A DSM-based Framework for Integrated Function Modelling
Concept, application and evaluation

Eisenbart, Boris; Gericke, Kilian; Blessing, Lucienne T. M.; MCAloone, Timothy C.

Important note
To cite this publication, please use the final published version (if applicable).
Please check the document version above.

CrossMark

ORIGINAL PAPER

A DSM-based framework for integrated function modelling: concept, application and evaluation

Boris Eisenbart[1] · Kilian Gericke[2] · Lucienne T. M. Blessing[3] · Timothy C. McAloone[4]

Abstract Function modelling is proposed in the literature from different disciplines, in interdisciplinary approaches, and used in practice with the intention of facilitating system conceptualisation. However, function models across disciplines are largely diverse addressing different function modelling perspectives and using different structures and forms for representing the contained information. This hampers the exchange of information between the models and poses particular challenges to joint modelling and shared comprehension between designers from different disciplines. This article proposes an integrated function modelling framework, which specifically aims at relating between the different function modelling perspectives prominently addressed in different disciplines. It uses interlinked matrices based on the concept of DSM and MDM in order to facilitate cross-disciplinary modelling and analysis of the functionality of a system. The article further presents the application of the framework based on a product example. Finally, an empirical study in industry is presented. Therein, feedback on the potential of the proposed framework to support interdisciplinary design practice as well as on areas of further improvement has been obtained from participants working in industry.

Keywords Function modelling · DSM · Interdisciplinary product development · Conceptual design · Empirical study

✉ Boris Eisenbart
b.eisenbart@tudelft.nl

1 Product Innovation Management Department, Delft University of Technology, Landbergstraat 15 - Building 32, Rm B-4-100, 2628 CE Delft, The Netherlands

2 Research Unit in Engineering Sciences, University of Luxembourg, Rue Richard Coudenhove-Kalergi 6, 1359 Luxembourg, Luxembourg

3 SUTD–MIT International Design Centre, Singapore University of Technology and Design, 8 Somapah Road, Singapore 487372, Singapore

4 Engineering Design and Product Development Section, Technical University of Denmark, Nils Koppels Allé, 2800 Kgs. Lyngby, Denmark

1 Introduction

The need for realisation of a rising number of functions in newly developed systems is a continuous stimulus for companies to search for new technologies, new ways to utilise existing technologies and increasingly often, to combine products with complementary services supporting function fulfilment. This leads to a higher level of technology integration in technical products as well as to combined offerings, such as so-called product-service systems (PSS). PSS are "system[s] of integrated products and services that companies develop and deliver, in order to fulfil a need with their customers" (Tan 2010, p. 27). The development of such integrated solutions necessitates close collaboration of experts from various design disciplines (Erden et al. 2008; Matzen 2009). Apart from more classical engineering disciplines, this includes software and service development as well as, e.g. mechatronics or systems engineering. Interdisciplinary collaboration is particularly important during conceptual design (Andreasen and Hein 2000; INCOSE 2010) requiring a joint effort of involved designers and thus a continuous exchange of information between them (Chakravarthy et al. 2001; Shai and Reich 2004). Therefore, a shared understanding of the problem and the emerging solution alike needs to be established in the design team (Kleinsmann and Valkenburg 2008; van Beek and Tomiyama 2009).

Function modelling supports solution finding early in the design process and on an abstract level (Chakrabarti and Bligh 2001). Its application in a large variety of disciplines predestines function modelling for supporting conceptual design of multi-technology systems. Erden et al., for instance, suggest that "the barriers between […] disciplines can be overcome by using [a] common language of functionality" (Erden et al. 2008, p. 147). This is similarly emphasised by Tukker et al. (2006) and Müller et al. (2007) for the integration of engineering disciplines with service development in PSS design. However, function modelling as a means to support interdisciplinary design teams is not widely applied thus far (Vermaas 2013; Eckert 2013). One problem could be that while function modelling seems common, for instance, in electrical engineering and software development, there is little evidence that it is applied to a similar extent in mechanical engineering practice (see, e.g. Bonaccorsi et al. 2009; Fantoni et al. 2009; Eckert 2013). A more fundamental problem, however, is that function models from different disciplines frequently differ in terms of addressed content, terminology and morphology (i.e. their structure and form). Furthermore, the underlying concept of function oftentimes diverges between different modelling approaches (see, e.g. Erden et al. 2008; Alink et al. 2010; Eisenbart et al. 2012, 2013a). Therefore, in spite of its large potential to facilitate integration in interdisciplinary system development, a "common language of functionality" has not—or not sufficiently—been attained. Exploiting the full potential of function modelling in terms of supporting interdisciplinary design requires adequate advancement from existing models, given that these typically originate from less integrated applications.

In this paper, a new approach for function modelling, the integrated function modelling (IFM) framework, is introduced as a means to integrate the perspectives of different disciplines on a system's functionality. It is ultimately intended to provide a practical support for joint function modelling in the development of technical systems, services and PSS. The following section briefly describes the existing diversity related to function modelling within and across disciplines. The obtained insights are used to determine distinct properties that an adequate integrative function modelling approach ought to provide modellers (i.e. designers from different disciplines) with. In Sect. 3, the IFM framework and its application are explained using the example of a mechatronic system including appendant services. Finally, Sect. 4 presents the initial evaluation of the IFM framework in industry. The article concludes with a discussion of the obtained feedback and implications for further development of the framework.

2 Towards integrated function modelling

At the beginning of a design project, typically, neither the problem nor the desired solution are thoroughly defined (Braha and Reich 2003). Conceptual design hence cannot seamlessly move from a problem to a solution. Instead, it is a continuous process involving iterative analysis and evaluation steps leading to a gradual increase in information about the addressed problem in parallel to information about the emerging solution. This process is typically referred to as "co-evolution" (Simon 1973; Poon and Maher 1997; Maher and Tang 2003). Function modelling is proposed in textbooks to support and guide the designers' reasoning in this particular transition (Far and Elamy 2005). Furthermore, it enables an individual's conceptual design considerations to become explicit and thus accessible to others for discussion during joint solution synthesis (Fowler 1998).

2.1 Diversity of function as a concept

A variety of frequently overlapping or even contradictory definitions of function can be found in the literature.[1] The definitions not only vary in terminology but most importantly with respect to the specific notions of function they convey, i.e. the underlying perception and meaning given to function as an abstract concept. Several of them, for instance, refer to function as the *ability* of a system to achieve a goal or fulfil a given task by showing certain *behaviour* (see, e.g. Roozenburg and Eekels 1988 or Buur 1990). Other authors define function as an intended or required *transformation*, *conversion* or *change of states* of distinct operands (i.e. typically specifications of material, energy or signals; see, e.g. Rodenacker 1970; Fowler 1998; Cockburn 2000 or Pahl et al. 2007). Finally, many authors refer to function to be equal to (Ropohl 2009; Ullman 2010) or derived from (Sakao and Shimomura 2007) the *purpose* of a system, respectively, in terms of fulfilling a goal which is a teleological interpretation of function.

All of these definitions have their place in design research insofar as they relate to a specific interpretation of function as a concept. Some scholars even provide more than one definition of function with respect to certain particularities related to functionality of a system or entity they want to discern (see, e.g. Chakrabarti and Bligh 2001; Kruchten 2004; INCOSE 2010). Hubka and Eder, for instance, differentiate between "purpose functions", which refer to the desired output effects as the overall task

---

[1] Comprehensive overviews and discussions are provided, e.g. in Warell (1999), Chandrasekaran and Josephson (2000), Maier and Fadel (2001), Chiang et al. (2001), Deng (2002), Chandrasekaran (2005), Vermaas (2009), Crilly (2010), Carrara et al. (2011), Aurisicchio et al. (2011, 2012), or Goel (2013).

assigned to a technical system, and "technical functions", which explicate the (required) capability of the system to fulfil its designated task (Hubka and Eder 1988, pp. 60–61 and p. 72). By this separation, the authors create different *types* or *primitives* of function in relation to different viewpoints taken. Both notions of function that Hubka and Eder propose use the idea of a purposeful conversion of distinct inputs to outputs as underlying perception of the functions of a system. This concept, though widely adopted in the literature, is often criticised for being too product-oriented. Several scholars call for a more universal interpretation (see, e.g. Umeda and Tomiyama 1997; Warell 1999; Maier and Fadel 2001; Crilly 2010). Chandrasekaran and Josephson (2000; see also Chandrasekaran 2005), for instance, differentiate between a *device-centric* and an *environment-centric* perspective on function. The former is related to the concrete behaviour of the system under consideration. The latter refers to the desired external effect that is realised by the system, which may also include humans and their interactions with the system. Similar differentiations are proposed, e.g. by Chittaro and Kumar (1998) or Deng (2002). Crilly (2010) provides a particularly comprehensive review of definitions of function and differentiates three main types: *technical functions* (related to the actual physical properties and behaviour of a system), *social functions* (related to overall effects in a user's social context), and *aesthetic functions* (e.g. "convey beauty"; see also Aurisicchio et al. 2011). More generally speaking, these scholars try to provide some sort of explanatory framework in order to grasp and relate between the different facets function as a concept can address. An inherent problem is, however, that the particular viewpoint taken can in fact vary. Logically, one may simply switch focus between sub-systems, components, users, etc., in relation to a system under consideration. This will then consequently change the notion of function relating to the respective devices and, in extension, also their roles in function fulfilment (see Nevala 2005 and Crilly 2015).

One thing that becomes apparent in this discussion is that there is no consensus on what function as a concept specifically entails. Based on comprehensive reviews, Vermaas (2009, 2011) and Carrara et al. (2011) similarly conclude that "[…] 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, p. 98). Vermaas (2009) further proposes a set of three notions of function that he considers to be archetypical in the sense that any definition of function provided by scholars, in the end, can be referred back to one or more of the following:

1. *behaviour-related notion*: function as the intended behaviour of an entity;

2. *outcome-related notion*: functions as the desired effects of the behaviour of an entity;

3. *task- or goal-related notion*: function as the purpose for which an entity is designed.

The latest is closely related to the particular uses that the system is intended to be put to. In addition, Vermaas discusses the concept of *capability* of a system or artefact—through its particular structure—to show a certain behaviour. This, in turn, enables a user to fulfil alternative goals or use plans with it (i.e. an intention-oriented perception of function, see particularly Houkes and Vermaas 2010; Vermaas 2013). Following these discussions, in this article, opposed to so-called affordances, function is considered to be something deliberately designed into a system. Affordances cover the entirety of uses (intended and unintended by the designers) that a system can be put to due to the specific characteristics (after Weber 2007) it possesses (Brown and Blessing 2005). To give an example, the main function of a shoe is to support a foot while walking, which means distributing forces comfortably, protecting it from sharp objects, etc. An affordance related to the shoe would be that, due to its solid structure, it can also be used to keep a door ajar, which is beyond its intended purpose.

2.2 Diversity in function modelling

Although function modelling is also particularly prominent in electrical engineering and software development, a considerable amount of function models originates from German-speaking mechanical engineering research conducted in the 1960s and 1970s (see particularly Rodenacker 1970; Pahl and Beitz 1977 or Hubka 1980). These usually represent function as verb/noun combinations related to a transformation of the states of basic operands between the input and the output of a system, as exemplified in Fig. 1. Therein, inherent transformation processes are linked together by relevant flows of these operands. These kinds of models have been widely adopted, particularly in the mechanical engineering literature (see, e.g. Ullman 2010; Roozenburg and Eekels 1995; Stone and Wood 2000; Ulrich and Eppinger 2008) but also in a few interdisciplinary design approaches (see, e.g. VDI 2004 and Cross 2008) and even for abstract modelling of biological systems (see Nagel et al. 2008).

The function structure after Pahl et al. (2007), in a way, has become a standard convention in the mechanical engineering literature and beyond (Aurisicchio et al. 2012). However, even within the mechanical engineering literature, a large diversity of alternative function modelling approaches can be found, such as the function–behaviour–structure framework (Gero 1990), structure–behaviour–function model (Iwasaki et al. 1993), Schemebuilder
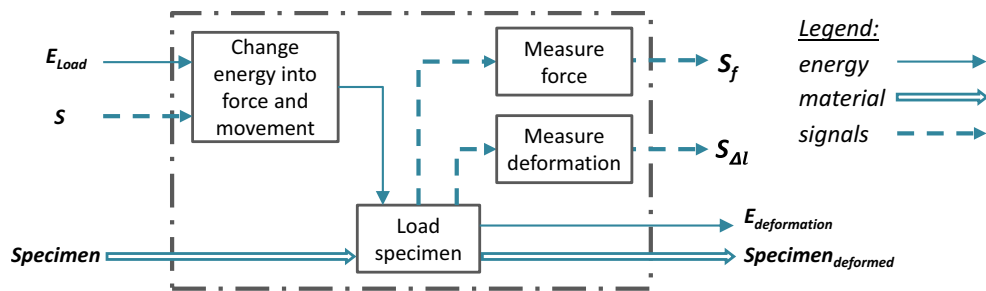
Fig. 1 Example of a function structure of a tensile testing machine (Pahl et al. 2007)

(Bracewell and Sharpe 1996), the function–behaviour–state model (Umeda and Tomiyama 1997), or the conglomerate approaches by Tjalve (1978) or Hubka and Eder (1988; Eder and Hosnedl 2008). The diversity increases tremendously when further disciplines are considered. While function models from mechanical engineering primarily employ hierarchical breakdowns or flows of operands as means of structuring the representation of functions and their dependencies, function modelling in software development, service development and product-service system (PSS) mainly build on a flow in time. Examples of such models include use case modelling (see, e.g. Kruchten 2004 and Weilkiens 2008), service blueprinting (Shostack 1982), IDEF-0 (USDoD 2001) or service process modelling after Watanabe et al. (2011, see Fig. 2). In electrical engineering, function modelling particularly focuses on distinct states and their transitions, for instance, using finite state machines, petri nets, etc. (see, e.g. Scheffer et al. 2006 or Dewey 2000)[2].

Apart from the particular representational aspects associated with how functions are linked, which were just discussed, more importantly, reviews by the authors of this article show that the particular inherent contents differ considerably between models. This will be discussed in the following.

2.2.1 Function modelling perspectives and morphologies

A comprehensive review of 76 function models[3] found in the literature (61 original models plus 15 variants proposed by different authors) by Eisenbart et al. (2012, 2013a, c) and a detailed analysis of 24 function models used in ten

companies (Eisenbart 2014) led to the identification of a set of distinct *function modelling perspectives* as well as specific *modelling morphologies* addressed and used, respectively, in function modelling within and across disciplines. Function modelling perspectives refer to the particular information (i.e. the concrete content) represented in a function model. Seven distinct function modelling perspectives were identified which are described in Table 1. Modelling morphologies refer to the way represented information is structured; this conveys information about how individual functions are linked or are dependent on one another. Essentially, information may be structured *hierarchically*, related to a *flow of operands* (e.g. in Fig. 1) or related to a *flow in time*.

In a few models, additional contents were found supporting the solution finding process and/or the reasoning about specific aspects of system functionality. These additional contents include

- *constraints and target values* for function execution (e.g. allowed performance deviation and required torques) and
- *impacts from/on the environment* (e.g. disturbances affecting function fulfilment);
- *bilateral impacts* and dependencies between allocated solution elements

Some modelling perspectives are more prominent than others depending on the discipline the respective function models stem from. For instance, while function models from mechanical engineering mainly address *transformation processes* and *effects* related to a flow of operands (see, e.g. Fig. 1), software development, PSS design, and systems engineering were found to primarily focus on modelling the flow of *transformation* and *interaction processes* (typically discerned into different *use cases*) related to *stakeholders* and *technical sub-systems* based on their sequences in time. The analyses further reveal a large diversity related to the proposed modelling processes. This refers to the specific way that authors propose in their textbooks for particular function models to be set up and gradually detailed (Eisenbart et al. 2012, 2013a, c).

---

[2] Albeit not all of these models are explicitly referred to as function models in the literature, they nevertheless serve the purpose of supporting the transition from a design problem to a potential solution concept during conceptual design (see Eisenbart et al. 2012). Reviews of function modelling approaches are found, e.g. in Chakrabarti and Bligh (2001), Chandrasekaran and Josephson (2000), Far and Elamy (2005), Erden et al. (2008), and van Eck (2010a, b).

[3] Models originate from mechanical engineering ($n = 20$ models), electrical engineering ($n = 8$), software development ($n = 10$), service and PSS design ($n = 16$), mechatronics ($n = 12$) as well as systems engineering ($n = 10$).

Fig. 2 Schema of a service process model after Watanabe et al. (2011)
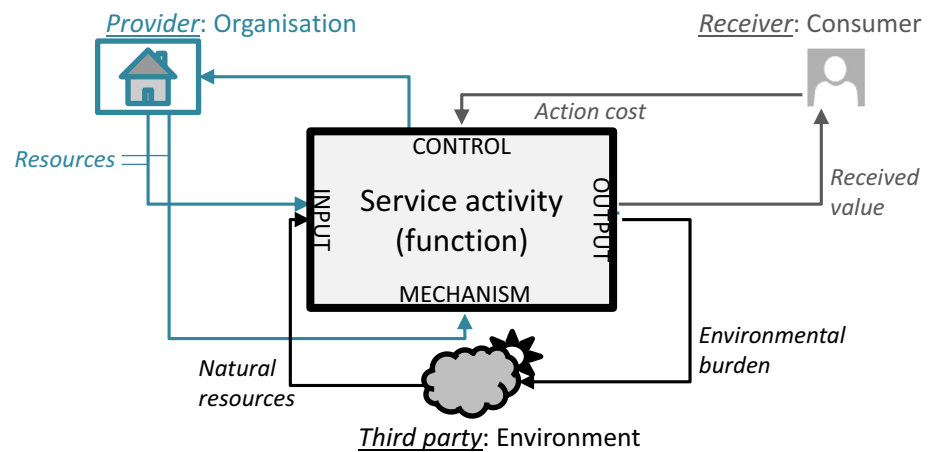

Table 1 Central function modelling perspectives (adapted from Eisenbart et al. 2013a)

States	Representation of the states a system can be in or of the states of operands before (input) and after (output) a transformation process
Transformation processes	Representation of the processes executed by the function carriers (technical products, stakeholders, etc.) that—from the designers' perspective—are part of the system under development and which may or may not result in a change in state of the system or of operands. Therein, *technical processes* are transformation processes executed by technical systems (technical products, devices, etc.), whereas *human processes* are executed by stakeholders involved in function fulfilment (this explicitly includes human activities, e.g. during service execution)
Interaction processes	Representation of interaction processes of stakeholders or of other technical systems, which—from the designers' perspective—are *not* part of a system, with stakeholders or technical systems, which *are* part of the system under consideration
Effects	Representation of the required physiochemical effects, which have to be provided to enable, respectively, support, the transformation processes that change the state(s) of operands and/or of the system into (a) new state(s)
Use cases	Representation of different scenarios of applying the technical system for a specific purpose (e.g. fulfilling a goal, changing the state of the system or user); this is typically associated with the interaction of stakeholders or another technical system with the technical system under development (interaction processes), which triggers, respectively requires, subsequent processes to be carried out by the system
Technical system allocation	Representation of the role of technical products, their sub-systems or any other kinds of (tangible or intangible) technical means acting as function carriers in performing or enabling one or more functions; these technical means may be either part of the system under consideration or interact with it
Stakeholder allocation	Representation of the roles of different stakeholders (humans or other animate beings), which may be users benefitting from a system or function carriers contributing to the system, e.g. through executing required processes or providing resources, etc

2.2.2 Hampered integration

A few reviewed function modelling approaches address a relatively large proportion of the identified function modelling perspectives and morphologies. Examples are the approach by Tjalve (1978); Hubka and Eder (1988; Eder and Hosnedl 2008), the Object-Process Methodology (Dori 1995), diagrams from Unified Modelling Language (UML) or System Modelling Language (SysML, OMG 2012), respectively, or the SAPPhIRE model (Chakrabarti et al. 2005). The approach by Hubka and Eder has further been expanded by Matzen (2009) to include service-related along with product-oriented functions, in order to enable abstract modelling of PSS. Therein, products and product use within services are modelled as alternative use cases

(or duty cycles, respectively). However, none of these extant approaches addresses the whole set of the identified function modelling perspectives and morphologies. In addition, they are usually fairly specific in their suggested application and also in terms of the particular notion of function followed (as discussed in Eisenbart 2014; Eisenbart et al. 2015a). Erden et al. (2008) and van Eck (2010b) argue that the large variety of extant function models is in itself proof for the diverse demands that individual designers have in terms of representing functions. This leads to the conclusion that direct integration of existing models or transfer of information between them, respectively, may not be possible (Erden et al. 2008) or not even be sensible, given that they have a specific focus that is different from other approaches' (van Eck 2010b).

The differences between existing function models are considered to be the main reason why integrated function modelling could not be attained thus far. However, consolidating the findings from the literature review and studies in industry by Eisenbart et al., an essential finding is that the function modelling perspective of *transformation processes* is the most prominent perspective within all reviewed disciplines. In most cases, these are modelled in relation to a flow in time, especially in the models found in practice (Eisenbart 2014). Eisenbart et al. (2013a) particularly stress that this can provide an opportunity for eventually building a basis for integration or consolidation of function modelling within and across disciplines. In the following, main research endeavours already undertaken related to supporting shared function modelling are briefly discussed.

2.3 Supporting shared function modelling

One particular endeavour in engineering design research to support shared function modelling is to increase the clarity of the generated models by using distinct semantic expressions in representing functions as well as their relations.[4] Related research typically employs one of two possible approaches pertaining to tackling the issue of divergent definitions of function:

- *introducing formalisation* in the representation of functions and their relations;
- *converging* to a common representation of function by *comparing* existing function modelling approaches in an attempt to *consolidate* an adequate *function ontology*.

Function ontologies, in essence, try to discriminate clearly between different aspects entailed in or related to function as a concept, respectively, in order to reduce ambiguity. Considerable research has been conducted resulting in numerous approaches for formalising the representation of functions. These typically employ specific models in conjunction with function taxonomies,[5] i.e. "a standard language of function" (Ahmed and Wallace 2003, p. 1), in order to raise clarity in the communication about functions (Kurfman et al. 2003; Sen et al. 2013). In other words, by introducing precision in what specific textual formulations and related visual representations semantically entail, the intelligibility of the generated models should be enhanced. Two such ontologies may be

highlighted as they integrate a particularly large variety of aspects related to function in design: the SAPPhIRE model (see, e.g. Chakrabarti et al. 2005) as well as the function and device ontology by Kitamura and Mizogushi (2007, see also Kitamura et al. 2004). Kitamura and Mizogushi's research ultimately aims at creating a design knowledge database through ontological systematisation of relevant information, in particular when it comes to describing products, their components and the particular functions these carry. Therein, function is put in a clearly hierarchical relation with the product's structure and behaviour in terms of a "by means of" relation. That is to say, a product fulfils its main purpose *by means of* the function it possesses, *by means of* the behaviour it shows, eventually, *by means of* interconnected components enabling its very behaviour (c.f. Sect. 2.1). The approach sets aspects pertaining to product functionality, particularly state changes in operands, in a systematic context with each other and with the functioning of the overall product, while keeping them conceptually distinct. The information is linked by formal expressions to minimise ambiguity and allow storage in a knowledge database. It is this high level of formalisation that sets this work apart from similar endeavours by scholars like Hubka and Eder (1988), Gero (1990), Iwasaki et al. (1993), Umeda and Tomiyama (1997), and others. The SAPPhIRE model similarly proposes ontological systematisation of functional descriptions; however, it relates components (or parts) of a product-to-product functionality through explicating the physical effects and phenomena that a conglomerate of functionally interrelated parts (i.e. the organs) jointly create. This goes beyond what is entailed in Kitamura and Mizogushi's work. The outcome, again, is described by a change in state of associated operands that eventually leads to an impact on, through the interaction with, the product's environment. These and other ontologies have been successfully applied in practice (see, e.g. Srinivasan et al. 2011; Kitamura et al. 2004; Sen et al. 2013). They are all attempts to provide designers with means for clearly describing relevant information of product functionality, but differ in the specific way this information is broken down and interrelated.

Regarding the second main endeavour, numerous reviews of existing function modelling approaches and definitions used can be found in the literature which eventually aim at converging on one (or few) common denominator(s) in function modelling. These would then serve as a modelling basis in shared, cross-disciplinary function modelling (see particularly Erden et al. 2008, but also, e.g. Chandrasekaran and Josephson 2000; Chakrabarti and Bligh 2001; Garbacz et al. 2011; and Srinivasan et al. 2012). Nevertheless, a common approach for modelling functions could not be attained thus far. Vermaas (2011,

---

[4] Comprehensive discussions are provided, e.g. by Chakrabarti and Bligh (1990), Chandrasekaran and Josephson (2000), Chandrasekaran (2005), Kitamura and Mizogushi (2007).

[5] Examples can be found in Szykman et al. (1999), Stone and Wood (2000), Hirtz et al. (2001), Ookubo et al. (2007), Sen et al. (2010), (2013).

2013), Carrara et al. (2011), and similarly Garbacz et al. (2011) argue that convergence is not possible due to the fact that modelling approaches proposed in the different disciplines are too semantically diverse.

More generally, in relation to applying functional ontologies and taxonomies, there is a controversy in terms of whether or not these can provide a broad audience of designers with the desired support. Prominently, Kurfman et al. (2003), Kitamura and Mizogushi (2007), and Sen et al. (2010, 2013) have been able to show an increase in clarity and intelligibility through use of function modelling ontologies and/or taxonomies in a mainly mechanical engineering design context. However, such approaches are also critically discussed, e.g. by Ahmed and Wallace (2003), van Eck (2010a, b) or Aurisicchio et al. (2012). The main point of criticism by these authors is that the vocabulary used in them is fairly restricted and forces designers to think in a rather abstract manner. Kitamura and Mizogushi (2007) reported on similar problems encountered while they implemented their ontology in practice. As part of the required abstraction, contextual information used for explaining particularities of individual functions can be lost. To give an example from the studies by Ahmed and Wallace, wherein functional descriptions that engineers in a company had formulated in a past project were formalised by use of the functional basis after Stone and Wood (2000), a function reading "supporting nozzle guide vane in axial or rotational or tangential locations" was consequently reformulated to state "support solid (rigid body)" (see Ahmed and Wallace 2003, p. 5). The new formulation no longer carries information identifying involved parts, their functional interrelation or the particular "support" being required. The discussed ontologies by Kitamura and Mizogushi, Gero, the SAP-PhIRE model, etc., may not necessarily encounter the same problem as contextual information is still provided through the relations to the product parts, related operands, etc. However, the preponderant focus on product parts and their contribution to function inherently leads these approaches to adopt an almost exclusively device-centric view. This widely precludes their seamless utilisation for functional descriptions of services and, in extension, service-related aspects of PSS.

2.4 Insights from research on the application of function modelling

Various scholars, for instance Blessing (1997), Kurfman et al. (2003) or Sen et al. (2010, 2013), describe experiences obtained from observing other researchers or students applying function modelling. Others, like Eckert and Alink (Alink et al. 2010; Eckert 2013) or Ahmed and Wallace (2003), conducted surveys or experiments concerning function modelling by practitioners in industry. It seems the practical application of function modelling depends on the overall design approach used, personal preferences of the involved designers and particularly the way new information is gradually generated during a design project. Furthermore, the scholars found that

- practicing designers tend to switch flexibly between different notions of function as well as ways of reasoning about and modelling functions;
- solution-neutral and inflexible function modelling (as often proposed in literature) is widely rejected or was found to lead to difficulties; in fact, designers tend to make assumptions about the potential solution and model functions accordingly in an iterative process (see particularly Blessing 1997, but also similarly Visser 1991 and Albers et al. 2010).

Regarding the first point, interestingly, the notions of function that the participants referred to in the experiments and surveys carried out by Eckert and Alink to a large extent correspond to the three archetypical notions of function after Vermaas discussed earlier in this article. In the studies, during modelling, several participants switched between considering the assumed inputs and outputs, expected behavioural aspects or particular purposes that the product as a whole or individual components were assumed to serve. This can further be interpreted as implicit switches between taking a device-centric or environment-centric view onto functionality (cf. Chandrasekaran 2005). It is concluded by some scholars that it is in fact imperative for designers to have flexibility in modelling functions, in order to support their reasoning concurrently, which means not to be forced to adopt a high level of abstraction or a single notion of function (Goel 2013).

2.5 Implications

When it comes to supporting function modelling in practice, it seems that two different schools of thought exist. The first school entails scholars working towards a clearly formalised description of function, eventually with the hope of establishing a (somewhat) computational function model. The aim behind these and similar endeavours is to provide unambiguous models, thereby facilitating clarity in modelling and, in extension, in communicating about them. The second school of thought is more inclined to look at how designers usually work with and draw benefit from function modelling in practice. These latter scholars typically highlight that function taxonomies and ontologies, by the large majority of practitioners, are perceived as too abstract to provide them with concrete support in finding solutions to given problems. Related empirical research suggests that practitioners tend to work around formalised

approaches or apply them less rigorously. In some cases, shared comprehension in design teams was in fact found to increase when natural language is used (see Eisenbart 2014).

The research presented here strives to provide practically applicable support to designers for the interdisciplinary development of complex technical systems, services and PSS. It is not intended to necessarily swing to either side of the highlighted discourse. Both endeavours have advantages in their own right, and despite the discussed criticism, the potentials that formalisation may offer in terms of providing clarity in the representation of function are large. However, seeing the arguments discussed earlier, whereby restrictedness in function modelling may effectively hamper designers while generating function models, a flexible and intuitive approach is considered beneficial. That means, following Goel (2013), designers should be able to reason about and to model functionality flexibly, which may include them implicitly switching between different notions and types of function in their considerations. Subsequent formalisation can still facilitate information handling and computing, which scholars like Kitamura and Mizogushi (2007) promote. Also, as it may be preferred to use extant taxonomies right from the start by different designers, integrated function modelling should remain open for these to be applied.

This article describes a novel approach for function modelling in interdisciplinary design work. The main intention is to enhance integration by linking the diverse contents (i.e. the function modelling perspectives and morphologies) found in function models from different disciplines, while allowing demand-specific application and change in the specific views onto system functionality taken. The found clear prevalence of representations of functions as flows of *transformation processes* in time lent itself as a vantage point in the development of such an approach (see Eisenbart et al. 2013a). In relation to the desired flexibility, it is expected that—depending on the specific disciplines involved, the designers' preferences as well as the specific course of the design project—different combinations of modelling perspectives and morphologies will be relevant at a time. This provides further incentives for integrated modelling to be set up in a way which makes it adaptable to the particular needs of designers and the rationales of different disciplines and companies.

3 A framework for integrated function modelling

In addition to what was discussed in the previous section, other properties that are vital for an interdisciplinary function modelling support to have include manageable complexity and consistency management pertaining to the represented information. The integrated function modelling (IFM) framework presented in the following tries to address these issues. It is based on the obtained insights from the discussed analyses of function models both from the literature and unpublished models developed by practitioners in different industrial branches. Observed strengths and shortcomings of these models and their application guided the development of the new modelling framework. In addition, continuous feedback from senior academics active in engineering design research, mechatronic system design, and software development as well as from practitioners in different industrial branches was considered. The result is a representational approach that is set up as a combination of modular matrices, using a combination of design structure matrices (DSM) or multi-domain matrices (MDM, see Kreimeyer and Lindemann 2011) and flow modelling. Matrices were selected as they provide a clearly structured representation that is expected to allow modelling and retrieving information quickly. Also, they are a relatively intuitive mean for modelling information; thus, it is expected that designers will be able to familiarise themselves with the framework quickly. An overview of the framework is provided in Fig. 3.

3.1 Entities and relations

The entities comprised in the IFM framework derive from the function modelling perspectives and complementary contents identified in the reviewed function models from textbooks and from industry (see Sect. 2.2.1). The final list of relevant entities as well as their respective definitions is provided in Table 2.

The specific relations between the entities comprised in the framework and their contribution to the functionality of a system under consideration are described in the following and illustrated in the UML-based class diagram in Fig. 4.
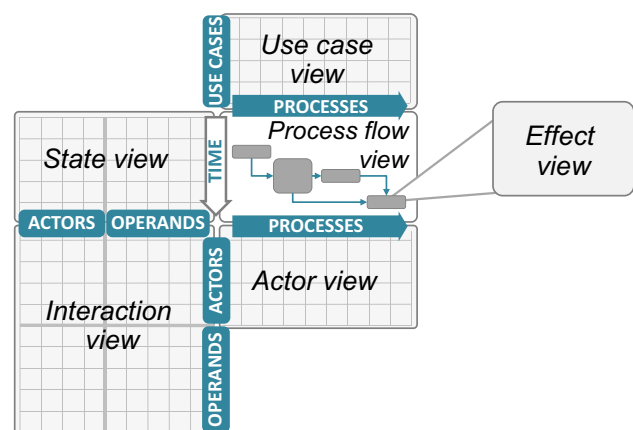

Fig. 3 Integrated function modelling framework

Table 2 Description of entities addressed in the IFM framework

| Entities	Description
Use case	Different scenarios for applying the system, which is usually associated with the interaction of actors with the system under development and may require subsequent transformation processes to take place. The outcome is typically an observable result (e.g. a change in state of related operands or of actors) providing value to users
Process	
Transformation process	Processes executed by actors that—from the designers' perspective—are part of the system under consideration and may lead to a change in state of actors and/or operands. *Technical processes* refer to transformation processes executed by technical means; *human processes* are related to stakeholders
Interaction process	Representation of processes executed by actors that—from the designers' perspective—are *not* part of a system and that include the interaction with actors that *are* part of the system under consideration
Effects	Representation of physiochemical effects or principles that are required or have to be provided, respectively, in order to enable or support the execution of transformation and/or interaction processes
States	Representation of the particular condition or state of affairs of actors and operands before (input) and after (output) a transformation process
Operands	Specifications or instances, respectively, of energy, material and information
Actor	
Stakeholder	Stakeholder comprises (groups of) animate beings affecting or being affected by the system under consideration
Environment	Environment includes all active and passive parts of nature in general surrounding the system under consideration
Technical sub-system	Technical sub-system encompasses technical systems (which may be combinations of mechanical artefacts, electrical or electronic systems or networks, and software systems, and may or may not have associated services) that are part of the system under consideration. They can be composed of more technical sub-systems
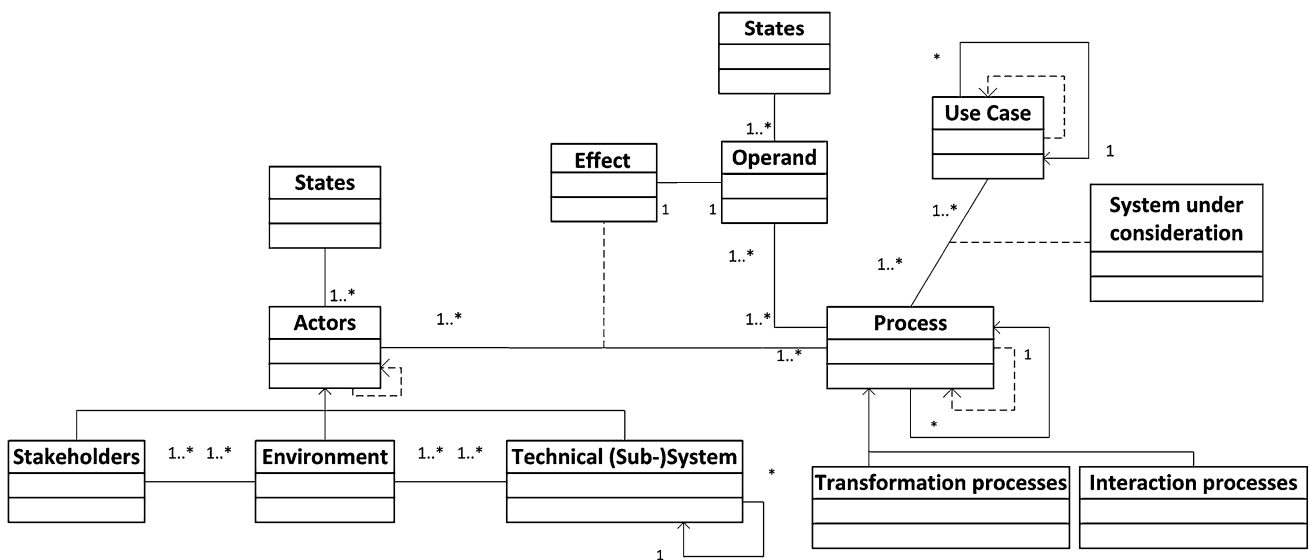
Fig. 4 Class diagram of the IFM framework

A system may support one or more use cases. Each use case may be decomposed into sub-use cases. Use cases may have dependencies among each other that may be bound by specific constraints (mutually exclusive, mutually inclusive, etc.—for all other situations, in Fig. 4, the dependencies shown are used to depict similar constraints). A use case may have one or more processes associated with it. Processes include transformation processes and/or interaction processes. There may be dependencies between individual processes, which may further be composed of sub-processes. A process may result in the transformation of the state of one or more operands and/or actors from a given state into another state. Processes which only indirectly contribute to a state change are regarded as supporting or auxiliary to the system functionality. Processes are enabled or supported, respectively, by effects. Effects are provided by actors. Actors—by providing the necessary effects—serve as operators or function carriers. Actors contain the sub-classes of stakeholder, technical (sub-) system and environment. They may also have

dependencies among each other. Operands may similarly interact and have dependencies among each other. Operands may temporarily assume the role of an actor during a use case by supporting the state transition of actors or other operands in relation to the execution of specific processes (as discussed by Nevala 2005 and similarly Crilly 2015).

With regard to these entity relations, the framework is respective of and relational towards the three archetypical notions of function proposed by Vermaas (see Sect. 2.1). It is centred on the representation of processes (see Table 2; Fig. 3) which lends towards the notion of function as the intended behaviour of a system under consideration. The representation of the states of operands and actors as well as their changes resulting from individual processes directly relates to the notion of function as the effects of the exhibited behaviour. Finally, different use cases relate to the particular applications that a system is put to, e.g. by a user according to his/her use plan (see Table 2), which corresponds to the notion of function as the purpose of a system or artefact (cf. Houkes and Vermaas 2010; Vermaas 2013).

3.2 Different views for representing system functionality

The framework consists of six central views, which represent the different entities and their dependencies: *process flow view*, *state view*, *actor view*, *use case view*, *effect view*, and *interaction view* (as indicated in Fig. 3). These views are strongly interlinked through mutually shared header rows and header columns in the specific, adjacently placed matrices forming them. They map different design information, in order to facilitate representation and analysis of specific dependencies between the represented entities. In the following, the individual views are described using the example of a mechatronic system, a customary coffee vending machine, which can provide a variety of warm drinks. It is complemented by an appendant service related to waste disposal, which is handled by a service operator. The variety of offered drinks represents different use cases associated with the vending machine. These include (among others) *prepare a cup of coffee*, *prepare a cup of cappuccino*, *prepare a cup of espresso*, *prepare hot tea water* and *automated cleaning*. The following descriptions for each view focus on the use case of "prepare a cup of coffee". Finally, Fig. 15 illustrates how the individual views relating to this use case are combined in the framework.

3.2.1 Process flow view

The *process flow view* is constitutive for the framework and centrally arranged (see Fig. 3). It represents the flow of

processes fulfilling one use case. The vertical direction visualises their flow in time allowing for indicating qualitatively, whether represented processes are to be carried out sequentially, in parallel or overlapping with each other (see Fig. 5; Process 3 and Process 4, for instance, are overlapping in time). Further, multiple (sequential) executions of the same processes in one use case can be represented (see Process 4 in Fig. 5). This vertical flow matches the flow in time of states from initial to final related to operands and actors in the *state view* (see Fig. 9). Processes are furthermore spread horizontally from left to right to enable a direct link to *actor view* and *use case view*. Reasonably, their horizontal order should follow their logical devolution in the function flow; however, there is a certain degree of freedom for modellers, for instance, to (re-)arrange processes that are starting in parallel to one another if this is more convenient or facilitates comprehension.

Figure 6 illustrates the flow of processes (P1 till P6) that are required for fulfilment of the use case "prepare a cup of coffee". Several of these processes contain further sub-processes. For instance, Process 1 "coffee is ordered" encompasses the sub-processes P 1.1–P 1.3 (as indicated by "zooming in" onto Process 1 in Fig. 6). Furthermore, quantities or constraints can be added to individual processes if preferred (see Process 2 "heat water"). As an alternative to this "zooming in" on individual processes, an eventual already existing service blueprint or a similar flow model can simply be attached to the view as an add-on, if preferred by the designers.

3.2.2 Actor view

The *actor view* indicates the specific involvement of one or more actors in the realisation of processes. Involvement can be, e.g. "affecting", "supporting" or "being affected by" a process (see Fig. 7). The view allows differentiating actors according to whether they—from the designers' point of view—belong to that part of the system that can be directly manipulated by the designer (e.g. the components to be designed, but also, e.g. humans and their activities as service operators in a PSS) or cannot, as, e.g. surrounding technical systems, the targeted users or external service providers.
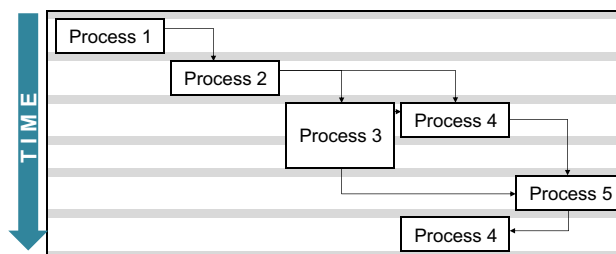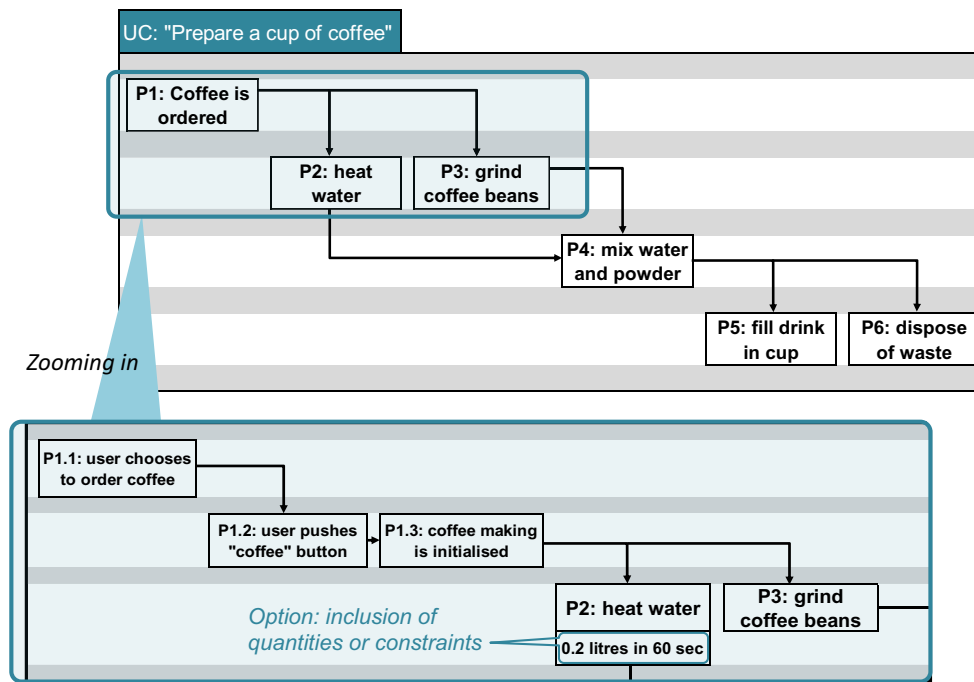
Fig. 5 Schema of a process flow view

Fig. 6 Process flow view for use case "prepare a cup of coffee"

Fig. 7 Schema of actor view

This differentiation also separates *transformation processes* (enabled by actors who are part of the system) from *interaction processes* (enabled by actors who are not part of the system, as defined in Table 2). Technical systems may contain further sub-systems that can be hierarchically discerned (as indicated in Table 2). This is indicated in Fig. 7 for "Technical sub-system 1" which is further discerned into the sub-systems "TS 1.1" to "TS 1.3".

Setting up and gradually filling the actor view is a particularly vital step as it, inherently, entails determining the particular system elements that will complementarily fulfil the required functionality (see also Sect. 3.5). Specifying the involvements (i.e. either supporting or affecting) may require a bit of good judgement by the designers, particularly in the beginning of a design project. Naturally, it is desirable to be as exhaustive in this step as possible;

however, modelling may sometimes also benefit from focussing on what is essential or some involvements may simply not be known yet. Filling the view jointly is expected to trigger a thorough discussion, thinking, and reasoning process among involved designers. It is aspired that this, inadvertently, leads to a strong engagement with the system and its elements early in the process and thus fosters building a shared understanding among involved designers regardless of disciplinary borders (see Sect. 1).

Figure 8 illustrates the allocated actors for the main process flow illustrated in Fig. 6. Actors include technical sub-systems, such as the heating system and grinder; stakeholders, such as service provider and user, as well as the environment. Their specific role in the realisation of the different processes is indicated with "**X**" for *affecting* or "**O**" for *being affected by*, respectively.

Fig. 8 Actor view for the use case "prepare a cup of coffee"

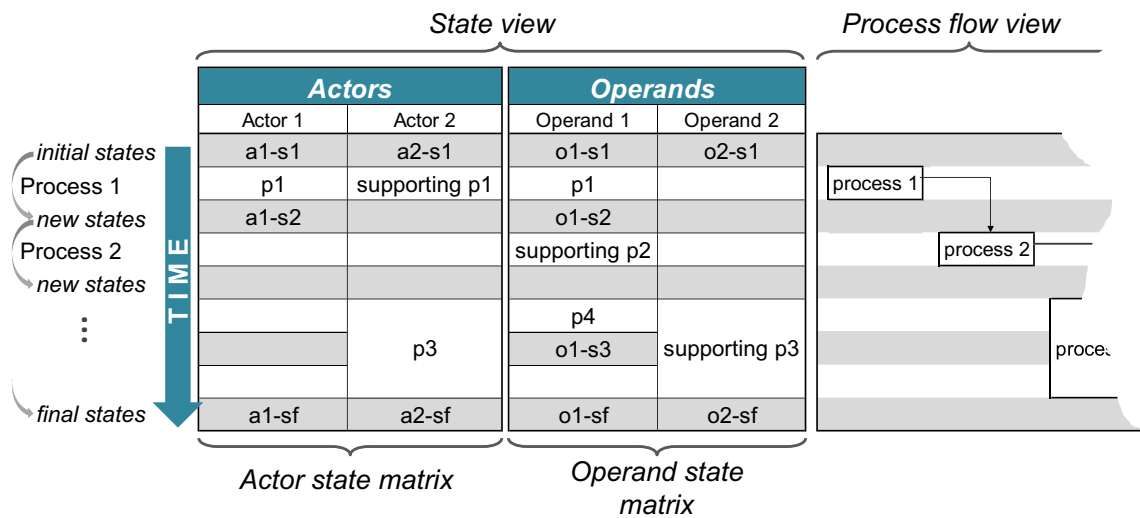| Actors	P1: coffee is ordered	P2: heat water	P3: grind coffee beans	P4: mix water and powder	P5: fill drink in cup	P6: dispose of waste
"Coffee" button	X					O
Heating system	O	X		X	X	
Grinder	O		X	X	X	X
Cup					O	
Control unit	O	X	X	X	X	
…						
Service provider						X
User	X		O			
Environment		O	O			O

System Border

Fig. 9 Schema of the state view—comprising the actor state and operand state matrices

3.2.3 State view

The *states view* (Fig. 9) represents the specific states from initial ("s1") to final ("sf") of operands ("o") and actors ("a"), as well as the state changes caused by (a) process(es) ("p"). Furthermore, it can be indicated whether operands and actors merely support a process without being changed in their states. The view consists of the *actor state matrix* and the *operand state matrix*. The adjacent placement of *state view* and *process flow views* allows for consideration of the required changes from initial to final states in parallel to the creation of the *process flow view* to facilitate their parallel development and ensure mutual consistency of represented information.

Figure 10 illustrates the *state view* for the use case "prepare a cup of coffee" including the allocated actors shown in Fig. 8. States of actors and operands are successively transformed from initial to final. For instance, the

state of water (operand) changes from 20 to 95 °C through Process 2 (P2) "heat water" and is ultimately transformed into "coffee" through Process 4 (P4) "mix water and powder".

3.2.4 Effect view

The *effect view* represents the effects, which enable individual transformation processes and are provided by actors. For each process block in the *process flow view*, a separate *effect view* may be generated using a similar representation as in the *process flow view* (see Fig. 11). *Effect views* can be modelled for one or more process blocks and allow detailed analysis of specific processes in relation to required physiochemical effects affecting them and/or contributing to their fulfilment.

Figure 11 illustrates an *effect view* for the example of Process 2 "heat water", in combination with an associated

"Coffee" button	Heating system	Grinder	Cup	Control unit	...	Service provider	User	Environ-ment		Water	Coffee beans	Energy
Actors										*Operands*		
ready for use	switched off	switched off	empty	active		inactive	desires coffee	N.A.		20°	whole	electrical
P1	P1	P1		P1			supporting P1					
locked	switched on	switched on										
	supporting P2	supporting P3		supporting P2 and P3						P2	supporting p2	P3
	switched off	switched off								95°	coffee powder	heat
	spporting P4			Supporting P4						P4	P4	
										coffee	waste powder	
P6			P5	supporting P5		supporting P6	P5				P6	
ready for use	switched off	switched off	filled	active		inactive	has coffee			coffee	emptied	heat

Fig. 10 State view for the use case "prepare a cup of coffee"

Water	Energy
20°	electrical
E1, E2	E1, E2
100°	heat

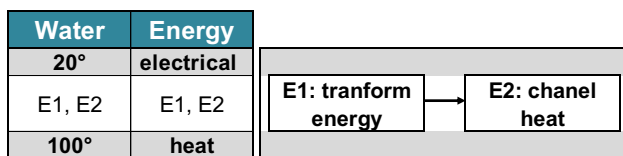| E1: tranform energy	→	E2: chanel heat

Fig. 11 Effect view with associated partial state view for Process 2 "heat water"

partial *state view*. The process requires the transformation of electrical energy into heat (through the effect E1 "transform energy"), which needs to be channelled towards the involved water (E2 "channel water"). This view effectively represents the most detailed breakdown of individual processes and, in relation to its content, corresponds closely with detailed function structures after Pahl et al. (2007), Hubka and Eder (1998; Eder and Hosnedl 2008) and similar approaches.

3.2.5 Use case view

The *use case view* lists the different use cases and indicates the involvement of individual processes within them. Dependencies between processes, which hinder their parallel or sequential execution, may impair the operability of use cases in which these are involved. The view is intended to support analysis of this kind of dependencies or similarly, e.g. dependencies between actors, operands, etc., involved in different use cases. Use cases are listed in the header column, while the flow of processes builds up the header row, which links the *use case view* with the *process flow view*.

Several of the transformation processes in the use case "prepare a cup of coffee" are also involved in multiple of the alternative use cases, as illustrated in Fig. 12. For instance, Process 2 is also involved in the use cases "prepare a cup of cappuccino", "automated cleaning", etc. However, while the use case "prepare a cup of coffee" requires 0.2 litres of water of about 95 °C, "automated

Use Cases	P1: coffee is ordered	P2: heat water	P3: grind coffee beans	P4: mix water and powder	P5: fill drink in cup	P6: dispose of waste
Prepare a cup of capuccino		x	x	x	x	x
Prepare a cup of espresso		x	x		x	x
Prepare hot tea water		x			x	
Automated cleaning		x				x

Fig. 12 Use case view for the example of the coffee vending machine

cleaning" requires different quantities of water and temperatures instead. Similarly, depending on the specific tea to be prepared, required temperatures may vary between 70 and 95 °C. All of these different use cases require a different parameterisation of Process 2 in the final design. This makes it vital for the designer to be able to discern between them. Also, occurrences of the same process in different use cases will in all probability have implications for their practicability.

3.2.6 Interaction view

The *interaction view* represents the bilateral impacts between actors and operands, as well as their complementary contributions to the realisation of a use case, associated processes, etc. As an addition, the realisation of bilateral impacts can be specified. Hence, this view is essentially an initial system structure (or interface matrix, see Fig. 13). It uses a classical DSM structure to represent the mutual relations, links, and dependencies between operands and actors.[6] Here, the specification of the

---

[6] In principle, any kinds of dependencies between actors and/or operands can be represented in this view going beyond technical or functional aspects to include, e.g. business relations and monetary flows.
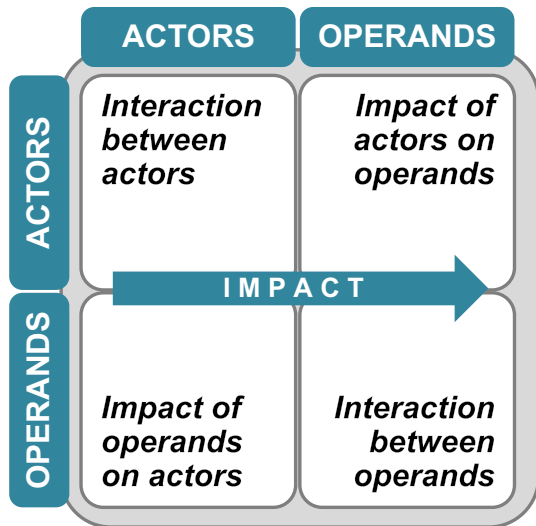
Fig. 13 Schema of interaction view

interactions between actors and operands includes the number of the respective process (to provide clarity, as numerous interactions may occur related to different processes) and a short statement specifying the interaction further. For instance, for Process 3 "grind coffee beans" in the given example, such a specification is written as "(P3)

send signal (12 V) for grinder to start", which denotes an impact between the control unit and the grinder (see Fig. 14). Conventionally, the direction of the impact is from the left to the right, which also indicates the roles of actors and operands in terms of either *impacting on* or *being impacted by* in relation to a process.

Examples of bilateral impacts to "prepare a cup of coffee" include

- the control unit (actor) impacts on the grinder (actor, see ① in Fig. 14) through signals triggering the grinder to start or stop, which are embodied through an electrical current;
- the hot water (operand) impacts on the cup (actor, see ②) and the user (actor, see ③) through transmitting heat, which is embodied through physical contact and radiation;
- operands may also impact on each other, as, e.g. energy (in form of heat) impacts on the water (see ④) during Process 4 "heat water".

A central aim behind the specific set-up of the IFM framework is to interlink the entities that are relevant for modelling the functionality of a system via the central *process flow view* in order to allow designers to take different perspectives in function modelling depending on

Fig. 14 Interaction view for use case "prepare a cup of coffee" (all "X"s in the figure serve as place holders purely for illustration purposes denoting those cells in the matrix that contain specifications

of impacts between entities as shown by zooming into the cell indicating the impact between "Control unit" and "Grinder")
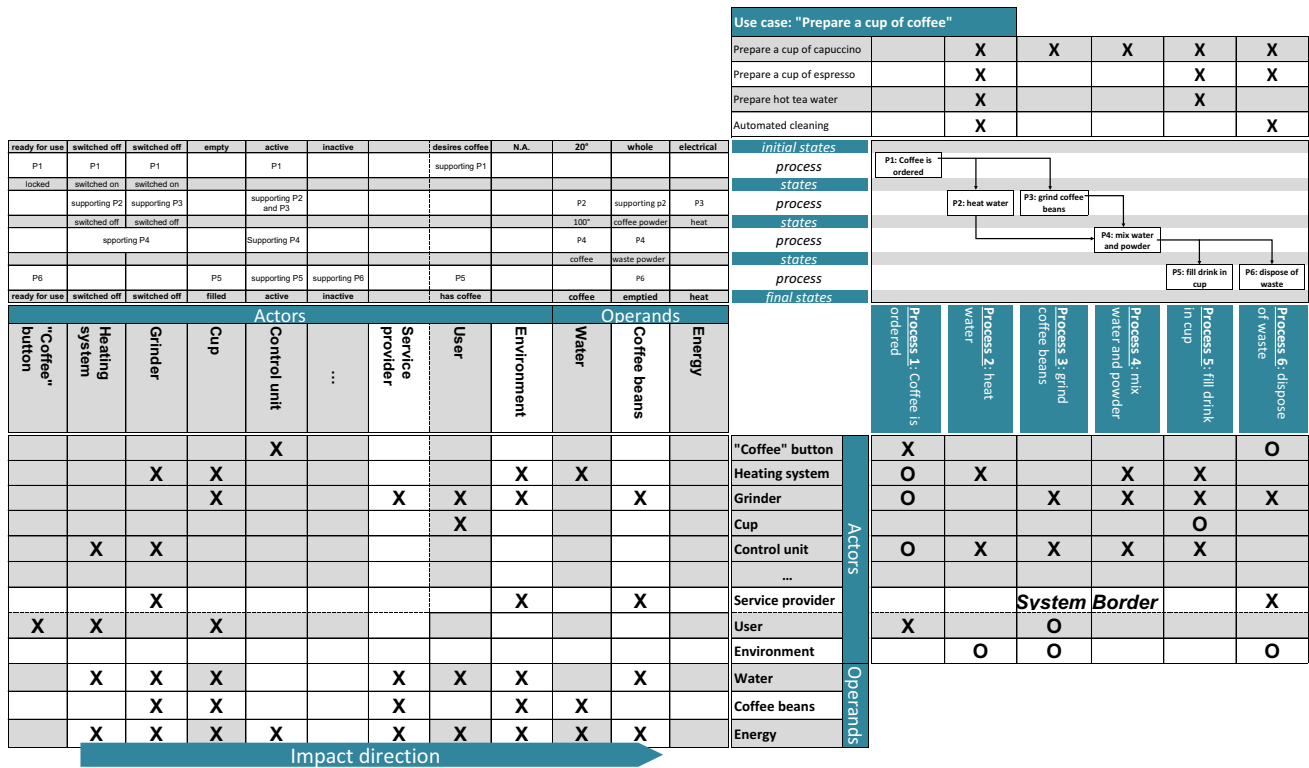
Fig. 15 Combination of associated views for the use case "prepare a cup of coffee"

what is relevant to them. At the same time, it realises a clearly structured and directly linked representation of all entities and related information. In the framework, diverse types of actors can be combined, hence, allowing modelling functional interdependencies between mechanical, electrical and software systems as well as human (or other animate) beings as well as their contribution to function fulfilment. The combination of views, therefore, integrates function modelling relating to various engineering technologies as well as services and is further expected to facilitate the cross-disciplinary exchange between designers from different disciplines. In particular, the explicit allocation of actors in the *actor view* in combination with making explicit the bilateral impacts among them in the *interaction view* is expected to endorse the designers' comprehension of the system beyond the scope of a particular discipline (Fig. 15).

3.3 Analysing functionality and function dependencies

The use of interconnected matrices in the framework should support designers in analysing system functionality by applying established analysis methods for DSM/MDM (see, e.g. Lindemann et al. 2009 and Eppinger and Browning 2012). Examples are the before-mentioned possibilities to do *consistency analysis* either for a completed model or already

during its generation. This may involve, for instance, analysing the logical consistency of the flows of processes in different use cases and the states of corresponding operands and actors and their successive changes from initial to final. This can be exemplified by rebuilding (see Fig. 16) the function structure by Pahl et al. (2007) as shown in Fig. 1. The clearly structured representation provides ease in verifying the completeness of processes, involved operands, and their sequential changes in function fulfilment. Compared to the original model, here it becomes apparent that the signal triggering and controlling the deformation of the specimen ("S" in Fig. 1, "Control signal" in Fig. 16) is not only involved in E1 "Change energy into force and movement" but logically also is involved in P1 "Load specimen" and remains in the state of "pending" after the specimen has been loaded. Temporal states of operands between processes are not included in the original. Such analysis may help designers ensuring completeness and preventing flaws in modelling and during change management.

Furthermore, the framework's set-up is expected to ease *conflict analysis* between (mutually dependent/exclusive) entities possibly preventing their involvement in the same use case (as discussed before), *change prediction* concerning elaboration on the effects of implementing changes to actors, provided functionality, use case fulfilment, etc., and vice versa, as well as *evaluating optimisation potential* such as modularisation opportunities and comparative
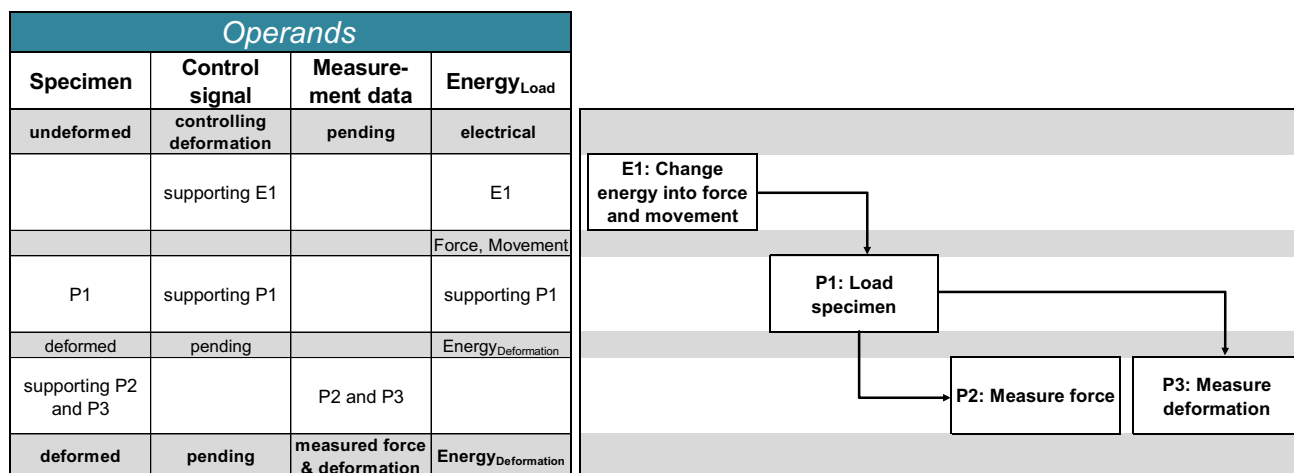
Operands			
Specimen	**Control signal**	**Measure-ment data**	**Energy$_{Load}$**
undeformed	controlling deformation	pending	electrical
	supporting E1		E1
			Force, Movement
P1	supporting P1		supporting P1
deformed	pending		Energy$_{Deformation}$
supporting P2 and P3		P2 and P3	
deformed	pending	measured force & deformation	Energy$_{Deformation}$

Fig. 16 Function structure of the tensile testing machine by Pahl et al. (2007, see Fig. 1) rebuilt using the process flow (*right-hand side*) and state view for associated operands (*left-hand side*) in the IFM framework

analysis of solution variants. The use of DSM/MDM to support function analysis is a novel application of the concept (see also Eisenbart et al. 2014).

3.4 Modularity and adaptation

As described earlier, the IFM framework has a modular set-up which allows omitting or (re-)introducing views seamlessly. This is expected to allow for demand-specific adaptation of the framework related to the preferences of (individual) designers or needs pertaining to a specific design project. Adaptation involves either augmenting, i.e. adding further information in the views (or depth in the descriptions of represented entities, respectively), or tailoring, i.e. omitting details in the different views or omitting entire views, if not required in a specific project. The latter should help in reducing modelling efforts and complexity when it is possible. Arguably, practical designers will almost always try to adapt any approach they are using to their particular needs, not only the IFM framework. However, the clear and salient distinction between contents due to their separation into views eases doing so. Depending on the specific disciplines involved and design approaches applied, the designers can flexibly select (and thus focus on) the specific views/information they require, while omitting the other views. This is exemplified in Fig. 16, wherein merely the *process view* and the *state view* are utilised. Modellers may further choose whether they would like to address the entire system at a time or focus on a specific sub-system. What is considered relevant at a specific point in time can be varied by modifying the system border (see Fig. 8). Specific adaptations of the framework to match different needs in modelling are further discussed by Eisenbart et al. (2013c). It can be imagined for future developments that, in specific design contexts, it may prove beneficial for designers to add entirely new views not included so far. While this has

not been thoroughly elaborated yet, in principle, there is no conceptual barrier for doing so if the matrix character is maintained.
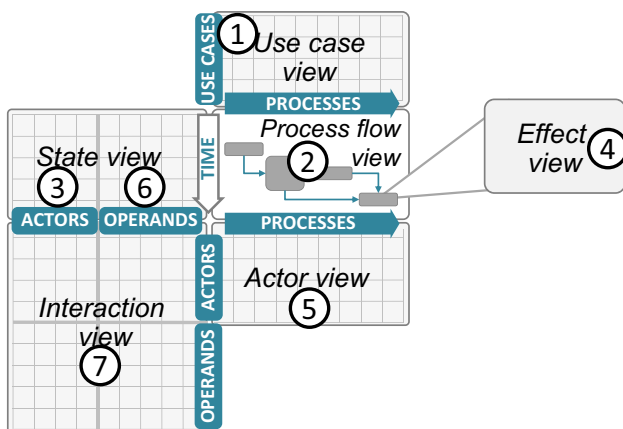
3.5 Application of the IFM framework

The framework is intended to allow for flexible application in alternative ways. Designers can start modelling using any (combination of) views and switch flexibly between them as required. In the following, *one potential way* for applying the IFM framework is described (see also Table 3). The proposed sequence of modelling activities is inspired by existing modelling approaches, which similarly differentiate inherent processes with respect to alternative use cases (see, e.g. Cockburn 2000; Kruchten 2004 or Weilkiens 2008). The assumed basis is a requirements' specification, which is initially analysed to derive central functions and sub-functions. Then, the central use cases are determined and specified based on the derived functions, which corresponds to the first step in Table 3. In this step, use cases are listed and roughly textually outlined (e.g. in terms of central goals and main involved actors, if already known).

The specific applications that a system is used in can change in the course of its life cycle. This is of particular relevance for life cycle-oriented systems such as PSS. Changes in the application of a system over its life cycle may be regarded as variant use cases and specified as such. Subsequently, Steps 2 to 7 are to be performed for each use case (see also Fig. 17). While filling individual cells, the designers may use extant function taxonomies (see Sect. 2.3) guiding the formulation of individual entries, if it is desired or demanded by the particular process applied. For describing transformation processes and operand flows established, for instance, approaches like the functional basis by Stone and Wood (2000) may lend themselves eminently.

Table 3 Potential modelling activities for applying the IFM framework

Modelling activity	Description
1 Use case definition	…includes the consolidation of the different use cases. The use cases are then listed in the respective column in the *use case view*
2 Process flow modelling	…involves determining separate flows of required processes related to each use case. Determined process flows are to be refined gradually by determining any relevant sub-processes. A multitude of alternative process flows may fulfil a use case. While modelling the process flows, the involvement of individual processes in multiple use cases needs to be considered. Modelling and selecting an alternative process flow may be facilitated through consistency analysis in relation to the required state changes in any already known operands and actors in parallel
3 Operand state modelling	…includes determining any required/involved operands and modelling their state changes in the operand state matrix (as part of the *state view*) related to the established flow of processes
4 Effect modelling	…involves establishing the required effects related to specific process blocks that are of particular interest to the designers for detailed analysis.
5 Actor allocation	…includes allocation of the actors, which are involved in the individual processes, either as affecting or as being affected by the respective processes. Allocated actors may initially be modelled as general function carriers without many details and subsequently be concretised. In early steps, such function carriers may be determined on a similar abstract level as, for instance, "organs" from approaches such as by Hubka and Eder (1988, Eder and Hosnedl 2008), Andreasen (1992) or the SAPPhIRE model (Chakrabarti et al. 2005), which are then gradually substituted with appropriate technical and non-technical actors collaboratively implementing the desired functionality. This step can be supported by established methods such as morphologic charts or creativity methods (see, e.g. Pahl et al. 2007; Ulrich and Eppinger 2008). The selected combination of actors defines the specific technologies to be integrated, i.e. whether the developed system combines mechanical with electrical components or further involves software systems as well as eventual services and their providers. This step essentially marks the transition to the potential solution concept
6 Actor state modelling	…includes modelling the state changes in allocated actors in the actor state matrix (as part of the *state view*) related to the chosen process flows
7 Interaction specification	…involves analysing and detailing the specific bilateral impacts among actors, among operands, and between actors and operands

Fig. 17 Potential sequence of modelling activities and associated views

4 Evaluation

4.1 Study design

The empirical study presented in the following is an initial evaluation of the proposed IFM framework. It is intended to obtain feedback from practitioners regarding the framework's potential usefulness, practical applicability as well as to identify potential for further improvement. Social sciences provide a wide selection of methods that can be used to analyse human perceptions, behaviours and products resulting from human behaviour (Diekmann 2001; Bender et al. 2002). Different variations in these methods are—and have been—widely applied in design research (see, e.g. Jänsch 2007; Blessing and Chakrabarti 2009). The aim of the presented study was to explore the specific opinions, needs and preferences of practitioners, who are active in technical system, service and/or PSS development, in relation to the proposed framework. This suggests the use of surveys (Yin 2009). These typically involve questionnaires and/or conversations, such as interviews, with subjects that are able to provide insight into the phenomenon the researcher is interested in (see Patton 2002; Blessing and Chakrabarti 2009). The study is guided by the following research questions:

1. Which specific contents and views in the IFM framework are considered useful, respectively, which are considered as less useful for function modelling?
2. What are potentials for further improvement?

4.1.1 Method

The presented evaluation study focuses on receiving feedback from practicing designers in different companies. Following suggestions by Cicourel and Haug (1974) and

Yin (2009), it was decided to use presentation workshops in which the framework and its central characteristics are presented to practitioners and ask for their feedback. Overall, the study comprised four phases: preparation including pilot studies, recruitment of participants, as well as data collection and analysis.

The preparation phase included the generation of the questionnaire and of the presentation to be used. This was supported by continuous feedback from two experienced researchers in the field of engineering design research. Furthermore, pilot studies were conducted with two practitioners in industry who both also have backgrounds in engineering design research. Received feedback was used to adapt the formulation of the questionnaire slightly, in order to improve its intelligibility.

Data were collected using the developed questionnaire as well as open discussion. The questionnaire comprises five main questions investigating the perceived usefulness of individual contents and views in the framework, its general applicability, and any desired changes. Questionnaires were answered anonymously. Verbal feedback provided by the participants during open discussions was collected through audio recording; in addition, notes were taken in order to put down particularly relevant statements.

Six workshops were conducted typically at the site of the involved companies. One workshop was conducted via an online conference tool. The workshops lasted on average about 90 min, with a minimum of 67 min and a maximum of 111 min. The workshops started by introducing the participants to the general aims and concepts of the IFM framework including central entities and their relations as well as offered possibilities for modelling and analysing functionality. The same presentation was used for all workshops. After introducing the general concept of the IFM framework, each view, its contents, and their links to other views were successively presented. The presentation part of the workshops took

between 30 and 40 min. The questionnaire is organised according to the structure of the presentation, and participants were asked to evaluate the usefulness of views and contents in parallel. Additional feedback could be given using provided comment boxes. Participants were further asked for any information they would have considered useful, but is not included in the framework thus far. Finally, participants were asked for feedback concerning the applicability and usefulness of the framework in general, again both in the questionnaire and through open discussions.

4.1.2 Participants profile

A total of 19 designers from six companies participated in the study. An overview of the participating companies, such as main market area and number of employed designers, is provided in Table 4 along with the particular disciplines that participants were involved in at the time of the study. In Companies E and F, two experts on function modelling could be recruited, who had implemented function modelling themselves in their companies prior to the study.

Company sizes vary between a small-sized company with below 30 employees and an annual turnover of about 3.4 million € (in 2012) to a company with more than 275.000 employees and an annual turnover of more than 100 billion € (in 2012). Participants comprise specialist engineers ($n = 8$) from mechanical engineering ($n = 2$), electrical engineering ($n = 5$), software development ($n = 1$) and service design ($n = 2$). Furthermore, system-level designers ($n = 9$, including systems engineers and project leaders) with backgrounds in mechanical engineering ($n = 6$), aerospace technology ($n = 2$) and electrical engineering ($n = 1$) participated. The majority of participants ($n = 12$) possessed professional experience of ten or more years, with a minimum of 4 years and a maximum of 23 years.

Table 4 Overview of companies and participants involved in the study

Companies	Main market area	Number of designers in company	Distribution of different design teams	Number of participants	Participants' disciplines				
					Mechanical engineering	Electrical engineering	Software development	Service design	System-level design
A	Aerospace	<50	One site	5	1	2			2
B	Aerospace	<50	One site	4		1		1	2
C	Automotive	50–250	Global	3		2			1
D	Manufacturing machinery	50–250	Global	5	1		1	1	2
E	Automotive	>250	Global	1					1
F	Hydraulics and energy systems	>250	Global	1					1

4.2 Results

Seventeen of the 19 participants in the workshops filled out the questionnaire in addition to providing verbal feedback; two participants chose to provide their feedback verbally only.

4.2.1 Contents and views considered useful in the IFM framework

Research Question 1 is addressed using the answers provided to Questions 1 and 3 in the questionnaire, as well as feedback obtained during discussions in the workshops. Therein,

- Question 1 asks participants to indicate which of the represented "elements and aspects [i.e. the contents in each view] are considered useful" for their work;
- Question 3 asks participants to indicate which "particular views […] are considered useful or not useful".

4.2.1.1 Assessments of views Of the total of 17 returned questionnaires, 14 provided comprehensive assessments of the views in the framework. In 13 of these 14 questionnaires, at least half of the six views in the IFM framework were marked useful, in two of the questionnaires even all six views. The specific combinations of which views are considered useful or not useful vary considerably between the questionnaires and only two combinations occurred more than once (twice and three times, respectively). Figure 18 illustrates the distribution of the provided assessments for each view.

One of the three participants who did not provide an assessment of the views wrote a comment that—based on his experience—not all the contents and thus not all views are relevant in each single design project. The person had

marked 22 of the 23 contents to be useful, which suggests that he nevertheless considered all views to be useful in principle, but apparently not in all situations alike. This in fact supports the demand for adaptability of the framework to the designers' needs, which was derived as an essential requirement earlier. In the two remaining questionnaires, no explanations were provided and could neither be found in the audio recordings from the respective workshops. However, in one of the two questionnaires, almost exclusively contents in the *actor view*, *use case view* and *effect view* were marked to be useful; all other contents (except for the "technical processes" in the *process flow view*) were marked not useful or "don't know", which suggests a certain preference for these views over the others.

4.2.1.2 Assessments of contents The participants were asked in the questionnaire to assess a total of 23 distinct contents represented in the IFM framework (see Fig. 19). Assessments provided are considerably positive: 16 participants considered 13 or more of the 23 contents to be useful; 12 participants considered 17 or more contents as useful. Every content was considered useful by at least eight participants. The specific combinations of which contents are considered useful or not useful vary strongly between participants, and no combination occurred more than once. Technical processes, quantities and/or constraints, concerned technical sub-systems, actor states and processes related to state changes were assessed as useful by at least 15 out of 17 participants who filled out the questionnaire. The minimum amount of contents considered useful is nine (found in one questionnaire); the maximum amount is 22 (found in two questionnaires). Eight participants marked specific contents to be not useful, with a minimum of one (in two questionnaires) and a maximum of five (in three questionnaires). Contents which are
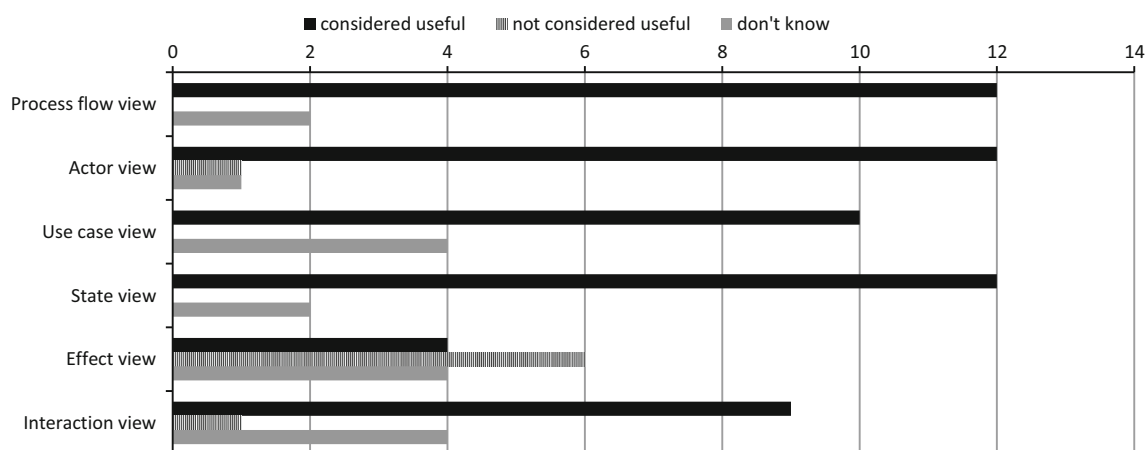

Fig. 18 Assessments of views encompassed in the IFM framework (*n* = 14)
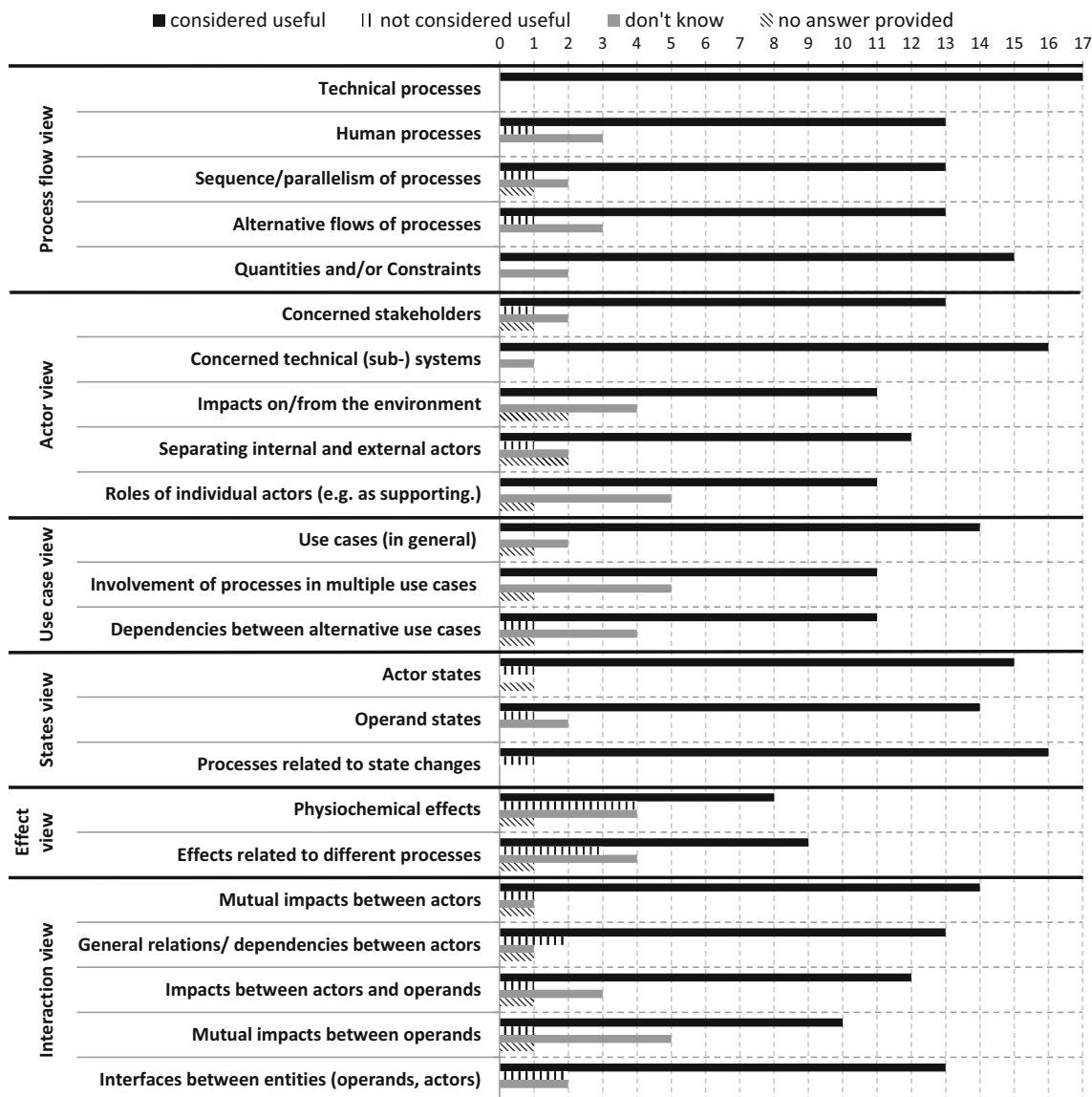
Fig. 19 Assessment of contents addressed in the IFM framework ($n = 17$)

considered most often as "not useful" are physiochemical effects and effects related to different processes (see Fig. 19). Five participants did not provide any assessment for some of the contents.

4.2.1.3 Discrepancies between assessments of contents and views Contents addressed in the *process flow view*, the *state view* and *actor view* are considered useful most often by the participants (see Fig. 19). In contrast, the *effect view* and its contents are considered useful least often. This corresponds to the assessments of the respective views (Fig. 18). Still, there are slight discrepancies between the amounts of participants who assessed specific contents of views useful in contrast to the number of participants who assessed the associated views themselves useful. This is most notable for the *effect view*:

while up to nine participants regarded the contents addressed in the *effect view* useful (see Fig. 19), only four considered the view as such useful. Only one of the five deviating participants provided an explanation in the questionnaire. The person stated that the contents in the *effect view* were considered to be very useful; however, he would have preferred them to be integrated into the *process flow view*, which is why he did not mark the *effect view* useful in Question 3.

4.2.1.4 Influences on the provided assessments The data collected in the questionnaires do not show particularly apparent groups which would suggest a discipline-specific accumulation of the provided assessments. However, the verbal feedback received hints towards a certain influence on the assessments coming from

- the *discipline* a participant is associated with and
- the overall *conceptual design approach applied* in a company.

An *influence from the participant's discipline* was suggested in the open discussions related to the *effect view*. While this view is considered useful the least number of times (see Fig. 18), two participants in two workshops (Companies D and E) verbally expressed that they perceive this view as one of the most beneficial in the entire framework. Both participants have a background in mechanical engineering, and the contents in the *effect view* are mainly prominent in function modelling proposed in the mechanical engineering literature as was briefly discussed earlier. This suggests at least a certain degree of dependency of the assessments to a participant's disciplinary background.

A considerably larger influence, however, is suggested for the *conceptual design approach applied* in a company. In two workshops (Companies A and C), this was particularly evident. Three out of the five participants from Company A repeatedly said during discussions that the *interaction view* was especially important to them. They further claimed they would want to use it as starting point for modelling with the framework. At the time of the study, the company used a matrix-based model, quite similar to the *interaction view*, for representing the basic structure of a system under development. This particular model was typically used as starting point in a development project for verifying to what extent parts of an already existing system might be reused. In Company C, use case modelling was typically applied, and two of the three participants correspondingly expressed that they would want to start modelling in the framework using the *use case view*.

4.2.2 Benefits and potentials for further improvement

Research Question 2 is addressed using the answers provided to Questions 2, 4 and 5 in the questionnaires. Therein,

- **Question 2** asks participants to indicate any additional "information they would have liked or considered useful" in the framework;
- **Question 4** asks participants whether they would "consider using the framework in future design work" and provide an explanation for their selection;
- **Question 5** asks participants whether they would "generally prefer an alternative set-up or representation for the framework" and whether they had "any other comments".

The comments provided to Questions 4 and 5 in the returned questionnaires are to a large extent overlapping so

that both questions were jointly analysed. The quantities provided in the following only serve as indicators rather than absolute values. Since questionnaires were answered anonymously, it cannot be controlled whether participants' verbal feedback also appears in their questionnaires, in which case it may be counted twice. To avoid this, whenever verbal feedback was also similarly found in questionnaires from the same workshop, either the number of questionnaires or the number of participants verbally raising a specific issue was counted, not both. This was the case in two workshops and could potentially make a difference in two persons regarding the issue of complexity (i.e. $n = 6$ instead of $n = 5$) and the benefits expressed in relation to doing function analysis with the framework (i.e. $n = 8$ instead of $n = 7$).

4.2.2.1 Expected application of the IFM framework in future design work   In all 17 returned questionnaires, the first part of Question 4 was answered positively (see Fig. 20). The large majority ($n = 15$) either stated to be willing to use parts of the IFM framework (e.g. a selection of views or specific contents only) in the future or to be willing to apply it, provided that certain adaptations are implemented.

The two participants who selected "yes" both come from Company D. They also assessed a high number of contents and views (one even all views) in Questions 1 and 3 to be useful. Overall, these two participants and a third participant from this company (who selected "parts of it" in Question 4) provided the most favourable feedback in the questionnaires in the entire study. However, the fourth participant from this company was the one to consider the least number of contents to be useful in the entire study ($n = 9$ contents, no answer provided to Question 3). He selected "parts of it" in Question 4, but no explanations were provided.

Of the five participants who stated that they would consider using the framework "with adaptations", three provided specific suggestions on how to adapt the framework in the provided comment boxes. The other two participants did not make suggestions but expressed concerns that the matrices in the framework may quickly become rather large and complex. A similar concern was also
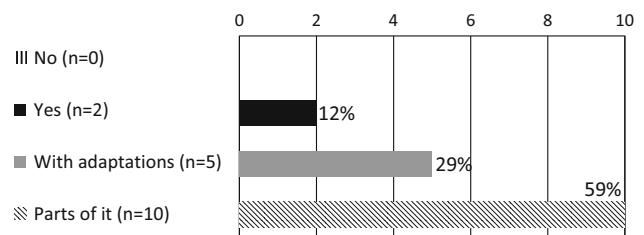
Fig. 20 Provided answers for whether participants would consider using the framework in future design work ($n = 17$)

expressed by one of the ten participants who selected "parts of it". Only three participants who expressed considering "parts of it" provided concrete explanations for their selection. All provided comments are discussed in the following.

4.2.2.2 Expressed benefits Over half of the participants ($n = 10$) provided explicitly positive comments. One of the participants highlighted that the combination of gradually filling the views in the framework and the ease of analysing entered contents in parallel—to him—could be a substantial contribution to building a comprehensive and shared understanding in the design team. The reason given was that it could promote iterative modelling and analysis of entered information, thus inherently intensifying the engagement of all designers with it. Participants from one company verbally expressed interest in applying the framework soon in one of their upcoming projects. Aspects that the participants expressed to be particularly beneficial comprise:

- ease of performing function analysis ($n = 7$), particularly referring to

  - modelling and analysing dependencies between actors ($n = 3$),
  - analysing the impacts from and to the environment ($n = 2$),
  - analysing the time dependencies between functions ($n = 2$),
  - consistency and completeness analysis while gradually detailing the function model of a system ($n = 2$) or in order to facilitate change management ($n = 1$);

- representing the aspects of system functionality in relation to a time flow ($n = 2$);
- making explicit the links between components that are developed in different departments, which may support exchange in the project management team as well as planning of discipline-specific and collaborative design tasks ($n = 2$);
- clarity of the representation of contents and their relations in the framework ($n = 1$);
- the matrix-based representation as being open to represent any dependencies between entities in a system making it potentially also applicable for business modelling ($n = 1$).

4.2.2.3 Desired additional contents Four participants expressed a desire for additional contents; these include:

- illustrating the chronological sequence of use cases ($n = 2$);

- making explicit the relation between the requirements and modelled functions ($n = 1$);
- making explicit the conditions for function execution ($n = 1$).

Participants raising the first issue expected added benefit in regard to making explicit what is required from the system under development at a particular phase of its life cycle. The second issue may facilitate traceability from requirements to functions, which was seen as a benefit from a process management point of view. The third issue refers to making explicit that some functions cannot (or should not) be executed unless specific preconditions are fulfilled, e.g. for safety reasons.

4.2.2.4 Desired adaptations Only a relatively small number of six out of 19 participants expressed concerns or made suggestions for specific applications. These encompass:

- the framework may become complex, if modelled for an entire system ($n = 5$),
- evaluation of variants in the flow of processes should be facilitated more strongly ($n = 2$);
- additional guidance on how to make decisions on the design of the system while modelling with the IFM framework may support conceptual design ($n = 1$);
- matrices may not be the most desirable representation for all designers ($n = 1$).

The first concern was expressed in relation to developing additional support for managing complexity, e.g. by providing guidance on how to make "trade-offs" between modelled contents and the required modelling efforts. The latter goes hand in hand with the third issue raised. The second issue was raised during a discussion, suggesting that by facilitating early comparison of alternative process flows, designers might be able to eliminate less suitable solutions quickly.

4.3 Limitations

Limitations of this study in relation to the validity of the findings are essentially twofold: the intelligibility of the questionnaire and presentations, as well as the limited number of participants. The former was tested and improved using feedback obtained from other researchers and during the pilot studies. In addition, participants were encouraged repeatedly during the workshops to ask for the clarification of questions and the presentation whenever required. It is a general impression of the mainly involved researcher that the participants were able to understand the presented concepts quickly. They frequently started asking rather specific questions about concrete aspects of the

framework early during the workshops, which suggests that the presentation has been suitable to communicate the characteristics of the IFM framework. A general limitation of feedback-based surveys in evaluation studies is that—for various reasons—participants sometimes may feel reserved to express criticism (see Blessing and Chakrabarti 2009) or may have been influenced by an acquiescence bias (see Watson 1992). In the study, it was attempted to minimise both these effects by giving the participants the possibility to remain anonymous in the questionnaires.

More generally, it needs to be highlighted that the study is limited to presentation workshops rather than applying the framework within concrete design projects, which may hampers validating its practical applicability thoroughly. Also, the relatively small sample size of 19 participants from six companies prevents generalisation of the obtained insights. Future research will need to corroborate the obtained findings through applying the framework in practice. However, all participants in the conducted study were experienced designers or even experts in function modelling who work in interdisciplinary product development projects on an everyday basis. As mentioned before, during the workshops, the participants frequently started discussing among themselves how they would use certain parts of the framework for modelling particular issues they saw in their own products. The received feedback is considerably rich and highly valuable as it led to the identification of concrete vantage points for further improvement. Central aspects of the received feedback surfaced in different companies alike, which gives further confidence in the presented insights.

4.4 Summary and discussion of the evaluation study

4.4.1 Modularity and possibilities for function analysis are particularly beneficial

The large majority of participants assessed contents and views in the framework to be useful, and no participant refused the idea of using the framework in future work. None of the participants considered the central *process flow view* to be not useful; in fact, the view and its contents are considered useful most often in the returned questionnaires. This can be regarded as particularly positive, insofar as the *process flow view* is constitutive to the framework and its current set-up. Similarly, the found diversity in relation to which specific combination of views and contents are considered useful substantiates the need for adjustability in function modelling discussed earlier. Different combinations of views and individual contents will be relevant depending on the specific used design approach, the project at hand and the designers involved. The possibilities for augmenting and tailoring the framework are explicitly

foreseen to support this diversity in the application; they are therefore considered one of the framework's main benefits.

4.4.2 Potentials for further improvement

The requested inclusion of conditions for function execution is relatively seamlessly realisable. Making explicit the links between requirements and functions in the framework, however, will require additional research. Adequate visualisation of these links is by no means trivial as it is dependent on how the requirements are documented to begin with (which is often specific for individual companies) and which of the entities in the framework they relate to. Similarly, modelling the chronological order of use cases needs further consideration, given that it will be difficult to apprehend the concrete sequence of different use cases over a system's life cycle accurately after a certain level of detail.

The most critical issues in the received feedback concern modelling complexity, the requested additional guidance regarding adaptation of the framework to the requirements of individual designers in a specific situation as well as taking design decisions while gradually moving towards a solution concept. As indicated earlier, the framework addresses the largest amount of function modelling perspectives compared to other reviewed function models, which is why the inherent matrices may become large rather quickly. Similar DSM-based modelling approaches like quality function deployment (QFD, see King 1989), which may yield rather large matrices as well, are widely applied in engineering practice, nonetheless. There is no particular reason, why this should not similarly apply to the IFM framework. Still, in the light of the received feedback, future research should focus on developing suitable guidance or supporting measures, such as checklists, training material or similar, to support the designers in tailoring the framework to their specific demands. It will be essential for this research to determine which specific contents have to be included to still gain benefit from the framework and limit complexity and modelling efforts at the same time.

5 General discussion and conclusion

The overall goal of the research presented in this article has been the development of a function modelling approach, which is capable of facilitating collaborative design in the development of technical systems as well as services and PSS. Within disciplines typically involved in related development projects, but also cross-disciplinary such as mechatronics, systems engineering or PSS design, a

multitude of function models can be found, both in the literature and practice. These models are specific with respect to the addressed contents (i.e. function modelling perspectives), the used modelling morphologies and the proposed application as well as the particular notions of function they are based on. These discrepancies hamper a consistent exchange of information between different function models used.

In this article, specific properties were derived that are deemed vital for an integrated function modelling approach aiming to bridge the found diversity and link relevant perspectives in modelling and reasoning about functionality. The paper proposes the integrated function modelling (IFM) framework as an attempt to provide such an adequate modelling approach. The framework interlinks salient information related to the identified function modelling perspectives and morphologies in an adaptable, clearly structured manner that should foster diverse application. The framework is comprehensive beyond extant function modelling approaches in terms of integrating the identified function modelling perspectives and morphologies prominent within and across such disciplines typically involved in the development of multi-technology systems and PSS. This is coupled with initial architectural modelling which may aid designers in the creative leap from functional considerations to an initial design solution (or vice versa). The IFM framework progresses function modelling compared to more established approaches, such as the SAP-PhIRE model or ontologies by Gero (1990), Kitamura and Mizogushi (2007), insofar as it facilitates abstract descriptions of services and products alike, as well as combined offerings such as PSS. At the same time, it provides a greater flexibility in modelling compared to rather formalised modelling languages, e.g. SysML or OPD. The IFM framework is not specifically intended to oppose extant research related to formalising function modelling to tackle difficulties stemming from semantic variances discussed by, e.g. Erden et al. (2008), Kitamura and Mizogushi (2007) and other scholars (see Sect. 2.3). It is envisaged to foster shared comprehension by linking different viewpoints onto functionality pertaining to complementary notions of function and related primitives or types of function, respectively (discussed by Chandrasekaran 2005; Aurisicchio et al. 2012). Different types of information relating to these notions and viewpoints are visualised distinct from but at the same clearly coupled with each other, which is inherently aspired to provide clarity in modelling to enhance a shared understanding among designers using the model. In addition, the framework does not hinder additional utilisation of existing function taxonomies to enhance formal clarity even further, if desired. From a methodological point of view, the authors—similar to Erden et al. (2008) and van Eck

(2010b)—believe that specific design contexts and tasks require specific modelling. The framework's genuine expected benefits originate from the relatively seamless adaptability pertaining to specific demands and rationales[7]. Furthermore, particular benefit is expected from the unequivocal visual and contextual interrelation of modelled information to ease modelling and analysing functionality. These expected benefits are substantiated in the conducted evaluation study.

Participants in the described workshops have diverse educational backgrounds and come from companies active in considerably different industrial branches. The disutility voiced by several participants towards some views and contents, in contrast to a number of others who considering these to be highly beneficial, is in fact to be expected. It confirms the need for demand-specific application of function modelling.

The offered comprehensiveness in relation to function modelling content goes beyond other reviewed modelling approaches. Still, extant models retain being of avail and may be favoured by individual designers. As the IFM framework builds on a process flow, it lends itself particularly to modelling non-static systems, though others can be represented as well (see Eisenbart et al. 2013a, b, c, d). Nevertheless, it is aspired by the authors that the profound possibilities provided by the IFM framework will support joint use by designers from different disciplines and for them to take interest in using the framework to look beyond the contents they typically focus on.

The presented research extends the scope of investigation compared to other scholars covering multiple disciplines in relation to function modelling, such as Buur (1990) or Erden et al. (2008). Yet, it remains focused on technical product and service development. Whether it may further be applicable to disciplines such as architecture or industrial engineering has not been addressed thus far. However, it is aspired to become a starting point for further investigation of function modelling across disciplines. Feedback from the participants in the evaluation study suggests that future research ought to address the inclusion of certain additional contents as well as to determine appropriate supporting measures for practitioners to adjust the framework to suit their specific needs and that of the project. From a critical point of view, the conducted

---

[7] In related research, it has been successfully applied as a substitute visualisation in a wide selection of modelling approaches (see Eisenbart et al. 2013a, b, c, d; Eisenbart 2014) including those proposed by Tjalve 1978, Hubka and Eder 1988; Buur 1990; USDoD 2001; Pahl et al. 2007, exemplified in Fig. 16; Sakao and Shimomura 2007; and Watanabe et al. 2011, see Fig. 2. In all cases, the substitution was successful requiring only minor adaptations, if at all. Furthermore, a thorough comparison with central diagrams and concepts entailed in UML/SysML was conducted (see Eisenbart et al. 2015a).

empirical study remains an initial evaluation, and it will be vital to advance into concrete practical applications. Areas in industry that will be of specific interest for further evaluation are, for instance, interdisciplinary design projects in the automotive and aerospace sector. They typically employ a variety of disciplines in globally distributed departments, thus creating a particular need for clarity in model-based communication. This is expected to strain the framework's capabilities, thus giving valuable insight. Future work will include the development of a software tool to assist the application of the framework. A prototypical software implementation has already been attained in collaboration with an academic partner (see Dohr et al. 2014).

References

Ahmed S, Wallace K (2003) Evaluating a functional basis. In: Proceedings of the ASME design engineering technical conferences and computers and information in engineering conference IDEC/CIE

Albers A, Sadowski E, Braun A (2010) Funktionsorientierte Produktentwicklung in frühen Phasen von Entwicklungsprozessen. 8. Gemeinsames Kolloquium Konstruktionstechnik

Alink T (2010) Bedeutung, Darstellung und Formulierung von Funktionen für das Lösen von Gestaltungsproblemen mit dem C&C-Ansatz. Dissertation, Institut für Produktentwicklung, Karlsruhe Institute of Technology, Karlsruhe

Alink T, Eckert C, Ruckpaul A, Albers A (2010) Different function breakdowns for one existing product. Experimental results. In:

Gero J (ed) Design computing and cognition—DCC. Springer, Dordrecht, pp 405–425

Andreasen MM (1992) The theory of domains. In: Ullman D, Blessing LTM, Wallace K (eds) Understanding function and function-to-from evolution: workshop report, CUED/C-EDC/TR 12. Engineering Design Centre, Cambridge, pp 21–47

Andreasen MM, Hein L (2000) Integrated product development. The Institute for Product Development, IPU, Copenhagen

Aurisicchio M, Eng N, Ortiz NJ, Childs P, Bracewell R (2011) On the functions of products. In: Proceedings of the 18th international conference on engineering design—ICED

Aurisicchio M, Bracewell R, Armstrong G (2012) The function analysis diagram. In: Proceedings of the ASME design engineering technical conferences and computers and information in engineering conference IDEC/CIE

Bender B, Reinicke T, Wünsche T, Blessing LTM (2002) Application of methods from social sciences in design research. In: Proceedings of the 11th international design conference—DESIGN

Blessing LTM (1997) Applying systematic design: the flight refuelling probe project, CUED/C-EDC/TR 48. EdC Cambridge Engineering Design Centre, Cambridge

Bonaccorsi A, Apreda R, Fantoni G (2009) A theory of the constituent elements of functions. In: Proceedings of the 17th international conference on engineering design—ICED 2009

Bracewell R, Sharpe JEE (1996) Functional descriptions used in computer support for qualitative scheme generation—Schemebuilder. Artif Intell Eng Des Anal Manuf AI EDAM 10(4):333–346

Braha D, Reich Y (2003) Topological structures for modeling engineering design processes. Res Eng Des 14(4):185–199

Brown DC, Blessing LTM (2005) The relationship between function and affordance. In: Proceedings of the ASME international design engineering technical conferences & computers and information engineering conference IDEC/CIE

Buur J (1990) A theoretical approach to mechatronics design. Dissertation. Technical University of Denmark, Copenhagen

Carrara M, Garbacz P, Vermaas P (2011) If engineering function is a family resemblance concept: assessing three formalization strategies. Appl Ontol 6(2):141–163

Chakrabarti A, Bligh TP (2001) A scheme for functional reasoning in conceptual design. Des Stud 22(6):493–517

Chakrabarti A, Sarkar P, Leelavathamma B, Nataruja B (2005) A functional representation supporting process and product knowledge in biomimetic design. Artif Intell Eng Des Anal Manuf AI EDAM 19(2):113–132

Chakravarthy B, Albers A, Schweinerger D (2001) Collaborative environment for concept generation in new products. In: Proceedings of international council of societies of industrial design—ICSID

Chandrasekaran B (2005) Representing function. Relating functional representation and functional modelling research streams. Artif Intell Eng Des Anal Manuf AI EDAM 19(2):65–74

Chandrasekaran B, Josephson J (2000) Function in device representation. Eng Comput 16(3–4):162–177

Chiang WC, Pennathur A, Mital A (2001) Designing and manufacturing consumer products for functionality: a literature review of current function definitions and design support tools. Integ Manuf Syst 12(6):430–448

Chittaro L, Kumar AN (1998) Reasoning about function and its applications to engineering. Artif Intell Eng Des Anal Manuf AI EDAM 12(4):331–336

Cicourel A, Haug F (1974) Methode und Messung in der Soziologie. Suhrkamp Verlag, Berlin

Cockburn A (2000) Writing effective use cases. Addison Wesley Professional, Indianapolis

Crilly N (2010) The role that artefacts play: technical. Soc Aesthetical Funct Stud 31(4):311–344

Crilly N (2015) The proliferation of function: multiple systems playing multiple roles in multiple supersystems. Artif Intell Eng Des Anal Manuf AI EDAM 29(1):83–92

Cross N (2008) Engineering design methods: strategies for product design. Wiley, Chichester

Deng Y (2002) Function and behaviour representation in conceptual mechanical engineering. Artif Intell Eng Des Anal Manuf AI EDAM 16(5):343–362

Dewey A (2000) Digital and analogue electronic design automation. In: Dorf RC (ed) The electrical engineering handbook. CRC Press, Boca Raton

Diekmann A (2001) Empirische Sozialforschung: Grundlagen, Methoden, Anwendungen. Rowohlt-Taschenbuch-Verlag, Reinbek bei Hamburg, Rowohlts Enzyklopädie

Dohr F, Eisenbart B, Huwig C, Blessing LTM, Vielhaber M (2014) Software support for the consistent transition from requirements to functional modeling to system simulation. In: sings of the 10th NordDesign conference

Dori D (1995) Object-process analysis: maintaining the balance between system structure and behavior. J Log Comput 5(2):227–249

Eckert C (2013) That which is not form: the practical challenges in using functional concepts in design. Artif Intell Eng Des Anal Manuf AI EDAM 27(3):217–231

Eder W, Hosnedl S (2008) Design engineering: a manual for enhanced creativity. CRC Press, Boca Raton

Eisenbart B (2014) Supporting interdisciplinary system development through integrated function modelling. Dissertation, University of Luxembourg, Luxembourg

Eisenbart B, Blessing LTM, Gericke K (2012) Functional modelling perspectives across disciplines: a literature review. In: Proceedings of 12th international design conference—DESIGN

Eisenbart B, Gericke K, Blessing LTM (2013a) An analysis of functional model approaches across disciplines. Artif Intell Eng Des Anal Manuf AI EDAM 27(3):281–289

Eisenbart B, Qureshi AJ, Gericke K, Blessing LTM (2013b) Integrating different functional modelling perspectives. In: Chakrabarti A, Prakash R (eds) Global product development, ICoRD'13., Lecture notes in mechanical engineeringSpringer, London, pp 85–97

Eisenbart B, Gericke K, Blessing LTM (2013c) Adapting the IFM Framework to functional approaches across disciplines. In: Proceedings of the 19th international conference on engineering design—ICED

Eisenbart B, Dohr F, Gericke K, Vielhaber M, Blessing LTM (2013d) Potentials for realising a consistent transition between function modelling with the IFM framework and early system simulation. In: Proceedings of the 19th international conference on engineering design—ICED

Eisenbart B, Gericke K, Blessing LTM (2014) Application of the IFM framework for modelling and analysing system functionality. In: Proceedings of the 13th international design conference—DESIGN

Eisenbart B, Mandel C, Gericke K, Blessing LTM (2015a) Integrated function modelling: comparing the IFM framework with SysML. In: Proceedings of the 20th international conference on engineering design—ICED

Eisenbart B, Gericke K, Blessing LTM (2015b) DSM for modeling and analyzing functionality: views of practitioners. In: Proceedings of the 17th international DSM conference

Eppinger SD, Browning TR (2012) Design structure matrix methods and applications. MIT Press, Cambridge

Erden M, Komoto H, van Beek TJ, D'Amelio V, Echavarria E, Tomiyama T (2008) A review of function modeling: approaches and applications. Artif Intell Eng Des Anal Manuf AI EDAM 22(2):147–169

Fantoni G, Apreda R, Bonaccorsi A (2009) Functional vector space. In: Proceedings of the 17th international conference on engineering design—ICED

Far BH, Elamy H (2005) Functional reasoning theories: problems and perspectives. Artif Intell Eng Des Anal Manuf AI EDAM 19(2):75–88

Fowler M (1998) Analysis patterns: reusable object models. Addison Wesley Longman Inc., Boston

Garbacz P, Borgo S, Carrara M, Vermaas PE (2011) Two ontology-driven formalisations of function and their comparison. J Eng Des 22(11–12):733–764

Gero JS (1990) Design prototypes: a knowledge representation schema for design. AI Mag 11(4):26–36

Goel A (2013) One thirty year long case study: fifteen principles implications of the AI methodology for functional modelling. Artif Intell Eng Des Anal Manuf AI EDAM 27(3):203–215

Hirtz J, Stone RB, Szykman S, McAdams D, Wood KL (2001) Evolving a functional basis for engineering design. In: Proceedings of the ASME design engineering technical conference—DETC2001

Houkes W, Vermaas PE (2010) Technical functions: on the use and design of artefacts. Philosophy of engineering and technology. Springer, Dordrecht

Hubka V (1980) Principles of engineering design. Butterworth-Heinemann, Oxford

Hubka V, Eder W (1988) Theory of technical systems: a total concept theory for engineering design. Springer, Berlin

INCOSE (2010) Systems engineering handbook: a guide for system life cycle processes and activities, INCOSE-TP-2003-002-03.2. International Council on Systems Engineering, San Diego

Iwasaki Y, Fikes R, Vescovi M, Chandrasekaran B (1993) How things are intended to work: capturing functional knowledge in device design. In: Proceedings of international joint conferences AI, AAAI

Jänsch J (2007) Akzeptanz und Anwendung von Konstruktionsmethoden im Industriellen Einsatz: Analyse und Empfehlungen aus Kognitionswissenschaftlicher Sicht. Dissertation. Produktentwicklung und Maschinenelemente, Technical University of Darmstadt, Darmstadt

King B (1989) Better designs in half the time: implementing QFD quality function deployment in America. GOAL/QPC, Methuen

Kitamura Y, Mizogushi R (2007) Ontology-based systematization of functional knowledge. J Eng Des 15(4):327–351

Kitamura Y, Kashiwase M, Fuse M, Mizogushi R (2004) Deployment of an ontological framework of functional design knowledge. Adv Eng Inform 18:115–127

Kleinsmann M, Valkenburg RC (2008) Barriers and enablers for creating shared understanding in co-design projects. Des Stud 29(4):369–386

Kreimeyer M, Lindemann U (2011) Complexity metrics in engineering design: managing the structure of design processes. Springer, Berlin

Kruchten P (2004) The rational unified process: an introduction. Pearson Education, Upper Saddle River

Kurfman M, Stock M, Stone RB, Rajan J, Wood KL (2003) Experimental studies assessing the repeatability of a functional modeling derivation method. J Mech Des 125(4):682–693

Lindemann U, Maurer M, Braun T (2009) Structural complexity management. Springer, Berlin

Maher M, Tang H (2003) Co-evolution as a computational and cognitive model of design. Res Eng Des 14(1):47–63

Maier JR, Fadel GM (2001) Affordance: the fundamental concept in engineering design. In: Proceedings of the ASME international

design engineering technical conferences & computer and information in engineering conference IDEC/CIE

Matzen D (2009) A systematic approach to service oriented product development. Dissertation, DTU Management Engineering, Technical University of Denmark, Copenhagen

Müller P, Schmidt-Kretschmer M, Blessing LTM (2007) Function allocation in product-service-systems: are there analogies between PSS and mechatronics? In: Proceedings of the AEDS workshop

Nagel RL, Midha PA, Tinsley A, Stone RB, McAdams D, Shu L (2008) Exploring the use of functional models in biomimetic conceptual design. J Mech Des 130(12):2–13

Nevala K (2005) Content-based design engineering thinking. In the search for approach. Dissertation, University of Jyväskylä, Jyväskylä

OMG (2012) OMG systems modeling language (OMG SysMLTM) specification. http://www.omg.org/spec/SysML/1.3/. Last visited: April 12th 2014

Ookubo M, Koji Y, Sasajima M, Kitamura Y, Mizogushi R (2007) Towards interoperability between functional taxonomies using an ontology-based mapping. In: Proceedings of 16th international conference on engineering design—ICED

Pahl G, Beitz W (1977) Konstruktionslehre. Springer, Berlin

Pahl G, Beitz W, Feldhusen J, Grote KH (2007) Engineering design: a systematic approach. Springer, Berlin

Patton MQ (2002) Qualitative research and evaluation methods. Sage, Thousand Oaks

Poon J, Maher M (1997) Co-evolution and emergence in design. Artif Intell Des 11(3):319–327

Rodenacker WG (1970) Methodisches konstruieren. Springer, Berlin

Roozenburg NFM, Eekels J (1995) Product design: fundamentals and methods, A Wiley series in product development planning, designing, engineering. Wiley, Chichester

Ropohl G (2009) Allgemeine technologie: eine systemtheorie der technik. Univ.-Verl. Karlsruhe, Karlsruhe

Sakao T, Shimomura Y (2007) Service engineering. A novel engineering discipline for producers to increase value combining service and product. J Clean Prod 15(6):590–604

Scheffer L, Lavagno L, Martin G (2006) EDA for implementation, circuit design, and process technology. CRC Press, Boca Raton

Sen C, Summers JD, Mocko GM (2010) Topological information content and expressiveness of function models in mechanical design. J Comput Inf Sci Eng 10(3):1–11

Sen C, Summers JD, Mocko GM (2013) A formal representation of function structure graphs for physics-based reasoning. J Comput Inf Sci Eng 13(2):021001–021013

Shai O, Reich Y (2004) Infused design: I. Theory. Res Eng Des 15(2):93–107

Shostack GL (1982) How to design a service. Eur J Mark 16(1):49–63

Simon H (1973) The structure of ill-structured problems. Artif Intell Eng Des Anal Manuf AI EDAM 4(3–4):181–201

Srinivasan V, Chakrabarti A, Pal U, Ranjan BSC, Ojha SP, Ranganath R (2011) Supporting process and product knowledge in biomimetic design. Int J Des Eng 4(2):132–158

Srinivasan V, Chakrabarti A, Lindemann U (2012) A framework for describing functions in design. In: Proceedings of 12th international design conference—DESIGN

Stone RB, Wood KL (2000) Development of a functional basis for design. J Mech Des 122(4):359–370

Szykman S, Racz J, Sriram R (1999) The representation of function in computer-based design. In: Proceedings of the international conference on design theory and methodology—DTM

Tan AR (2010) Service-oriented product development strategies. Dissertation, Management Engineering, Technical University of Denmark, Copenhagen

Tukker A, van den Berg C, Tischner U (2006) Product-services: a specific value proposition. In: Tukker A, Tischner U (eds) New business for old Europe. Greanleaf Publishing Ltd, Sheffield

Ullman D (2010) The mechanical design process. McGraw-Hill series in mechanical engineering. McGraw-Hill Higher Education, Boston

Ulrich K, Eppinger SD (2008) Product design and development. McGraw-Hill Higher Education, New York

Umeda Y, Tomiyama T (1997) Functional reasoning in design. IEEE Expert 12(2):42–48

USDoD (2001) Systems engineering fundamentals. Defence Acquisition University Press, Fort Belvoir

van Beek TJ, Tomiyama T (2009) Connecting views in mechatronic systems design, a functions modelling approach. In: Proceedings of ASME 2009 international conference on mechatronic and embedded systems and applications

van Eck D (2010) On the conversion of functional models: bridging differences between functional taxonomies in the modelling of user actions. Res Eng Design 21(2):99–111

van Ecke D (2010b) Explaining and relating different engineering models of functional decomposition. In: Proceedings of design research society (DRS) international conference

VDI (2004) VDI 2206—design methodology for mechatronic systems. Beuth Verlag, Berlin

Vermaas P (2009) The flexible meaning of function in engineering. In: Proceedings of 17th international conference on engineering design—ICED

Vermaas P (2011) Accepting ambiguity of engineering functional descriptions. In: Proceedings of 18th international conference on engineering design—ICED

Vermaas P (2013) On the co-existence of engineering meanings of function. Four responses and their methodological implications. Artif Intell Eng Des Anal Manuf AI EDAM 27(3):191–202

Visser W (1991) Evocation and elaboration of solutions: different types of problem-solving actions: an empirical study on the design of an aerospace artifact. In: Proceedings of the third COGNITIVA symposium

Warell A (1999) Introducing a use perspective in product design theory and methodology. In: Proceedings of the ASME international design engineering technical conferences & computer and information in engineering conference IDEC/CIE

Watson D (1992) Correcting for acquiescent response bias in the absence of a balanced scale. An application to class consciousness. Sociol Methods Res 21(1):52–88

Weber C (2007) Looking at 'DFX' and 'Product Maturity' from the perspective of a new approach to modelling product and product development processes. In: Proceedings of the 17th CIRP design conference

Weilkiens T (2008) Systems engineering mit SysML: modellierung, analyse, design. dpunkt.verlag, Heidelberg

Yin RK (2009) Case study research: design and methods, applied social research methods series. Sage, Los Angeles