INTERNATIONAL DESIGN CONFERENCE - DESIGN 2012
Dubrovnik - Croatia, May 21 - 24, 2012.

FUNCTIONAL MODELLING PERSPECTIVES ACROSS DISCIPLINES: A LITERATURE REVIEW

B. Eisenbart, L. Blessing and K. Gericke

Keywords: functional modelling, function formulation, cross-disciplinary

1. Introduction

Product development in industry increasingly requires the integration of different technologies, necessitating a closer collaboration of experts from different (engineering) disciplines. The inclusion of services in relation to the product, e.g. as integrated Product/Service Systems (hereafter PSS), which has become increasingly important in the recent past, extends the need for cross-disciplinary collaboration even further. The term "technical system" used in this paper encompasses both technical products and PSS.

The integration of expertise from different disciplines into a technical system needs to be based on a shared understanding of the constraints and design objectives in order to support sound decision making on alternative discipline-specific concepts or integrative concepts. Hence, designers need to elaborate, visualise, and clarify to each other requirements and functions of the system under development. Communication between different designers is typically facilitated through design models, such as textual descriptions, function models, as well as sketches and drawings [Buur and Andreasen 1989].

Supporting integrative functional modelling seems particularly beneficial to support the establishment of a shared understanding across disciplines, as it addresses solution finding early in the process and on an abstract level. Functional modelling is proposed within design methodologies to facilitate the formulation and linking of required functions. Function formulation is regarded as an essential step in the transition from the requirements specification to the potential concepts, which is eventually transferred into a comprehensive representation of the final technical system [Blessing 1994].

Despite their importance, the exchange of models between disciplines is often hindered because of a lack of knowledge with respect to the other disciplines, differences in terminology, and different modelling approaches. Even within the disciplines a shared understanding of function seems widely missing [Vermaas 2011] and seems to differ among researchers [Ullman et al. 1992]. Potential consequences could be design flaws leading to additional development time and – if undetected – to problems with the implementation and use of the technical system.

The research presented in this paper discusses the different understandings of function which hamper the establishment of a shared understanding and shared functional modelling. Furthermore, function models proposed in different disciplines are analysed, in order to identify the different inherent functional modelling perspectives. The analysis focuses on mechatronic product development – covering mechanical engineering design, electrical engineering design and software development – as well as PSS development.

2. Diverse understanding of function

The functions of a technical system allow users to draw value from the system and fulfil a certain purpose. Despite the centrality of function formulation to technical system development – as highlighted in Section 1 – "[…] function lacks a single precise meaning. It is a term that has a number of co-existing meanings, which are used side-by-side in engineering" [Vermaas 2011] and various definitions of function exist [Crilly 2010]. Based on a comparison of twelve different definitions of function from engineering design Crilly argues that, depending on the emphasis, essentially two ways of understanding function exist: "Some definitions emphasise that a transformation must take place for a function to be fulfilled […], whereas others require that some purpose, goal or requirement must be satisfied […]" [Crilly 2010, p. 313].

The "transformation" understanding of function has been introduced in the 70s to engineering design mostly by German authors and has since been widely taken up in design research [Stone and Wood 2000]. Function is referred to as transformation between states of some basic operands, which are typically specifications of material, energy, and information (also called signals), between the input and output of a technical system [Pahl et al. 2007]. An example of a model representing this type of function is given in Figure 1. In the upper model, the state of the main operand (potatoes) is transformed from "in the ground" to accepted and rejected potatoes, as well as some left over materials. The lower model represents the succession of individual transformations of distinct operands that can be used to achieve this overall state change.

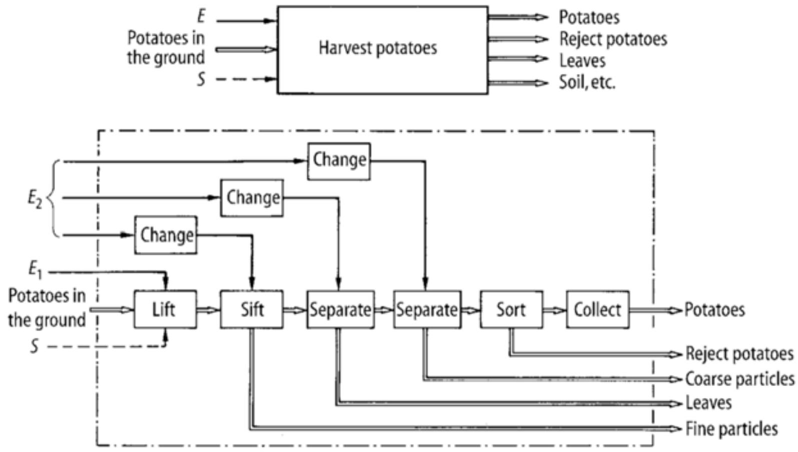

Figure 1. Example of representing function related to transformation of operands: function structures after [Pahl et al. 2007]

Various authors argue that numerous functions simply cannot be represented by transformations, and that satisfying a purpose through a technical system is by far not limited to input/output relations (see also [Carrara et al. 2010], [Vermaas 2011]). For example, providing status, is a function, which can be relevant to a user, but which does not (directly) refer to a transformation in the system.

To account for the particular purpose a function aims to satisfy (rather than which effect it causes – i.e. transformation of an operand) Crilly [2010], seeks to derive a limited set of function types (referred to as "classes") which cover most of the types discussed in the literature. He differentiates three main types: *technical* functions (related to the actual physical properties and behaviour of a technical system), *social* functions (with respect to e.g. a user's social context), and *aesthetic* functions (e.g. "convey beauty"). A similar differentiation is proposed by [Aurisicchio et al. 2011], who, in addition, distinguishes emergent functions (unwanted behaviour or properties of the system) and economic functions (a rather process management oriented perspective of function).

3. Approaches to shared functional modelling

To support the communication of the functions of a technical system various design researchers aim at supporting the process of function formulation. Despite the existing diverse understandings of function (see Section 2) they aim to build functional modelling on one common understanding of function. Some aim at deriving this common understanding by comparing different representations of function within function models. Others seek – respectively impose – one distinct understanding, which is *bridging* the existing diversity [Vermaas 2011].

Based on a recent and comprehensive review, Erden et al. [2010] argue that a common understanding of function cannot be derived from the function models they analysed. They emphasise that this is due to the large diversity in the way function is represented by the different models particularly across disciplines.
Bridging diversity by imposing one understanding of function is typically facilitated by proposing formalised functional taxonomies (e.g. [Stone and Wood 2000]), involving distinct classes of verb-noun representations of function. Therein, the verb depicts the particular effect caused to an object, which is depicted by the noun. Figure 2 shows examples of such distinct classes and indicates the links between the classes proposed by the different authors.

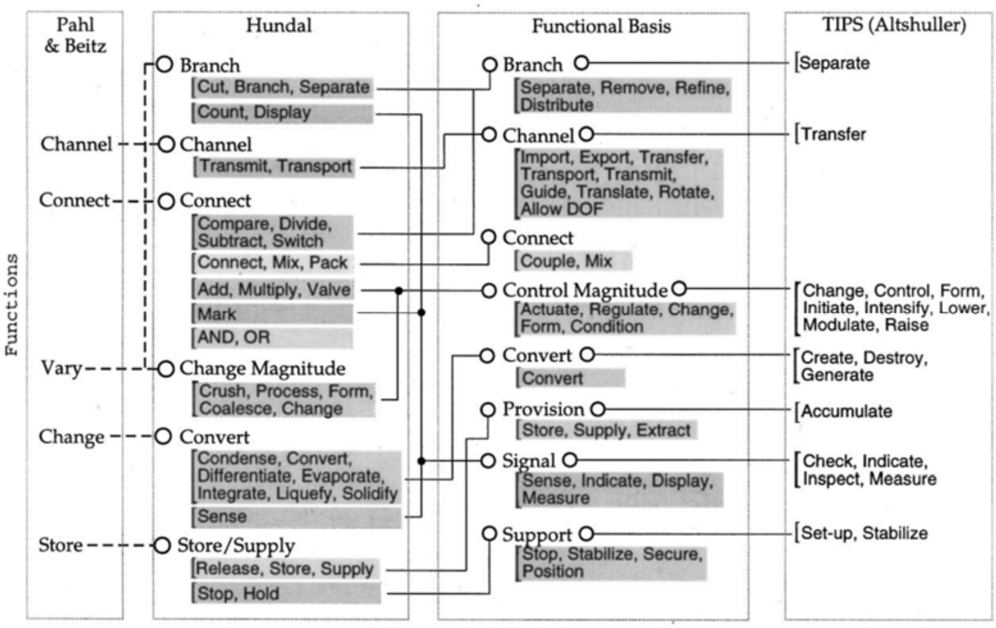

Figure 2. Comparison of different classes of function after [Stone and Wood 2000]

[Ahmed and Wallace 2003] analysed designers in industry trying to translate functions from existing textual representations in natural language into two different taxonomies. The translation frequently led to loss of (mostly context-related) information and, in some cases, could not even be performed. Other researchers, such as [van Eck 2010] found that the conversion of functions between different taxonomies could lead to loss or change of information. Vermaas, in a recent review [Vermaas 2011], argues that not all of the taxonomies are indeed proposing or respectively using a truly unambiguous understanding of function, but – at least implicitly – accept alternative understandings side-by-side. This (implicit) ambiguity is also argued by Vermaas to subsist in approaches aiming at finding a bridging understanding (e.g. by [Chandrasekaran and Josephson 2000]).

4. Towards integrated functional modelling

Considering the short-comings of the approaches to shared functional modelling and the diverse understandings of function discussed in literature, various authors e.g. [Vermaas 2011], [Carrara et al.. 2010] argue that this diversity is in line with common design practice in technical system development and should therefore be accepted.
From a modelling perspective, the development process of a technical system moves through a succession of distinct abstract design states (see e.g. [Blessing 1994]). A design state denotes all the available information about a technical system at a particular point during its development – stored within successively created design models [Buur and Andreasen 1989]. Focusing on the design state representing functions of the technical system, we identified several distinct (sets of) models, proposed in the different development methodologies of the disciplines studied. The models are proposed to support designers with the formulation of functions.

Erden et al. analysed *how* functions are represented in the different models, classifying models with regard to used ontology, representation formalism, used semantic definition, etc. The research presented in this paper aims at deriving *what* is represented within functional models. From this, we aim at deriving the particular functional modelling perspectives, which need to be integrated to support the establishment of a shared understanding across disciplines.

4.1 Approach

The review focuses on mechanical engineering design, electrical engineering design, software development, and interdisciplinary system development approaches, including PSS design. In addition, function models from building design have been included for comparison reasons.

4.1.1 Categorisation

To derive the different modelling perspectives addressed in the analysed function models each model was categorised using on Buur and Andreasen's modelling morphology [Buur and Andreasen 1989]. Table 1 shows the application of the modelling morphology for the function models in Figure 3. Particularly the differentiation between *object* (e.g. function structure, feature list, etc.), *property* (e.g. specification of overall and sub-functions, transformation flow of operands, etc.), and *purpose* (e.g. structure hierarchically, indicate sequence of transformations in time, etc.) of individual function models has been beneficial for deriving the functional modelling perspectives used in each model.

Table 1. Example of the application of the modelling morphology

design model	object	property	purpose	user	code	medium
Use case schematic	Actors and use cases	Actors, user interaction processes, processes within the system	Visualise	Project team, project manager, the designer himself	Symbols for actors, arrows indicating flow of request, etc.	Various options (e.g. paper, computer screen, etc.)
Function tree	Hierarchy of system functions	Overall and auxiliary functions	Structure hierarchically		Tree hierarchy, verb-noun representation	

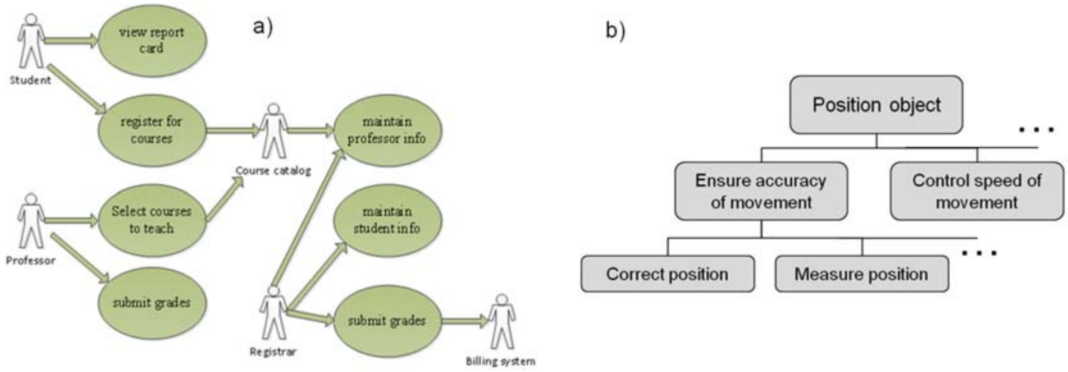

Figure 3. a) example of a use case schematic b) example of a function tree

4.1.2 Analysis

The analysed function models have been summarised in Table 2[1], showing which of the identified modelling perspective(s) each of them addresses. For several function models variants are proposed, even within one discipline. In these cases, each variant has been analysed and categorised separately. When multiple perspectives are represented within a function model, one particular perspective typically drives the generation of this particular model; this driving perspective is highlighted (for details see Section 4.1.3).

[1] The references of the publications in Table 1 have not been included in the article because of the required limit of 10 references. They can be obtained from the authors.

Table 2. Overview of functional modelling perspectives proposed across disciplines

	Model	Author(s)*	Model used/ built upon	Functional modelling perspectives: States	Effects	Processes within the system	Interaction with the system	Use cases	(Potential) operating system allocation	Stakeholder allocation	Structuring of representation: Function hierarchy	Flow related to time	Flow of operands
Mechanical engineering design	Function structure	Pahl & Beitz	Black box model	x	**x**	(x)					x	(x)	Energy, material, signals
		Ulrich & Eppinger		x	**x**	**(x)**			**x**		(x)	o	
		Stone & Wood		x	**x**	(x)	(x)					x	
		Hundal				(x)						o	
		Altschuller				(x)						o	
		Rodenacker		(x)	**x**	**(x)**						o	
		Hubka	Technical process structure		**x**	(x)	(x)		x	x	x		
		Ulman	Function tree, as partial models		**x**	(x)							
		Ehrenspiel	Functions list	x (after each operation)		**x**						x	
	Technical process structure	Hubka	Black box model	x	(x)	**x**	(x)		(x)	x		x	E, M, S
	Transformation process model	Hubka	TP structure, function structure	x	**x**	**x**	(x)		x	x	x	x	E, M, S
	Process structure	Blessing & Upton		**x** (after each operation)								x	
	Process flow model	Tjalve	Black box model	**x**		x	o					x	Material
	Process/function chart	Tjalve	Man/machine separation list	x		**x**	x		x	x		x	Material
	Functions list	Ehrenspiel		(x)		**x**					x		
	Functions tree	Pahl & Beitz	Black box model	x	x	(x)					**x**		E, M, S
		Pugh		o	o	x	o	o			**x**		
		Ulman	Black box model		x	(x)					**x**	(x)	
	Function means tree	Tjalve	Process/function chart		(x)	**x**			**x**		x		
		Hansen & Andreasen											
		Hubka											
	User action sequence	Ulrich & Eppinger	Black box model	x			**x**		x			x	
	Man/machine separation list	Tjalve					(x)		**(x)**	**(x)**			
	Use concepts	Roozenburg & Eekels					**x**	x					
Electrical engineering	Function tree	Dewey			x			(x)	x		**x**		Signals
	State diagram	Beck et al; Scheffer et al.		**x**				(x)				o	
	Function table	Beck et al; Scheffer et al.		x	**x**		o				(x)		
	(V)HDL-specifiaction	Dewey				**x**		o			x	x	
	Sequence diagram	Beck et al; Scheffer et al.				x	o			x		**x**	
	Function table	Beck et al; Scheffer et al.			x		o						
	Petri net	Beck et al; Scheffer et al.		**x**								**x**	
	CFRL	Fikes et al.		x		**x**		o				x	

- *empty* does not apply; not included *
originates from software development but is also used in PSS design
- *bold* driving aspect ** discrete,
continuous, and signal event flows
- *brackets* functional modelling perspective may be included into function model *** includes
technology allocation (mechanical, electrical, etc.)
- "o" implicitly included in function model (as discussed in Section 4.1.2)

Table 2. (continued)

Software development	Feature list	Kruchten				x	x	o			(x)		
	Release backlog	Schwaber (SCRUM)				x	x	o			(x)		
	Storyboards	Cooper		(x)		x	x	(x)	(x)	x		x	
	Use case schematics	Kruchten	Feature list			x	x	x	x	x		x	
		IAGB (V-Model XT)											
	Use case description	Kruchten	Use case schematics			x	x	x		x	x	x	
		IAGB (V-Model XT)											
	Sequence diagram	Kruchten	Use case description/schematics			x	x			x		x	
	Activity/event diagram	Kruchten	Use case schematics			x	x			x		x	
	Activity diagram	IAGB (V-Model XT)				x	x			x		x	
	Functional flow diagram	Bosman				x						x	
PSS design	SADT (IDEF0)*	Ross		x (after each operation)		x	x		x	(x)		x	
	Service process model	Watanabe et al.		x (after each operation)		x	x		x	x		x	
	Service blueprinting	Shostack				x	o		x	(x)		x	
	Customer chain (CC) model	Donaldson et al				x	x		x	x			
	Customer activity cycle (CAC)	Tan				x			x	x		x	
	Block diagram	Fisher & Schutta				x			(x)	(x)		x	
	Interactor network	Maussang							x	x			
	Function Analysis System Technique (FAST)	Bytheway		x		x			x		x	x	
		Maussang				x			x		x		
	Functional bloc diagram	Maussang				x	x		x	x	x		
	Scenario model (transition graph)	Sakao& Shimomura		x						x			
	Flow model	Sakao& Shimomura								x	x		
	Scope model	Sakao& Shimomura	Flow model	x							x		
	Chain of actions	Sakao& Shimomura	Scenario model			x	x		o	o		x	
	View model	Sakao& Shimomura	Flow, Scenario and scope model	x		x					x		
	View model and realisation structure	Sakao& Shimomura		x		x			x		x		
Mechatronic system development	Function structure	Kajitani			x								E, M, S
	State transition diagram	Salminen & Verho	Context and flow diagram	x		x		(x)					
		Buur		x				x					
	Function tree	Salminen &Verho				x		x			x		
	Events list	Salminen & Verho				x	x		x	x		o	
	Context and flow diagram	Salminen & Verho	Event list	x		x	x		x	x			**
	(Active) purpose functions	Buur	State diagram			x		x				x	
	Transformation functions	Buur			x			x				o	
	Expanded function/means tree***	Buur	Purpose and transformation function models		x	x			x		x		
	Function + principle sol.	Möhringer	System structure		x						x		
BD	"Grössengliederung"	Engel; (similarly: Jones)			x		(x)		x		x		

The columns on the right hand side of the table indicate the way the different function models structure the represented information and link the individual represented functions. Three main ways have been identified: *hierarchy*, *flow of operands*, and *flow related to time*.
Functions lists or function trees (see Figure 3), for example, structure the different functions hierarchically according to their significance regarding the fulfilment of the overall aim of the technical system. Main functions typically represent the function of the system on a very abstract level. These are decomposed into sub functions. So called auxiliary functions enable particular sub functions (or principle solutions); e.g. controlling an electric motor will require electric energy to power the control unit. Typically, the representation of links between individual functions is limited to which sub function is part of a higher-level function.
The flow of operands is typically used to represent the *successive* transformations of one particular operand from input to output. Therein, the link between individual functions is represented by the output of one function serving as input of a successive function.
Flow related to time emphasises the sequence or parallelism of different (flows of) operations, regardless of the concerned operand. Although a flow of operands shows *successive* operations, the flow related to time is only implied, and not made explicit. Figure 4 illustrates the difference. Following the flow of operands in Figure 4a, the user of the model may derive how one particular operand is transformed within the system, but cannot derive the particular point in time when a transformation is taking place in relation to transformations in parallel flows. From the flow related to time shown in Figure 4b the relative point in time (and duration) of the different successive operations – here, the error consideration – can be derived. To account for cases in which the functional modelling perspectives or the way of structuring the representation are only implicit in a function model, the corresponding modelling perspective or way of structuring in Table 2 has been marked with an "o".

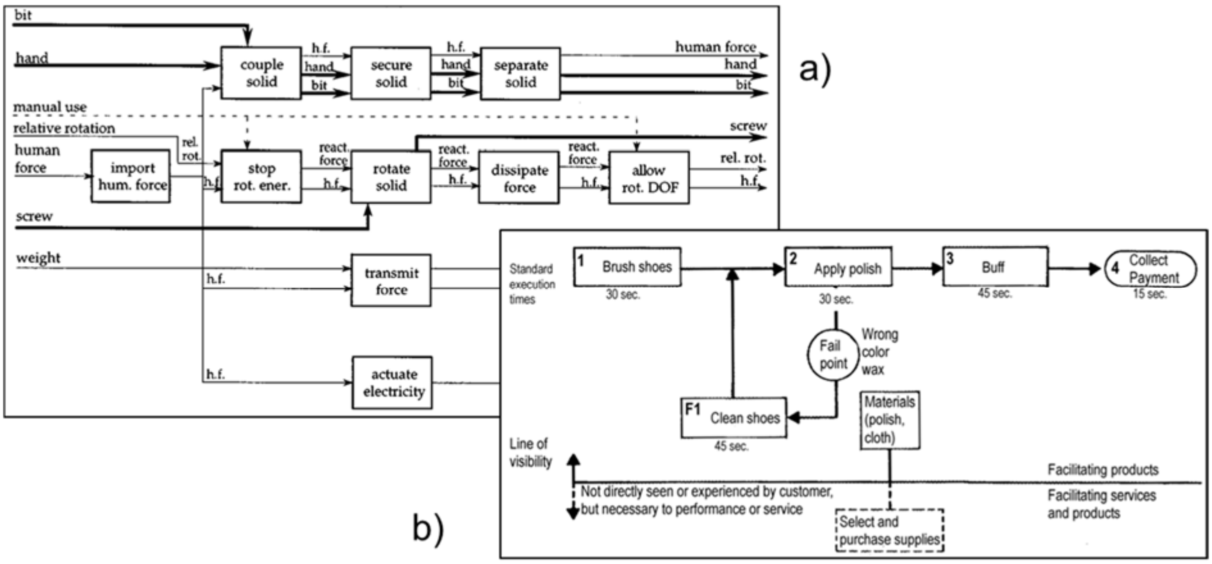

Figure 4. a) function structure after [Stone and Wood 2000] b) service blueprinting, after [Shostack 1982]

4.1.3 Identified functional modelling perspectives

The following main functional modelling perspectives were identified:

- *"states"*: Representation of the states a system can be in, or of the states of operands before (input) and after (output) an operation (e.g. transformation or technical process). The function model in Figure 5 represents the changed state after each operation.
- *"effects"*: Representation of the required physiochemical effects, which enable the transformation of one state into another state. Typical examples are the representation of *transformations* on an operand, see e.g. "change" operand E_2 (energy) in Figure 1.

- *"processes within the system"*: Representation of the processes executed by or within the technical system in order to change the state of the system or operands. *Technical* processes within the system require various physiochemical effects to take place to enable the processes. An example is Hubka's [1984] technical process structure. Detailing the individual technical processes will finally lead to a representation as basic physiochemical transformation processes within the system, i.e. the effects. There can be also *human* processes within the system, if e.g. a service is part of the technical system (e.g. in a PSS). An example is "brush shoes" in Figure 4b.
- *"interaction with the system"*: Representation of interaction processes of users (who are in this perspective not part of the system) or other technical systems with the technical system under development. A typical example is a service process model after [Watanabe et al. 2010].
- *"use case"*: Representation of different cases of applying the technical system. This is typically associated to the interaction of an actor (either the user or another technical system) with the technical system, which triggers, respectively requires subsequent processes to take place within the system. A typical example is a use case schematic (see e.g. Figure 3a).
- *"(potential) operating system"*: Representation of the role of an additional, or supporting technical system, which is supposed to perform or enable a sub-set of required *effects* or *processes*, either within the technical system or by interacting with it. A typical example model is Hubka's [1980] technical process structure.
- *"stakeholder allocation"*: Representation of the roles of different stakeholders, which may be users benefitting from a system or operators contributing to the system, e.g. through executing required processes or providing resources, etc. An example is a service process model after [Watanabe et al. 2010].

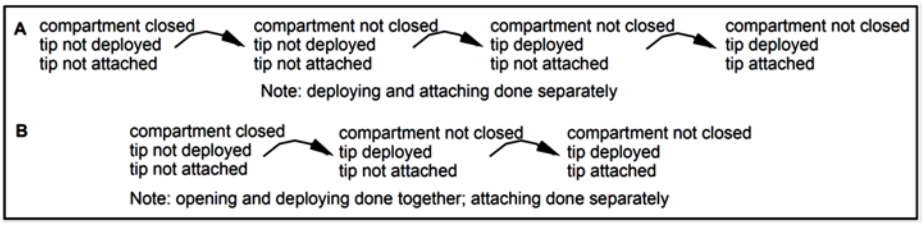

Figure 5. Two alternative process structures [Blessing and Upton 1997]

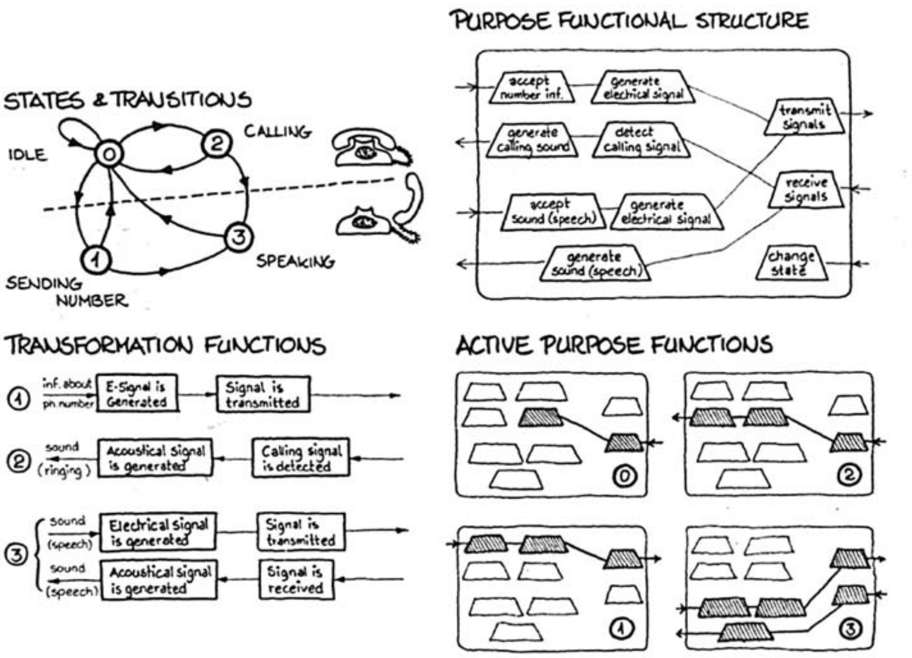

Figure 6. Different models representing functions of a telephone after [Buur 1990]

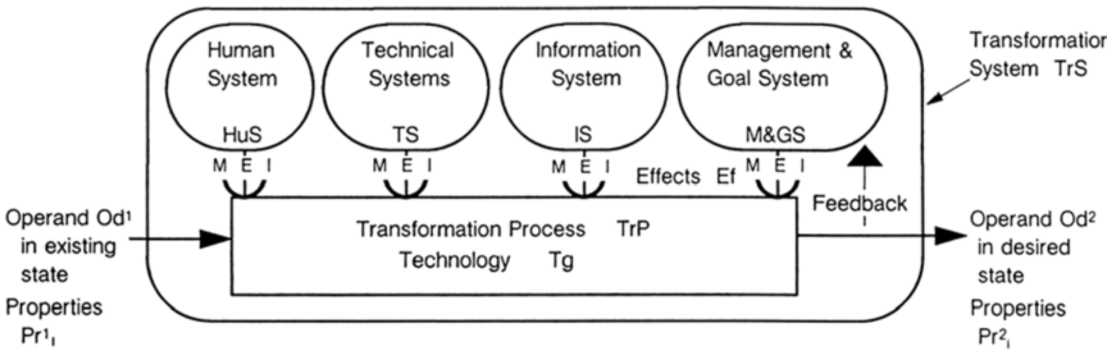

Figure 7. Technical process model after Hubka, taken from [Hubka and Eder 1988][2]

In three of the analysed function models additional approaches have been identified, which particularly aim at supporting the reasoning about the functions of a system:

- *Explicit function sharing considerations*: Blessing and Upton's process structure [Blessing and Upton 1997] emphasises considerations towards combining different state transition steps based on evaluating alternative flows of the required changes of states (see Figure 3).
- *Inclusion of conditions and quantities*: Specific characteristics of a system's task or operand may prevent certain principle solutions a-priori. For example, transporting an object which weighs two tons cannot be performed manually. Pahl et al. [2007] suggest considering quantities into the function formulation process, while Salminen and Verho [1989] explicitly include these into their state transition diagram.
- *Combination of system structure and functional modelling*: Most system development projects are carried out as variant design based on an existing system structure. Hence, Möhringer proposes to integrate functional modelling into the existing system structures (see Figure 8), by using a tree hierarchy of principle solutions and purpose-related functions. Other than function-means trees, modelling starts based on the structure of an existing system.

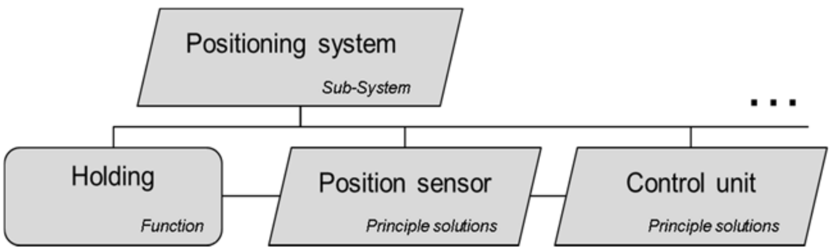

Figure 8. Representation of functions and principle solutions, simplified after [Möhringer and Gausemeier 2001]

4.2 Discussion

The functional modelling perspectives identified in the analysed models seem to confirm that the functions of a technical system can indeed not be represented by transformations related to operands alone, as discussed in Section 2. Rather, function models address different use cases, technical processes in manufacturing machines and service processes within a service (processes within the system) as well as interaction with the system and allocation of (potential) operating systems and stakeholders (e.g. users or operators). An approach integrating all these different modelling perspective could support the establishment of a shared understanding between different designers.

In order to do so, the literature review revealed the need to investigate in more detail *what* is represented in the different function models, in addition to *how* it is represented.

[2]The illustrated model from [Hubka and Eder 1988] refers to "technical processes" as it is used in this paper and originates from [Hubka 1984] as "transformation processes". The model from [Hubka 1984] is not shown here, as it is only available in German language.

Function models from mechanical engineering design mainly seem to represent the functions of a technical system related to transformations (or required effects) to be performed by the technical system, either hierarchically structured or by flow of operands. One interesting exception is Hubka's model, which will be discussed further down. Functional modelling in building design in general seems to be limited to list-like hierarchical representations and, hence, does not provide inspiration to technical system development. Software and PSS development particularly emphasise processes (at a specific point in time), including a representation of how users will interact with the system, the processes need to take place within the system to enable interaction and how to react on errors in the system or process, as illustrated in Figure 4b. As highlighted in chapter 1, services for the developed technical systems are increasingly often included into business models, user satisfaction has become more important as part of more value-oriented system development, suggesting the necessity of including the user's interaction into function formulation for all kinds of technical systems. In PSS design most function models have been adapted from service design approaches.

None of the reviewed models addresses all of the identified functional modelling perspectives. The model addressing most perspectives is Hubka's technical process model (see Figure 7). The multiple addressed perspectives, however, make the model rather complex and the distinction between different perspectives may become blurred, which may hinder a clear function formulation [Matzen 2009]. Furthermore, user interaction processes are not explicitly included in the model, requiring for it to be expanded, in order to integrate all the different functional modelling perspectives.

Another possibility for an integrative modelling approach is provided by [Buur 1990] and [Salminen and Verho 1989], who propose multiple complementary function models, covering a large variety of functional modelling perspectives. As an example, Figure 6 illustrates the function models proposed by Buur. Table 2 suggests that additional complementary models would still be required for both approaches to cover all the identified perspectives.

Having one model covering all the identified functional modelling perspectives may become confusingly complex very quickly, as it is the case with Hubka's model. Using multiple complementary models, on the other hand, may support comprehension of the individual model, as focus can be put on a limited number of functional modelling perspectives within each model. Furthermore when using multiple models, functional modelling perspectives, which are not relevant for the development of a particular technical system, can be left out rather easily so as to reduce of the effort required for the function formulation and the complexity of the model. However, integration of the different functional modelling perspectives – within as well as across disciplines – may be hindered when continuing using separate models, in particular as this is a complex cognitive task. Similarly, the establishment of a shared understanding across disciplines may not be sufficiently achieved, when using separate models.

5. Conclusions

As the main design decisions are taken when conceptualising the system, a shared understanding of the required functions is essential for the development of truly integrated technical systems. Integrative functional modelling may serve as a basis for the establishment of a shared understanding across disciplines. Although current literature discusses this issue, it does provide little support. Various authors suggest that this is due to an ambiguous understanding of function by individual designers and researchers, resulting in a large diversity of functional modelling approaches. As a consequence, different function models address diverse functional modelling perspectives. The presented analysis and categorisation of function models existing in different disciplines led to the identification of several functional modelling perspectives, which need to be integrated in order to develop an integrative functional modelling approach able to support technical system development (technical products and PSS). The analysis also identified various issues that need further investigation before an integrative approach can be developed.

First of all, research is needed to determine the benefits of having one model, covering all the identified functional modelling perspectives, as opposed to using multiple models. Research is also needed to address which function models – and hence, which functional modelling perspectives – are de-facto relevant to designers in industry, and whether functional modelling perspectives not typically

used within one particular discipline (e.g. use cases from software development) can be beneficial to other disciplines.
As a second step, it needs to be investigated how functional modelling in interdisciplinary technical system development can be realised and what kind of modelling support is indeed helpful to designers in practice. This includes research on the particular barriers and enablers for the establishment of a shared understanding based on particular functional modelling approaches. Eventually, a framework needs to be developed supporting and structuring the process of integrative function formulation and functional modelling.

Acknowledgement

The authors would like to thank the *Fonds Nationale de la Recherche* Luxembourg for funding this research (project: AFR PhD-09-186) as well as Prof. Mogens Myrop Andreasen for valuable discussions.

References

Ahmed, S., Wallace, K., "Evaluating a Functional Basis", Proceedings of ASME Design Engineering Technical Conferences (DETC), 2003

Aurisicchio1, M., Eng, M., Ortíz Nicolás, J.C., Childs, P.R.N., Bracewel, R.H., "On the Function of Products", Proceedings of 18th International Conference on Engineering Design, ICED 11, 2011

Blessing, L. T. M., A Process-Based Approach to Computer-Supported Engineering Design, 1994

Blessing, L., Upton, N., "A Methodology for Preliminary Design of Mechanical Aircraft Systems", Proceedings of 1997 World Aviation Congress, 1997

Brown D.C. and Blessing L., "The relationship between function and affordance", Proceedings of IDETC/CIE Conference, 2005, DECT2005-85017 (ASME)

Buur, J., "A Thoeretical Approach to Mechatronics Design", PhD thesis, 1990

Buur, J., Andreasen, M. M., "Design Models in Mechatronic Product Development", Design Studies Vol. 10, Issue 3, 1989

Carrara, M., Garbacz, P. Vermaas, P.E.. "If engineering function is a family resemblance concept: assessing three formalization strategies", Applied Ontology, Vol. 6, Issue 2, 2011

Chandrasekaran, B., Josephson J.R., "Function in device representation", Engineering with Computers, 2000, Vol. 16, pp.162-177

Crilly, N., "The Role that Artefacts Play: Technical, Social, and Aethetic Functions",Design Studies, Vol. 31 , Issue 4, 2010, pp. 311-344

Erden M.S., Komoto H., Van Beek T.J., D'Amelio V., Echavarria E. and Tomiyama T., "A reviewof function modeling: approaches and applications", Artificial Intelligence for Engineering Design, Analysis and Manufacturing, 2008, Vol. 22, pp. 147-169.

Hubka, V., "Theorie Technischer Systeme", Springer, 1984

Hubka, V., Eder, W., "Theory of Technical Systems", Springer, Berlin, 1988

Matzen, D., "A Systematic Approach to Service Oriented Product Development", PhD thesis, 2009

Möhringer, S., Gausemeier, J., "Integration der Funktions- und Prinziplösungsmodellierung Mechatronischer Systeme", Proceedings of 12th Symposium "Design for X", 2001

Pahl, G., Beitz, W., Feldhusen, J., Grote, K.H., "Engineering Design – A Systematic Approach", Springer, 2007

Salminen, V., Verho, A.J., "Multi-Disciplinary Design problem in Mechatronics and some Suggestions to its Methodical Solution in Conceptual Design Phase", Proceedings the 7th International Conference on Engineering Design (ICED'89), 1989, pp. 533-554

Shostack, G. L., "How to Design a Service", European Journal of Marketing, Vol. 16, Issue 1, 1982

Stone, R., Wood, K., "Development of a Functional Basis for Design", Journal of Mechncial Design, Vol. 122, 2000, pp.359-370

Ullman, D. Blessing, L.T.M., Wallace, K., "Understanding Function and Function-to-From Evoluiton", Workshop Report, CUED/C-EDC/TR 12, 1992

van Eck, D., "On the Conversion of Functional Models", Research in Engineering Design, Vol. 21, 2010, pp. 99-111

Vermaas, P.E., "Accepting Ambiguity of Engineering Functional Descriptions", Proceedings of 18th International Conference on Engineering Design, ICED 11, 2011

Watanabe, K.; Mikoshiba, S.; Tateyama, T.; Shimomura, Y.; Kimita, K. (2011): Service Design Methodology for Cooperative Services. In Proceedings of the ASME IDETC/CIE 2011.

Boris Eisenbart
Engineering Design and Methodology Lab, Université du Luxembourg
6, rue Richard Coudenhove Kalergi, L-1359 Luxembourg
Email: Boris.Eisenbart@uni.lu