.
To support method deployment, Braun et al developed the Münchener Methoden Modell (MMM) [Lindemann, 2007]. As a prescriptive model, the MMM explains the necessary actions for the phases of method deployment: clarification of method deployment, choice of methods, adaptation and execution of methods and provides advice regarding which elements and attributes are relevant in these corresponding stages.
Similar to Braun,[Reinicke, 2004] developed a system to prepare, analyze, adapt and execute methods and evaluate the deployment afterwards. Her main focus lies in the consideration of customers, so she recommends the identification of primary, secondary and tertiary users and consequentially the determination and comparison of customers' attributes. Based on this the identification of weaknesses and potentials of solutions takes place, which leads to measures and to adaptations of methods.

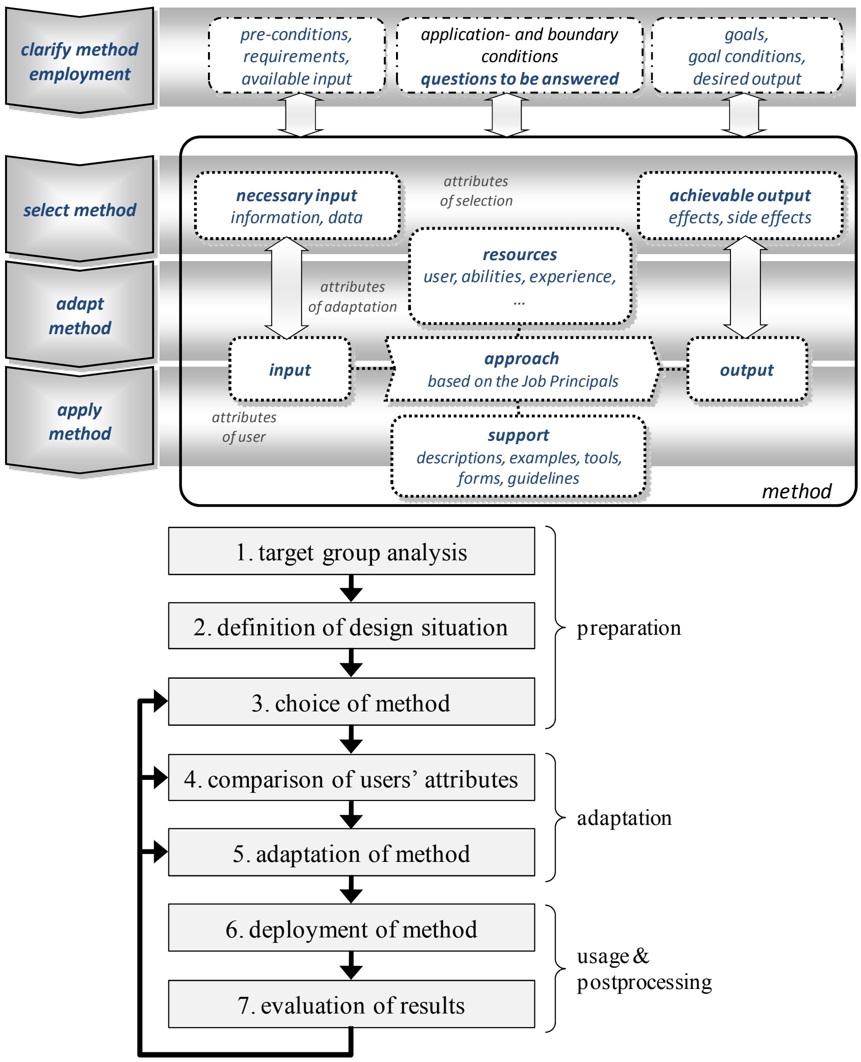

Figure 4. Procedure of method deployment according to the MMM (up) and to Reinicke (down)

Based on an extensive interview study, success-relevant behavior was structured by the BEMAP-project (an interdisciplinary project of psychologists and designers) and clustered by Geis, Birkhofer and Badke-Schaub [Geis, et al., 2008]. Similar to Braun and Reinicke, clarification, choice, adaptation and execution were elementary items of method deployment. But information management is also an important topic when designing.

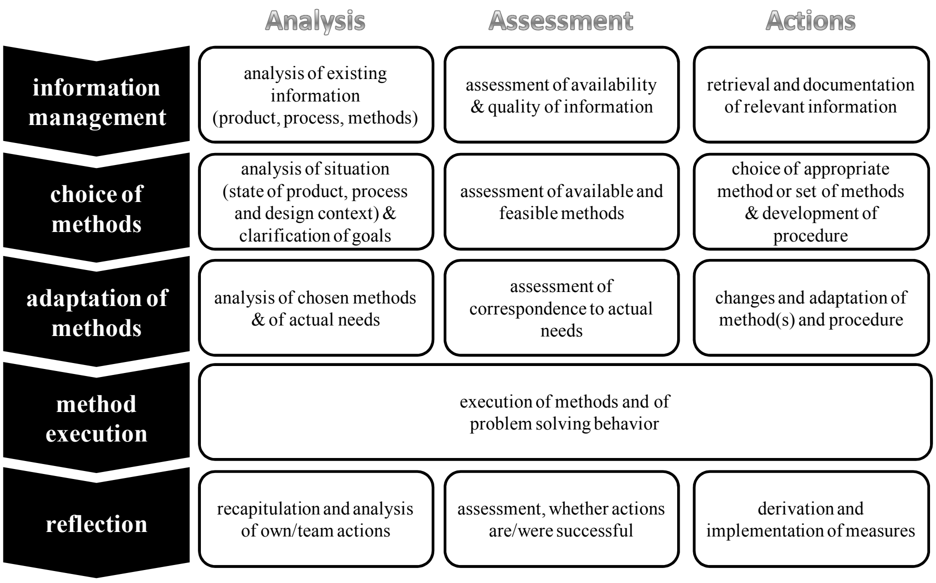

Figure 5. SAM-model

Therefore, based on literature and the Behavioral Marker System as a result of the interview study, existing prescriptive models of method deployment were enhanced by adding the components "information management" and "reflection", which are more or less regarded unisonously as important aspects of method deployment in literature. This model is called SAM (situation-appropriate method deployment).

Systems Engineering provides users with a micro-logic that can be understood as an orientation guide for problem solving [Lindemann, 2007]. Situation analysis is followed by goal clarification (and formulation respectively). Hereafter, synthesis can take place and the following assessments and decisions can be made.

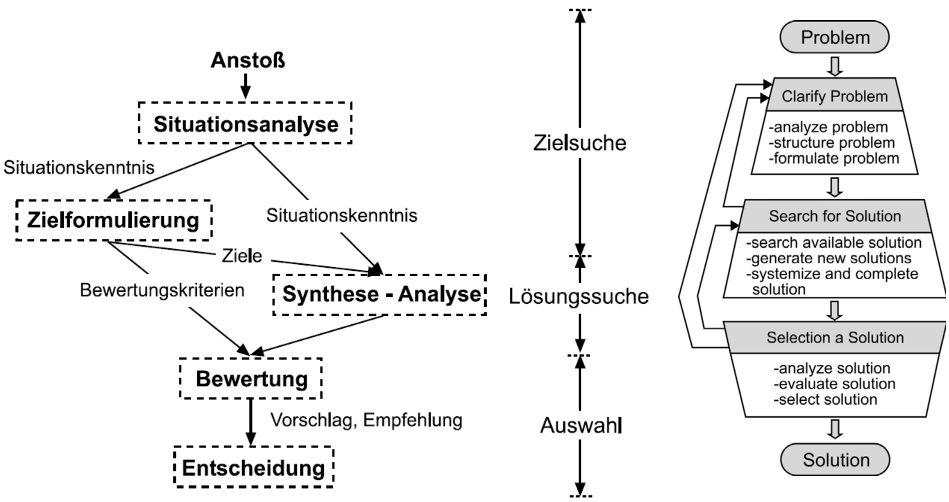

Figure 6. Problem solving according to Daenzer (left) and Ehrlenspiel (right) [Lindemann 2007]

Systems Engineering's three main phases (goal clarification, search for solutions and choice of solutions) are mirrored in Ehrlenspiel's model from 2003, where these stages are subdivided into two or more sub-steps [Lindemann, 2007]. Again (like most models) a sequential cycle is provided with possible iterations if necessary.

Pahl explicitly states that iterative cycles are necessary in designing, but methodical procedure can help to reduce these loops (that are similar to elementary reasoning processes, such as the T.O.T.E.-scheme) [Pahl, 2007]. Pahl's recommended methodical procedure is depicted in Figure 7.

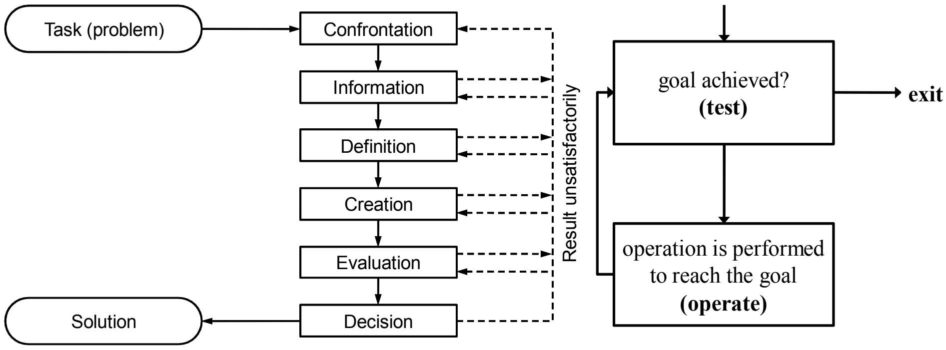

Figure 7. Problem solving according to Pahl (left) and T.O.T.E. (right) [Pahl 2007]

Other – more descriptive – design models focus on reasoning processes and information management. Smithers (Knowledge Level Theory of Designing, K_LD^E) for example characterizes designing as an exploration of what is possible, rather than a search for what is needed [Smithers, 2002]. He identifies basic classes of knowledge: such which is used and such that is created during designing. This knowledge usually has a certain type role and stands in relation between one or more knowledge types (as depicted in Figure 8). With this classification and structuring of knowledge, he wants to make the types of knowledge explicit to users to reach a better understanding of designing.

Knowledge Type Roles		Knowledge Type Relations	
A *Supporting* role	Rol.sp.	*Embedded in*	Rel.em.
A *Constructing* role	Rol.cn	*Supports*	Rel.sp.
A *State maintaing* role	Rol.sm	*Used in* (the construction of)	Rel.uc
		Increments and/or modifies	Rel.im

Figure 8. Definition of Knowledge Type Roles and Relations in K_LD^E [Smithers 2002]

The C-K Theory [Hatchuel and Weil, 2003] also offers a different approach in designing, dealing with the reasoning of designers. Dynamics of design is modeled, seeing “design” as the process by which a concept generates other concepts or is transformed into knowledge.

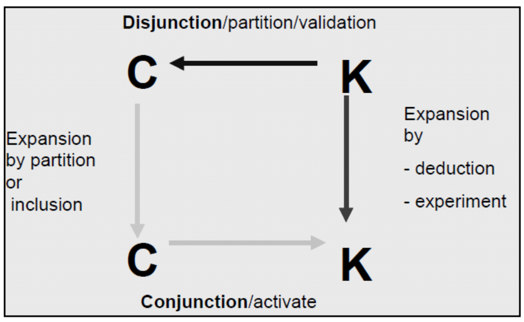

Figure 9. Design square of C-K Theory [Hatchuel and Weil 2003]

Therefore Hatchuel and Weil developed the design square (see Figure 9) consisting of the space of concepts and knowledge including the relevant operators for transformations (for example C→K is equivalent to the validation of concepts, e.g. by tests). The C-K Theory deals with the reasoning of the designer.

Other current rather descriptive approaches view designing from a more mathematical standpoint and focus on features and/or attributes of design and the transformation of these attributes. They formulate basic axioms and develop algorithmic models based on them. An example for these kinds of models is the “General Design Theory”, developed by Yoshikawa in 1980. GDT is intended to be a universal

design theory [Tomiyama and Schotborgh, 2007]. Physical entities are determined and categorized according to their attitudes and functions. As result of set operations, a region within this set of entities can be identified that contains design solutions. Depending on the number of design specifications and the character of the entities, the solution space can decrease.
Nam P. Suh developed “Axiomatic Design”. Herein, he set up two axioms for designing: the independence axiom (meaning that all functional requirements are independent) and the information axiom, stating that the best design is a functionally decoupled design with minimum information content [Suh, 1998]. Every entity and/or its representations can be described by three other axioms: the axioms of recognition, correspondence and operation. Suh sees design as mapping (transformation) between functional requirements (FRs) and design parameters (DPs). This concept of mapping can also be extended to other domains (as depicted in Figure 10.).

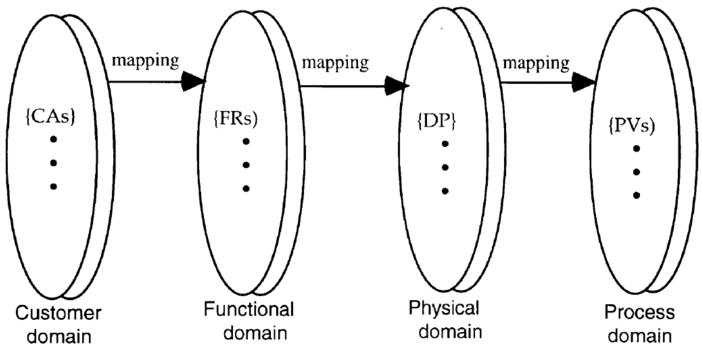

Figure 10. The four domains of the design world acc. to [Suh 1998]

4. Classification and synthesis of models

So models which support designers are necessary and many different models or theories are provided to researchers and potential users.
The difference between procedural models such as the VDI 2221 and problem solving routines (such as the Systems Engineering approach) can be shown by distinguishing these models into strategic and operative models. While e.g. the VDI or the MVM give advice regarding the procedure, the problem solving models and theories deal with more elementary actions necessary in every process stage. The models of reasoning support operative procedures as they deal with cognitive and elementary aspects of designing. Solution generation (as part of Ehrlenspiel's model) for example is a methodical process of knowledge processing that can be described by the reasoning theories. The algorithmic models however take on a more abstract view on the process when designing systems. They describe how subsequent phases and solutions are based on earlier stages and requirements and how requirements and solutions are related to each other.
As companies often provide fixed design procedures, designers need support in using and executing the methods themselves. Therefore, we introduce a further category named “tactical procedure“. We suggest a distinction of the models in five categories that support and partially depend on each other

Algorithmic Models

Algorithmic models are based on axiomatic models and provide an abstract view on the process that describes design states with entities and sees design as the transformation of parameters and thus of state of product and process. Axiomatic Design and General Design Theory (GDT) are examples for these kinds of models.

Strategic models

Strategic Models help to develop the design process and lead through it. Strategic models offer general approaches to design and suggest certain phases and results that should arise from these phases. These models have a highly iterative character. Strategic models are for example the VDI 2221 or the MVM.

Tactical models

Following the strategical decision regarding processes and their outcomes, method deployment has to be thoroughly analyzed and planned. On a tactical level, situations, processes, methods and results are analyzed and assessed. Necessary information is retrieved, appropriate methods are chosen and method deployment is reflected upon. The MMM, Hubka's and Reinicke's model as well as SAM are tactical models.

Operative models

After the procedure is established and the method deployment is defined, steps and elements of problem solving have to take place which can be described by operative models. Important tasks like clarifying problems and tasks, solution generation, solution analysis, etc. take place within almost every phase in the process. Therefore, operative models were developed for example by Daenzer, Ehrlenspiel and Pahl.

Models of reasoning

Models of reasoning depict how designers think and how they use and treat knowledge to develop new knowledge, ideas, concepts and/or solutions. These models describe what can be called micro-logic – actions and elements that are necessary for successful problem solving. The C-K theory as well as K_LD^E are models of reasoning.

Now that these models have been sorted into five categories, the question must be: in what way do they rely upon each other? Giapoulis [Lindemann 2007] developed a model with three levels: planning of the design project, operative planning and the result level. The necessary actions are carried out "zigzagging" between these levels. This model can now be extended by inserting relevant new levels, or layers respectively, resulting from the differentiation mentioned before. Regarding the fact that all leaps between the levels are transformations of entities (which can be described by algorithmic models) we can integrate all the different kinds of design theories into one model.

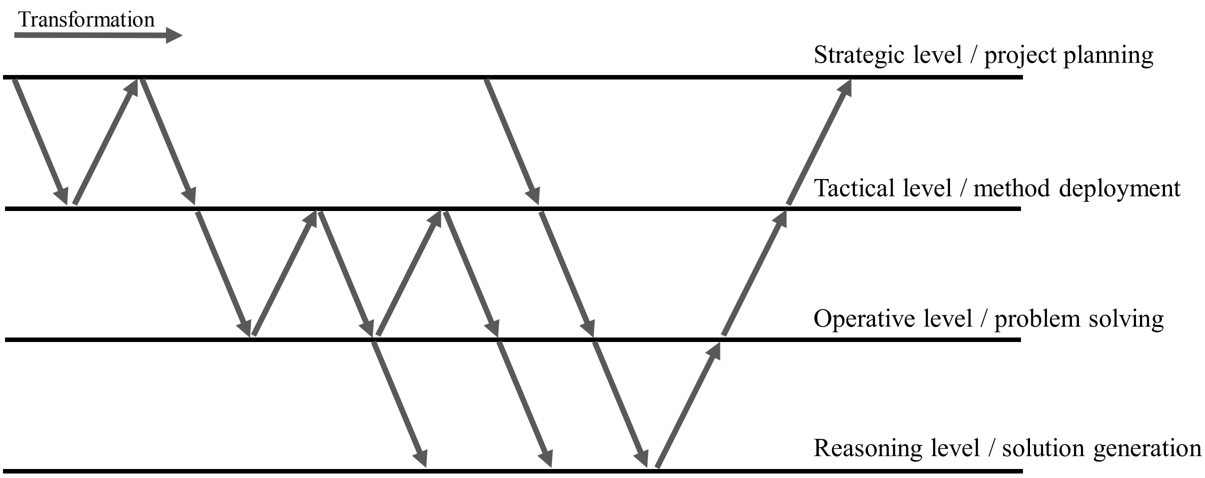

Figure 11. Zigzag-model of designing

Different behavior and actions are necessary on all levels. On the reasoning level for example, solutions and results are generated that can be analyzed and integrated into a new method deployment (flyback) or used as inputs on operative or tactical levels again. These different actions are described in the corresponding theories and for each layer some rule of thumb can be given to the user, for example how problem solving should look like on an operative level (e.g. the problem solving cycles according to Pahl or Ehrlenspiel).

A rather simple and short example can show how this works: A user must design a certain product (e.g. a gear box for a winch). He first lays out a time schedule and defines milestones (strategic level). He then decides to analyze the requirements and to compile the specification (tactical level). So he proceeds to the operative level and begins to analyze the task and situation (moments, loads, forerunner products, solutions from other disciplines like mobile hydraulics, market situation,…). This is done by execution of operative and reasoning tasks such as the analysis of customers, application scenarios or markets, the synthesis of results and/or an assessment. Back on a tactical level the user can evaluate these results (this is again running on the operative or even deeper on the reasoning level) and can continue with calculations of transmission, diameter of the cable, etc. The user's procedure continues as described before: He always jumps between the strategical, tactical, operative and reasoning level. Conceptual design for example is defined on a strategical level, the methods and procedure are decided on a tactical level, the tasks are carried out on an operative level and the concepts and solutions are generated on a reasoning level. All these transformations and changes of the entities that describe the solution (or the processing of information according to Hubka [1976]) can be described by algorithms.

No reasoning about solutions can take place without the preceding steps (such as definition of requirements or clarification of situations). Similarly, no operative steps can be taken without tactical actions beforehand and a tactical procedure itself needs strategic decisions and/or goals. So – combining this deductive, sphere-model-like point of view with the zigzag-model of designing – a holistic model of design, which integrates and embeds the theories/models of designing (which describe the tasks in the different levels of designing themselves), could take on a form as depicted in Figure 12.

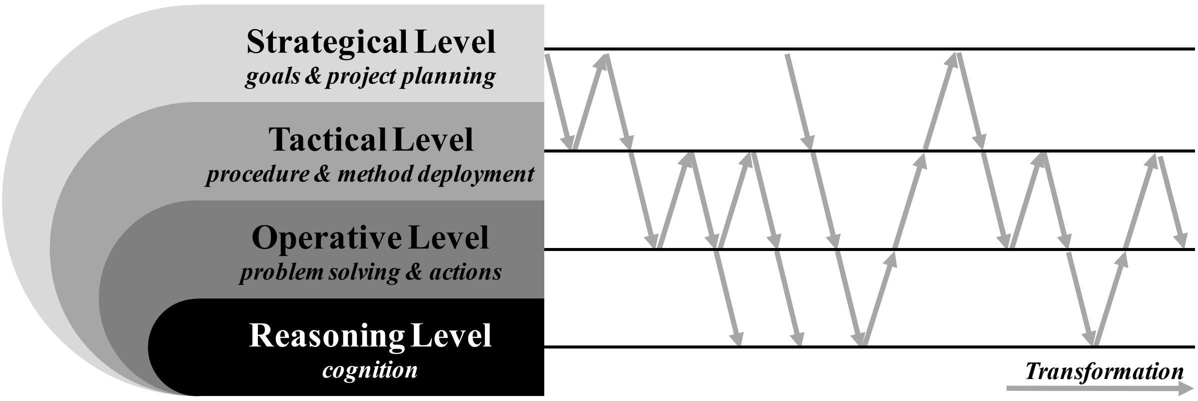

Figure 12. Integrated model of designing

By this classification and description of the models' interrelations, every user can create a kind of roadmap for designing, giving him access to the relevant theories and procedures he uses in certain levels and stages of designing respectively. This holistic way of designing aids users in planning, interacting and adapting to specific needs in their processes.

5. Conclusion

Designers nowadays can use and follow various approaches respectively when willing to adopt a methodical design approach. At first, these models and/or design theories may seem confusing and contradictory to possible users. But when considering the fact that all these models were developed with certain focal points and with certain stances of their authors, the models do not differ significantly from each other. On the contrary, the models approve and support each other and can partially be clustered and synthesized. In our models we combined the five elementary categories of design models: algorithmic models, strategic models, tactical models, operative models and models of reasoning. Algorithmic models describe the state of design entities and the transformation between the parameters of these entities. Strategic models support planning, organization and control of the process while tactical models help to specifically adopt methods to address actual situations and needs. Operative models on the other hand describe the process of problem solving that takes place in the particular stages of method deployment and models of reasoning can help to understand and explain the processes and actions executed there. So our model can support users and enhances understanding.

Furthermore, it delivers a base for additional research with a focus on creating interconnections and interfaces between these proposed levels.

References

Geis, C., Bierhals, R., Schuster, I., Badke-Schaub, P., Birkhofer H., "Methods in Practice - a Study on Requirements for Development and Transfer of Design Methods", Proceedings of the 10th International Design Conference - DESIGN 2008, D. Marjanovic (Ed.), FMENA, Zagreb, 2008, pp. 369-376.

Geis, C., Birkhofer, H., Badke-Schaub, P., "Development of a Sustainable Training Program to Enhance Methodological Competence of Designers", Proceedings the NordDesign 2008, L. Roosimölder (Ed.) Tallinn, 2008, pp. 200-209.

Grabowski, H, Geiger, K., "Neue Wege Zur Produktentwicklung", Raabe Stuttgart, Berlin, Bonn, Budapest, Düsseldorf, Heidelberg, Prag, Sofia, Warschau, Wien, 1997.

Hatchuel, A., Weil B., "New Approach of Innovative Design: An Introdutction to C-K Theory" Proceedings of the 14th International Conference on Engineering Design (ICED 03), K. Gralén and U. Sellgren (Eds.) Stockholm, 2003.

Hubka, V., "Theorie der Konstruktionsprozesse: Analyse der Konstruktionstätigkeit", Springer-Verlag Berlin, New York, 1976.

Lindemann, U., "Methodische Entwicklung Technischer Produkte: Methoden Flexibel und Situationsgerecht Anwenden", Springer Verlag Berlin, Heidelberg, New York, 2007.

Pahl, G., Beitz, W., Feldhusen, J., Grote, K.-H., "Engineering Design - a Systematic Approach", K. Wallace and L. Blessing (Eds.), 3 ed. Springer-Verlag London, UK, 2007.

Reinicke, T., "Möglichkeiten Und Grenzen Der Nutzerintegration in Der Produktentwicklung: Eine Systematik Zur Anpassung Von Methoden Zur Nutzerintegration", Verlag Dr. Hut Berlin, 2004.

Smithers, T., "On Knowledge Level Theories and the Knowledge Management of Designing", Proceedings of the 7th International Design Conference DESIGN 2002, D. Marjanovic (Ed.), FMENA, Zagreb, 2002, pp. 87-91.

Suh, N. P., "Axiomatic Design Theory for Systems" Research in Engineering Design, No. 10, 1998, pp. 189-209.

Tomiyama, T., Schotborgh, W. O., "Yet Another Model of Design Synthesis" Proceedings of the 16th International Conference on Engineering Design 2007 - ICED 07, Paris, 2007.

VDI2206, "Design Methodology for Mechatronic Systems", Beuth Verlag GmbH (Ed.), Verein Deutsche Ingenieure Düsseldorf, 2004.

VDI2221, "Systematic Approach to the Design of Technical Systems and Products", Beuth Verlag GmbH (Ed.), Translation of the German edition 11/1986, VDI Verlag GmbH Düsseldorf, 1987.

Dipl.-Ing. Christian Geis
Research associate
Technische Universität Darmstadt
Department product design and machine elements Darmstadt (pmd)
Magdalenenstrasse 4, 64289 Darmstadt, Germany
Telephone: +49 6151 163055
Telefax: +49 6151 163355
Email: geis@pmd.tu-darmstadt.de
URL: http://www.pmd.tu-darmstadt.de