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

DESIGN 2008

A REVIEW OF THE FUNDAMENTALS OF SYSTEMATIC ENGINEERING DESIGN PROCESS MODELS

D. Motte

Keywords: systematic engineering design process model

1. Introduction

Systematic engineering design process models form the mainstream of prescriptive engineering design procedures. The work of their foremost representatives, [Pahl & Beitz 2007], is cited in 20-25 % and 10-20% of the papers of the last ICED and Design Conferences respectively. In an overall perspective, systematic engineering design process models constitute the key process upon which current product development models are built. The structure that these systematic models give to the engineering design process and their inherent logic explain in part their success. Along the way, however, several limitations have been noticed, triggered by the fact that the models have been slowly adopted by industry and practitioners. This low integration of the models has led to a new research focus towards the adaptation of systematic engineering design process models regarding 1) the engineering designer's way of thinking and working and 2) the industrial context. This predominantly empirical research, inspired by cognitive psychology on one hand and sociology on the other hand, has resulted in new descriptive models that have challenged the more prescriptive ones.
This paper tries to go one step further, and takes a slightly different approach by directly questioning some of the fundamentals of systematic engineering design process models. It also indicates alternative, flexible ways of conceiving the engineering design process.

2. The fundamentals of systematic engineering design process models

The engineering designer has to cope with a wide range of activities within a wide range of circumstances: [Pahl & Beitz 1996] classifies the set of engineering design tasks with no less than 28 characteristics regrouped in 7 types (Origin of the task, Organisation, Novelty, Batch size, Branch, Complexity, Goals; see pp. 2-6). In order to deal with such complexity, it is necessary to have a more structured way of working than the sole use of intuition. To that end, engineering design process models and various methods have been developedprogressively, especially after World War II. (A review of the early works can be found in [Bender 2004], [Bjärnemo 1983], [Hubka & Eder 1996], and [Pahl & Beitz 2007].)
Gradually, the different systematic engineering design process models have adopted a common ground and they differ only in peripheral variations. The models referred to in this paper are mainly [Hubka & Eder 1996], the different editions of [Pahl & Beitz 2007][1], [Ullman 1997], and [Ulrich & Eppinger 2007].[2]

[1] The 1st and 3rd German editions (1977, 1993) and the 2nd and 3rd English editions (1996, 2007) were consulted. The last English edition has been used for citation unless otherwise specified.
[2] "Systematic" (resp. "systematic approach") has been employed for translating the German "methodisch" (resp. "Methodik"); see for example the translation of the VDI 2221 title, thus emphasizing only one of the characteristics of those engineering design process models.

The following assumptions behind the organisation of various systematic engineering design process models (see also [Hubka & Eder 1996], [Pahl & Beitz 2007], and [Bender 2004] who reviewed them) are described below.

A systematic engineering design process model aims at making it easier to find an optimal design for a product-to-be. To that end it is necessary to encompass the broadest range of solutions, that is, to search for solutions in a structured, systematic way. The breadth-first top-down strategy is adopted, which means first finding the largest possible number of abstract solutions (breadth-first) and then more concrete ones (top-down). The different concretization levels are based on product models whose archetype is Hubka's technical system (TS) model (see below). The reasons are that each solution on an abstract level represents a set of different concrete solutions, and that it is more costly and time-consuming to develop, describe and evaluate concrete solutions than abstract ones.

The TS model developed by Hubka [Hubka & Eder 1996, p. 106] consists in (from abstract to concrete) a) an overall function decomposed in b) a function structure, c) an organ structure — "working principle structure" in [Pahl & Beitz 2007], or "concept"[3] in [Ullman 1997], [Ulrich & Eppinger 2007] — and d) a component (constructional) structure. The overall function and the function structure are the most abstract possible TS representations because a function structure tells *what* the product must do, whereas its form or structure conveys *how* the product will do it [Ullman 1997, pp. 21, 122], [Pahl & Beitz 2007, p. 31]. The organ structure is the means of realising the function structure. The organs are in turn an abstraction of different types of components.

The process models have been developed with the intent to be either generic or ideal (or both). Generic would mean that the process *can* be adapted to each particular case; ideal means that the systematic engineering design process model could be followed in the ideal case where all the constraints are relaxed: available time, resources, costs, etc. Then the process model *has to* be altered to fit specific contexts. Thus, an ideal engineering design process model would be the following (see [Hubka & Eder 1996, p. 173]). An overall function is found, which is decomposed into subfunctions. The systematic variation of generally valid subfunctions (see [Pahl & Beitz 2007, pp. 174-175] and [Hirtz et al. 2002]) and task-specific subfunctions leads thus to the largest set of possible function structures. The most promising structures are then chosen, and for each subfunction a set of working principles are elaborated. These working principles are then combined into principle solutions (or concepts) from which the best one is chosen (or the best ones). The same tactic is then applied for the development of the constructional structure.

In [Ullman 1997] and [Ulrich & Eppinger 2007] the function decomposition serves mostly to simplify the task and is not used to systematically investigate variations of function structures. But their process model is still based on a continuous refinement (concretisation) of the TS [Ullman 1997, p. 33].

In [Pahl & Beitz 2007], the systematic variation of generic and task-specific subfunctions is emphasized, but with some restrictions (only simple decompositions, evaluation if possible, etc., see pp. 178-181).

The systematic engineering design process models are organised as problems to solve, following the archetype (here, Simon's model, [1961]): a) understanding the problem, b) generating solutions, c) evaluating and choosing solutions, d) implementing. These models also comport gates to take decisions (the so-called stage-gate model, for which the stages or phases in the systematic engineering design process models are often called conceptual design, embodiment design and detail design phases). The models allow for iterations between every step, and the problem solving process is also recursive: the first step can for example be considered as a problem in itself and be decomposed accordingly.

In summary, systematic engineering design process models are characterised by:

- an intent to be generic and/or ideal,
- the search for an optimal design,
- step-by-step concretisation (or refinement) procedure following a general TS model (see Figue 1),
- a problem solving process structure,

[3] In [Pahl & Beitz 2007] the term "concept" is reserved for the principle structures that have been sufficiently developed to be evaluated.

- a stage-gate form,
- decomposition-combination (discretisation) and systematic variations,

the specific benefits of the function decomposition are that it clarifies and structures the design problem [Ulrich & Eppinger 2007], it simplifies the research of solution and sub-solutions can be elaborated separately [Pahl & Beitz 2007, p. 37], it is central to creative design [Ullman 1997, p. 120-121] and it is solution-independent [Pahl & Beitz 2007, p. 31].

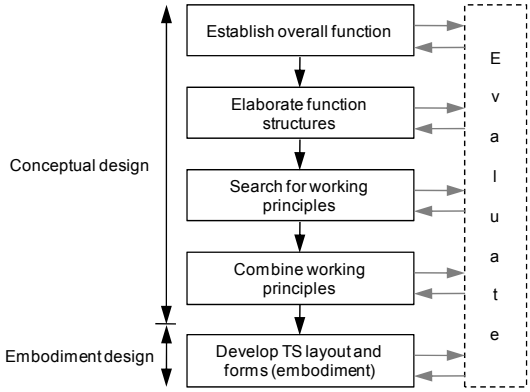

Figure 1. Part of the systematic engineering design process model following the TS model

3. Challenging the assumptions behind the systematic engineering design process models

From the beginning, discussions arose regarding the way these models were set up. They were presented in a very formal, near-algorithmic way (see e.g. Bender's historical review [Bender 2004, chap. 2.1]), suggesting that it would be possible to solve any kind of design problems with a general procedure. [Franke 1976] showed the theoretical impossibility of such a procedure. 10 years later, a collection of papers from the workshop "new impulses for systematic engineering design" at the ICED'85 conference exposed the limitations of such formal processes, triggered among other things by the fact that these procedures, although taught in nearly all technical universities, were still rarely used in industry or by practitioners. This really gave an "impulse" to more empirical studies of the engineering design process, placing the main user of the procedures, i.e. the engineering designer, in the centre of the investigations. These studies have been ranging from the observations about design thinking by a single engineering designer up to the area of social interactions and corporate culture. The early results of the studies of the engineering designer, mainly from the observations of the design activity modelled as a problem-solving activity, have been rapidly integrated into engineering design methodologies (already in the 3rd German edition of [Pahl & Beitz 1993], in [Ehrlenspiel 1995]).

The systematic engineering design process models are now presented as helpful to the engineering designer in case of complex engineering design tasks rather than as a procedure that needs to be followed in any case: "All the procedural plans proposed in this book have to be considered as *operational guidelines for action* […] In a practical application of these procedural plans, the operational guidelines for action blend with individual thinking processes. This results in a set of individual planning, acting and controlling activities based on general procedures, specific problem situations and individual experiences" [Pahl & Beitz 2007, p. 124, emphasis in original].

The core of the procedures present in the literature has not evolved, however; they present mainly the same characteristics as the first VDI 2221 in 1973 [VDI 1987]: a formalized process model that is assumed to theoretically, or based on best practices, lead to an optimal product. This is quite at odds with current research findings, but a first step has been taken towards trying to make a synthesis between the new findings and the engineering design process models (for example trying to harmonise

the models to the abilities and limitations of the engineering designers), without discarding them. As mentioned in the introduction, these models are still used as references in the engineering design research community.
Nevertheless, some criticisms concern the fundamentals of systematic engineering design process models *implying a renewal of these models*, especially when the very proponents of systematic engineering design are those who recommend the strongest deviations from the model they propose. The main arguments are presented below, and, when applicable, have been selected from the original works where the models are being presented. The next sections will focus on these areas: continuous concretisation following the TS model (sections 3.1, 3.2, 3.3), the TS model itself (section 3.4) and the decomposition/composition principle (section 3.5). A summary is given in section 3.6.

3.1 Continuous concretisation following the TS model

As already pointed out by Franke in 1985 [Franke 1985], this approach involves several problems, both theoretical and practical.
One difficulty is that there is nearly no one-to-one correspondence between functions, organs (function carriers) and components —apart from specific types of TS. That is, the choice of an organ often leads to the addition of a function, which changes the function structure. In the same way, the choice of the components will change the organ structure, which in turn will change the function structure. [Franke 1985] shows the necessary iterations with the development of a boiler feed pump (p. 920).

Main function: "increase the pressure"
Solution: multi-step centrifugal pump in a synchronised arrangement with radial separated housing and a common shaft
1. *Consequence:* subfunction "prevent axial shear"
   Solution: plunger discharge
   1.1. *Consequence:* subfunction "prevent remaining shear"
      Solution: deep groove ball bearing
      1.1.1. *Consequence:* subfunction "provide lubrication"
2. *Consequence:* subfunction "provide shaft sealing"
   Solution: gliding ring sealing
   2.1. *Consequence:* subfunction "Protect elastomer and gliding ring from too high temperature"
      Solution: cooling system
      2.1.1. *Consequence:* subfunction "control the closed cooling cycle"

Figure 2. Functional decomposition of a pump (from [Franke 1985, p. 920])

Moreover, [Malmqvist 1993]'s work shows in a more formal way the inevitable necessity of iterations. Malmqvist uses a set of generally valid functions and models the organs with the help of bond graphs, which implies that the organ structure is built solely with the use of physical effects, without considering the working principles material and geometric characteristics of the working principles. This follows literally the recommendations of the systematic engineering design procedures. It can be seen that the function structure almost unavoidably changes as the organs are added, because either no organs are known or because an organ needs auxiliary subfunctions to work properly (see e.g. pp. A.15ff and B.20ff).
Another problem is that it is impossible to select a solution based on *technical criteria* without at least some rough embodiment and dimensioning. On the functional level, all the function structures *that work* (which means that give the desired output given the specifications and constraints on the technical systems) have the same value. Similarly, a working principle structure needs to be refined before it is evaluated. As is mentioned by [Pahl & Beitz 2007, p. 131]: "Often, however, a working structure cannot be assessed until it is transformed into a more concrete representation. This concretisation involves selecting preliminary materials, producing a rough dimensional layout, and considering technological possibilities. Only then, in general, is it possible to assess the essential aspects of a solution principle and review the objectives and constraints." In [Pahl & Beitz 2007], this

concretisation is executed in the activity "firming up the principle solution variants", which incorporates engineering design tasks from the embodiment design phase although the activity is found in the conceptual design phase. In the fuel gauge example, one variant has to be developed at a detailed embodiment level before being rejected [Pahl & Beitz 2007, pp. 191-192]. Conversely, in the impulse-loading test rig for shaft-hub connections example presented in the embodiment design chapter, the calculations related to the main function carriers led to the development of new function structure variants [Pahl & Beitz 2007, pp. 424-425].
If the concept of a solution has to be selected upon *economic criteria* as well, the manufacturing system has to be taken into account, which means going deeper into the detailed embodiment of the product.
These arguments speak against the possibility in general to develop the function structures, working principles and product layouts separately, which is the way the process models are organized. Consequently, the possibility to find an optimal TS is hampered.

3.2 Concretisation as a necessary engineering design step order

The arguments above are against a linear process (development of the overall function excluded). Now, if iterations are added (Figure 3), should we still follow the same order, i.e. 1. elaborate function structures, 2. searching for working principles, 3. combining the working principles into working principle structures, 4. developing constructional structures?

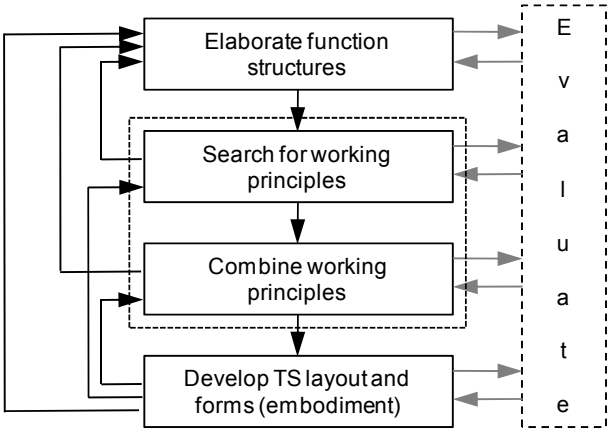

Figure 3. Iterative concretisation of the TS

Let us consider the way a function structure is developed, once the overall function has been determined, utilising an example. Be the overall function: "remove dirt from the window", represented in Figure 4. How do we go from there? A top-down solution-independent approach would require using generally valid functions (e.g. [Hirtz et al. 2002]). However, as stated in [Pahl & Beitz 2007, p. 175]: "In many cases in industry it may not be expedient to build up a function structure from generally valid sub-functions because they are, in fact, too general and thus do not provide a sufficiently concrete picture of the relationships for the subsequent search for solutions." Moreover, the use of generally valid functions results in an infinity of possible structures. Indeed there is no limitation in the variation of the use of functions; see examples of function structures in Figure 5 decomposing the overall function in Figure 4 (solution principles will be evoked in section 3.4.2).

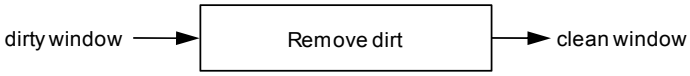

Figure 4. Overall function "remove dirt from the window"

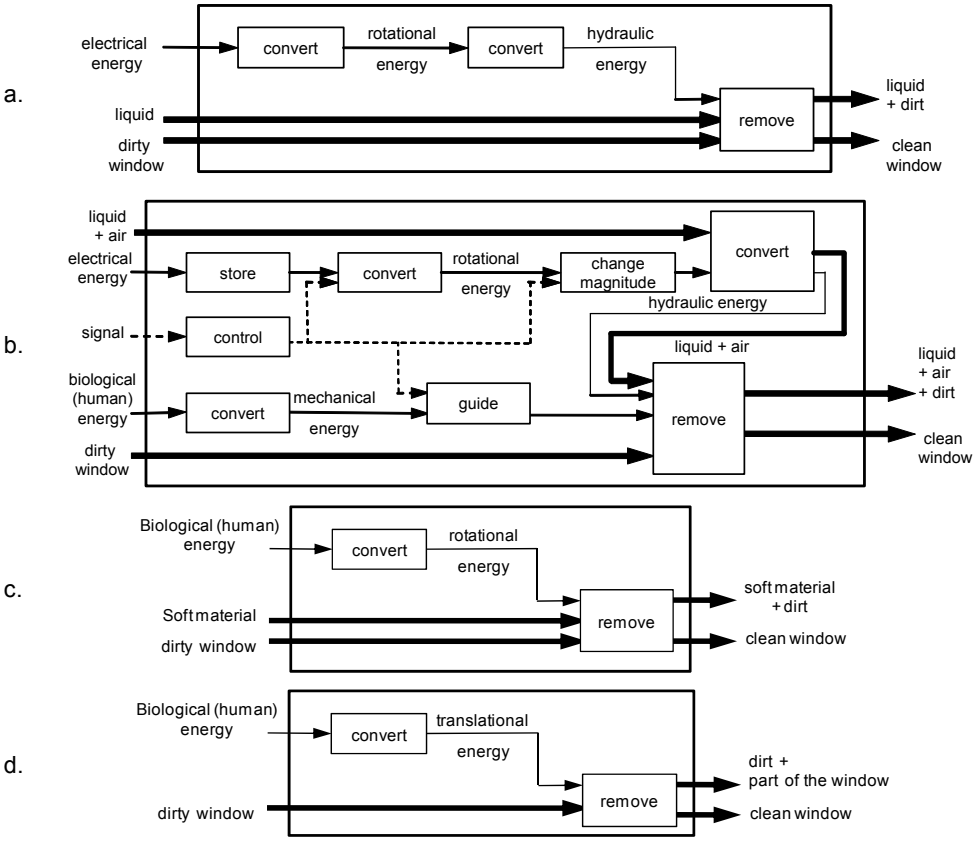

Figure 5. Examples of function structures based on the overall function "remove dirt from the window"

What remains is the use of more task-specific functions and/or the analysis of existing systems [Pahl & Beitz 2007, p. 178]. But in the first case it seems difficult to imagine task-specific functions without implicitly thinking of some tasks-specific organs or knowing some components that fulfil that function (step 2, 3 or 4 then 1). The analysis of existing systems is by nature a bottom-up approach. Is it also possible to always have step 2 4? As [Pahl & Beitz 2007] put it: "the *stepwise* generation of working principles, through the search for physical effects and the subsequent form design features, is often mentally integrated by producing *sketches of solutions*. This is because design engineers think more in configurations and representation of principles than in physical equations." (p. 189, emphases in original.) Finally, in [Pahl & Beitz 2007] itself it is admitted that in some cases "it is perfectly legitimate to conceive a preliminary solution and then abstract this by developing and completing the function structure by a process of integration" (p. 179), although the latest edition of *Engineering Design* still presents the process as perfectly linear [2007, p. 160].

Finally, if the engineering design process cannot follow the order 1. elaborate function structures, 2. searching for working principles, 3. combining the working principles into working principle structures, 4. developing constructional structures, is it necessary that at least the steps be *completed* in that order? Again the answer is no. The impulse-loading test rig for shaft-hub connections example from [Pahl & Beitz 2007] evoked above is a good illustration that the function structure is not completed before the rough layout is. The three steps are highly interrelated and cannot be apprehended separately (Figure 6).

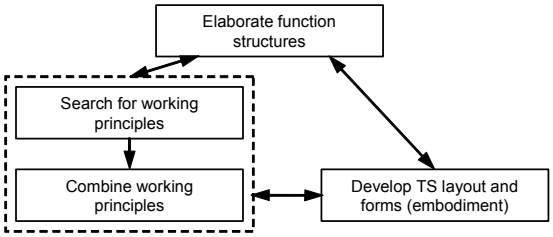

Figure 6. The TS-related design activities are not ordered

3.3 Concretisation steps as necessary for new product design

It is clear that the interplay between function structure, working principle structure and embodiment helps in finding new solutions. However, other creative methods also lead to novel technical systems. Franke wrote in 1985: "Let us mention for example, brainstorming and synectics. Beside the often mentioned inconvenience that they are never used at the right time, or are forcefully used, we need to acknowledge that these methods in a very short time can lead to very concrete alternatives, which are comparatively of good quality." (p. 915) This point, and the others in the preceding two sections, speak against concretisation as a necessary feature of the engineering design process.

3.4 The TS model

3.4.1 Briefly on the concept of working principle

The current models of technical systems include the working principles as an abstract junction between the concrete component and the abstract function. This system works fine for the classification of technical systems, and for the documentation of developed products, or the creation of databases (see e.g. the design repository, [Hirtz et al. 2002]). The concept of working principle or organ, however, leads to some difficulties when used during the creative steps of engineering design. As described in [Pahl & Beitz 2007]: "Only the combination of the physical effect with the geometric and material characteristics (working surfaces, working motions and materials) allows the principle of the solution to emerge. This interrelationship is called the *working principle*." (p. 40, emphasis in original.) [Hubka & Eder 1996, p. 106] go even further by focusing on the geometrical aspect only: "main features for the organs are the action spaces, surfaces, lines, etc." Moreover, it is mentioned that the main problem, during the combination of working principles into a solution principle, "is ensuring the physical and geometrical compatibility of the working principles". This makes it difficult to separate the activities of working principle from the output of the embodiment design activity: "*Embodiment design* incorporates both *layout design* (the arrangement of components and their relative motions) and *form design* (the shapes and materials of individual components)." ([Pahl & Beitz 1996, p. xx, emphases in original.)

One consequence of that is that it makes the engineering design activities, previously separated, even more entangled (Figure 7).

3.4.2 Briefly on the function structure as an abstract model

The saying that the function structure represents *what* the product must do, whereas its form or architecture conveys *how* the product will do it, leads to some subsequent problems. We come back to the example "remove dirt from the window". If each single function tells what it does more than how it does it, the function structure certainly is not solution-free, but merely an abstraction of how the product would perform. First of all, if the function structure were truly solution-independent, there would be only one core function structure for each overall function. However, the function structures presented in Figure 5 are quite different. A solution principle for the 2 first ones would correspond to pressure washers, the second one a simple wiping system controlled by a human, the third one a thin plastic layer removed from the window (with the presupposition that a set of thin plastic layers were added to the window during manufacturing). It can also be noticed that it would be very challenging to

find a solution given the function structure presented in Figure 5.d. It would be even more challenging to generate such a function structure (part of the window removed with dirt) without having a solution in mind.

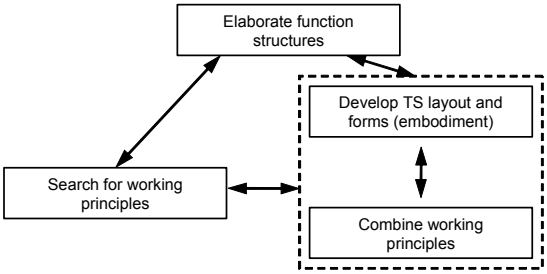

Figure 7. The interconnections between the TS-related engineering design activities

This conception of the function is not shared by all the stakeholders; [Hubka & Eder 1996] and [Pahl & Beitz 2007] allow the generation of several function structures. The function structure is here only a more abstract way to describe a TS. But even this milder way of considering functions does not guarantee the function structure as central to an engineering design process model. The function structure is well adapted to TS that imply a continuous flow of energy or matter. However, there are cases where the use of functions is difficult to adapt. Here are some examples:

- Mechanisms: they can be perfectly modelled with kinematic diagrams.
- Hydraulic systems: there exist dedicated symbolic representations for developing them.
- Describing "Passive" technical systems by functions is of little help for finding the working principles. A footbridge, for example, has for unique function "channel" (matter, i.e. pedestrians); a rail buffer is described by the function "stop" (also matter). At a conceptual level, the symbolic language used in material sciences would be of greater help.
- Many products perform multiple tasks. [Ullman 1997] presents a function diagram for a TS with 2 tasks (p. 131), but these tasks were independent (installation and use). It is not rare though that a system may perform different tasks depending on different contexts. A symbolic extension towards handling conditional or temporal actions is needed, as is done with Petri nets or Grafcet.

3.5 Decomposition/Combination

The possibility to decompose, and then combine, is also at the heart of the systematic engineering design. It allows a systematic search for all the subfunctions, increasing the chances for finding the optimal combinations. The awkward side-effect is the explosion of combinations of solutions as well as their evaluations. Help from the proponents of such methods is rather poor and contradicts the very motivation behind the systematic engineering design methodology: to rely on the engineering designer's intuition and experience for these complex tasks. [Ulrich & Eppinger 2007] proposes for example multivote to exclude the least wanted, [Pahl & Beitz 2007] recommends to "concentrate on promising combinations and establish why these should be preferred above the rest" (p. 186). The authors also recommend using the morphological matrix for only a subset of all the subfunctions in order to limit the combinatorial explosion. Selection methods in the literature are only applicable in particular cases: when all the subsystems are independent, the best solution of each subsystem gives the final solution (given that they are compatible with each other); conjoint analysis can be used if the hypothesis that a maximum of 2 subsystems are dependent holds. Otherwise, the general case is that the sum of the evaluations of each working principle does not equal the evaluation of the solution principle.

Moreover, decomposition hinders the finding of integrated (sub)solutions; it makes systematically difficult the investigation of the compatibility between subsystems; and some requirements that

concern the whole product (aesthetics, vibrations) must be overlooked until late in the combination (see [Franke 1985]).
Finally, it hinders the use of a more holistic approach, where the whole solution is found and then adapted in its details to the requirements or to some subfunctions.

3.6 Summary

The arguments presented above make it impossible for process models to be taken as ideal, or even as generic. It is not possible to independently handle function, working principle and component structures; it is not even possible to claim that "form follows function". The function structure is not the only possible abstract representation of a TS. The concept of working principles as a physical effect together with geometric and material characteristics is difficult to work with because they are strongly interrelated with form and layout (embodiment). Adaptations of the systematic engineering design process models to some of the 28 types of engineering design tasks (cf. section 2) are usually presented together with the systematic engineering design process models, but some other types that are very remote from the proposed procedure, e.g. the development of a production system-dependent TS (common in industry) are left up to the engineering designer. Finally, even in the case of totally new products, where the models are supposed to fit best, there is the incorrect assumption that the best product is sought. This is rarely the case, for the complexity and insecurity that go together with a novel design result in the discovery of *one* functioning TS being largely satisfactory. Therefore, a step-by-step concretisation of the product with a systematic variation of subsolutions and recombination shall not be taken as a central and necessary feature of an engineering design process model.

4. (New) bases for engineering design process models

Taking away those central features of the traditional engineering design process models seems to leave an empty space: by what should they be replaced? We will see below that this question may be misleading: it is not about finding some new *generic* process model. However, it is legitimate to require descriptions of the basic elements that would constitute *other* possible engineering design process models. In the following, only some elements are presented; further research is needed to complete them.
To begin with, that some elements constituting the systematic engineering design process models are no longer central to the engineering design process does not mean that they have to be completely rejected or abandoned. They are on the contrary very useful. Secondly, the arguments presented in section 3 concern only some of the fundamentals. The problem-solving process, even though its use can also (and has been) criticised, can still be part of any process. In a way, it represents the basis of nearly all possible tasks – understanding of the task, generation of one or more possible strategies to solve the task, choice of one strategy, implementation. Recall, too, that only the process model was discussed, not the methods.
Thirdly, the stage-gate model is not invalidated. However, the stages should not be dependent on the continuous concretisation of the product. What decision makers need, at least for the first gate (i.e. concept design evaluation; product planning will be dealt with later on), is enough information on the *technical and economical feasibility* of the product-to-be. This implies knowing with a certain confidence that the product will "work", that it can be manufactured at the planned costs, and that its market still exists. Thus, during the "conceptual design" phase (*feasibility study* would be more appropriate), it is important to focus on the critical or weak spots of the TS, and solve the related issues. In most cases this will involve work at the detailed level, and possibly with advanced analysis tools, even for novel design. Less critical spots can be left untouched, either because they have already known solutions or because the engineering designer *knows* (within a certain confidence interval) that these spots can be solved.
As the weak spots differ from product to product *and* from engineering designer to engineering designer, there is a need for a *very flexible model*, with different methods available for different engineering design problem configurations. It is also necessary to acknowledge, not the impossibility of a generic and/or ideal process model, but the possibility that there are many cases where specific

models should be applied. Ultimately, what should be proposed is a *model for modelling* specific engineering design procedures.

4.1 Flexible engineering design process models

The idea of a flexible engineering design process is not new. [Bjärnemo 1983, p. 390] already presented the need for flexibility for engineering design procedures. Even from the first edition of [Pahl & Beitz 1977] it is argued that their process needed to be adapted to different situations (what has been shown above is that there is always a need for so many adaptations that it undermines the very procedure).

The word "flexible" has to be understood along several dimensions. A first one is that the concept of a flexible model is about choosing the accurate methods and tools for the given situation. It presupposes some criteria for choosing them. Some classifications already exist, e.g. [Olsson 1976, pp. 147-148]. A literal application of this view has some downsides, as it also presupposes that the engineering designer knows the methods and tools, or can learn and master them fairly quickly in order to apply them. [Bender 2004] proposes that a "method pool" be assembled, learned and trained. He also extends the concept of flexibility in engineering design. He simply recommends *trusting* the engineering designer to use his/her own experience for solving an engineering design problem, and *being confident* that he/she will use explicit structured process models when necessary, (re-)placing the engineering designer in the centre of the engineering design process. The motivation for that stems from findings about learning in cognitive psychology.

The learning process incorporates three phases: a cognitive phase where the explicit knowledge of facts and rules is built; an associative phase where explicit knowledge is internalised by training and becomes implicit; and an autonomous phase where this knowledge is mastered and used in an almost automated way (see pp. 57-61). In short, [Bender 2004]'s experiments with engineering designers also showed that an experienced engineering designer presented better design results by using implicit knowledge than by using explicit knowledge. The fact is that the *knowledge* of past solutions, diverse technologies, and various scientific disciplines, possession of problem-solving *skills*, and the designing *experience* are already fairly good tools for dealing with a complex activity like that of designing a new product. *A prerequisite for any engineering design process model is to involve the engineering designer's experience.*

For a novice, [Bender 2004] showed that following a structured process model (like the ones referred to above, although others are possible) reduced the risk of failure (but also reduced the quality of the design in case of success). However, he also showed that starting with a concrete level of representation of a TS is more appropriate that starting on a more abstract level. The novice needs to acquire and internalise the necessary knowledge, which means shifting between the two dimensions of active/passive and abstract/concrete learning. Learning engineering design process models and using them in industrial projects covers the passive/abstract, active/abstract, and active/concrete pairs. What seems to be missing is the passive/concrete pair, the concrete approach to TS, necessary for an engineering design process model based on experience. Experience of TS could be enhanced by re-introducing "learning by example", i.e. learning by studying and re-creating existing designs. Many good engineering designers have in their past acquired a feeling for mechanical products by disassembling and examining them, but the systematic engineering design process models, by their aspiration to be generic and their claim that engineering design needs to be top-down to be successful, implicitly ruled out such learning activities.

4.2 The place of systematic engineering design methods

As observed above, the principle of systematic engineering design need not be abandoned, but its elements need to be re-positioned in the engineering design process according to what it really contributes. The overall function helps to clarify the task. The systematic variation is a heuristic for solution generation that works for each level of representation of the TS, especially for form design. The iteration between the different levels of abstraction of the TS is also a creative method: "The exploration step of concept generation usually acts more as a guide for further creative thinking than as the final step in the process" [Ulrich & Eppinger 2007, p. 118]. It is to be placed on the same level

as other creative methods like brainstorming and synectics. The TS model is also a helpful formalism to document the finalised product. The function structure can also be used for analysis of existing products, and to search for new solutions with the help of bond graphs or for design catalogues. As seen earlier, it would be a mistake to deduce from this multitude of uses of the functions at different stages and levels that it is central to engineering design. In conclusion, these diverse elements can be used for understanding the design problem and generating solutions, but need to be placed on the same level of "importance" as other methods.

4.3 Integration into the product development process

As an ongoing basis for engineering design process models, the stage-gate process model implies that the engineering design process must be much more integrated in the product development process. The systematic engineering design process models were mainly built around the TS; the engineering design process model must be set towards the goals behind product development: developing a product-to-be in accordance with production, marketing and corporate strategy. This is the domain of integrated product development, or IPD, where the current models still use the systematic engineering design process models as a basis for the engineering design activity, with activities decomposed by the concretisation level of the product.

There are other ways of conceiving the development of products. If one agrees that technical and economic feasibility is the key to successful further development (and many do, as the emphasis of most papers on the "conceptual" phase is based on the hypothesis that this phase is mainly responsible for the success of the rest of the project), then one can consider it as a step of its own. If the product is considered feasible, then a (routine) product development project can begin. Several variations on this proposal can be advanced. One of those is the R-I-D framework by Hatchuel et al. [2001]. It is proposed to organise the development of technologies and products within three clearly differentiated functions: Research, Innovation and Development (R-I-D): The Innovation function defines development directions, new product values, new competencies and knowledge. It also encompasses product planning as well as strategic decisions traditionally devoted to the Research departments. The Development function is then concerned with routine product development and the Research function would assume the controlled process of knowledge production (Figure 8)

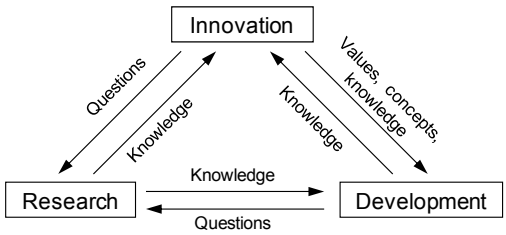

Figure 8. The R-I-D framework [Hatchuel et al. 2001]

5. Conclusion

This paper has demonstrated that the central features of systematic engineering design process models — a step-by-step concretisation of the product with a systematic variation of subsolutions and recombination — are not absolutely necessary features of an engineering design process model. The current models are neither generic nor ideal. Instead of using the product characteristics as a basis for an engineering design process model, these models should focus on the concrete goals of product development in industry. As these goals are manifold, the engineering design process needs to be flexible, allotting more freedom to the engineering designer. Eventually, it will be necessary to develop a model for building specific engineering design procedures. The stage-gate model is important because it is necessary to assess whether a project should continue, be changed or abandoned. One important gate is product feasibility assessment (preferable to the term conceptual

design) and for that, the usual scenario of a product development project from product idea to finalised documentation is challenged. It is conceivable to separate the feasibility study from a more routine product development.
The aim of this paper is not to replace the existing working models with a "better" one. Indeed the (new) bases are not new: It is about acknowledging the necessity of a flexible approach to the engineering design process, which among other things gives greater prominence to the engineering designer.
One concern is the learning and teaching of this different conception of the design process. The systematic engineering design process models has the advantage of being relatively sequential and structured, which eases the apprenticeship (even if iterations were included). However, would it really be more difficult to learn several specific processes, or to learn how to model one's own process? Flexibility does not contradict structure. Efforts must be invested in acquiring a complete engineering design experience [Bender 2004], which among other things implies re-introducing "learning by example" or developing other, similar, methods for "passive/concrete" learning.

References

Bender, B., Successful Individual Design Process Strategies in the Earlier Phases of Product Development (In German), Fortschritt-Berichte VDI (Reihe 1, Nr. 377), Faculty for Vehicle and Machine Systems, TU Berlin, Berlin, 2004.

Bjärnemo, R., Formalized Approaches to Engineering Design (In Swedish), Machine Design, Lund University, Lund, 1983.

Ehrlenspiel, K., Integrated Product Development (In German), Carl Hanser Verlag, Munich, 1995.

Franke, H.-J., Investigations of the algorithmisability of the design process (In German), TU Braunschweig, Braunschweig, 1976.

Franke, H.-J., "Engineering design methodology and practice - A critical consideration" (In German), 3rd International Conference on Engineering Design - ICED'85, WDK 12, Vol. 2, pp. 910-924, 1985.

Hatchuel, A., Le Masson, P., Weil, B., "From R&D to R-I-D: Design strategies and the management of 'innovation fields'", 8th International Product Development Management Conference, pp. 415-430, 2001.

Hirtz, J., Stone, R. B., McAdams, D. A., Szykman, S., Wood, K. L., "A functional basis for engineering design: Reconciling and evolving previous efforts", Research in Engineering Design, Vol. 13, No. 2, 2002, pp. 65-82.

Hubka & Eder, V., Eder, W.E., Design science, Springer, Berlin, 1996.

Malmqvist, J., Towards Computational Design Methods for Conceptual and Parametric Design, Dept of Machine and Vehicle Design, Chalmers University of Technology, Gothenburg, 1993.

Olsson, K. G. F., Systematic Engineering Design (In Swedish). Machine Design, Lund University, Lund, 1976.

Pahl, G., Beitz, W., Engineering Design – A Systematic Approach, (3rd Edition), Springer, London, 2007.

Simon, H.A., The New Science of Management. New York, NY, Harper, 1960.

Ullman, D.G., The Mechanical Design Process, (2nd Edition), McGraw-Hill, New York, NY, 1997.

VDI, VDI Guideline 2221: Systematic Approach to the Design of Technical Systems and Products (Translation of the German edition 11/1986), VDI-Verlag, Düsseldorf, 1987.

Damien Motte
Lund University, Dept of Design Sciences LTH, Div. of Machine Design
P.O. Box 118, 221 00 LUND, Sweden
Tel.: 00 46 (0)46 222 85 13
Fax.: 00 46 (0)46 222 80 60
Email: damien.motte@mkon.lth.se
URL: http://www.mkon.lth.se