Adaptation of the integrated function modelling framework

Merlin F. Krüger,[1] Kilian Gericke[1]

[1]University of Rostock

Abstract: The integrated function modelling framework is used for functional analysis of technical moderate complex systems, but requires specific adaptation in modelling to visualize the behaviour of specific technical systems. We have applied the integrated function modelling framework on a usual technical system within a usual environment of a design project and documented the steps of adaptation. Two different applications had been made with two different ways of adapting the method. A Comparison of the results with other existing applications of the integrated function modelling framework indicates that the application of the method differs for various process types. Three different process types have been identified for the provided method.

Keywords: Design structure matrix, Integrated system modelling, Design methods, Process types

1 Introduction

System modelling, as a means to support analysis and synthesis during evolutionary design activities, often requires the usage of multiple models, especially in interdisciplinary projects. Various models focus on different design entities by using different terminology. Thus, communication between different members of the design team is hampered. The Integrated Function Modelling (IFM) framework is a method that is intended to support interdisciplinary collaboration and reduces the challenges of varying understanding of the term function by integrating a range of entities that are used to create discipline-specific function models (Eisenbart *et al.*, 2017).

It has been observed that the IFM framework provides benefits in most applications. However, some systems pose challenges for the modeller. These challenges concern modelling granularity, modelling procedure, limitations for adapting the IFM framework and training material (examples and guidance for instance).

Concerning model granularity in the modelling process, Maier et. al. (2017) distinguished between cost, quality, data and modelling practice as the four dimensions of challenges for choosing an appropriate level of detail in modelling activities. The cost limit is often the driving factor of decision-making in a modelling process. Model quality may improve with the amount of detail, but it also increases expenditure time and costs. The availability and quality of data are important factors as models are often built on data from existing repositories. "The skill of modellers and users is another factor that has to be taken into account" (Maier, Eckert and John Clarkson, 2017). Next to the level of detail, the modelled information must not be too abstract to ensure the modeller's and user's understanding.

The IFM framework supports a coherent application in practice by using design structure matrices. It allows flexible adaptation in order to ease its application and future development. However, the understanding of adaptation needs of practitioners, and specific adaptations of the modelling framework are yet limited. Existing applications of the IFM framework by Eisenbart et. al. (2015, 2017, 2020) already led to adaptations of the IFM framework for individual purposes with an increase in expenditure time. That could be an indicator of a posterior refusal of the method in practice because, next to the inconsistency of terminology, high expenditure time is a reason for many firms to refuse to adopt design methods for their design projects (Schmidt-Kretschmer and Blessing, 2005). For instance, Pannenbäcker's (2001) study of 400 participants shows that nearly 75 percent refuse to use design methods, reasons including inconsistency of terminology and high expenditure time.

To make the IFM framework more accessible and prevent it from a refusal in practice, it must allow specific adaptation for individual purposes. However, the understanding of advisable adaptations for specific situations is limited yet. This paper aims to contribute a better understanding by differentiating modelling tasks and conducting a case study that compares different adaptation scenarios.

2 The Integrated Function Modelling Framework

The IFM framework (Fig. 1) is a method for modelling and analysing the functionality of technical, moderate complex systems. It depends on Design Structure Matrices (DSM), Multi-Domain Matrices (MDM) and flow modelling. It specifically aims at relating between different function modelling perspectives of different disciplines. Therefore, using different views to describe the functionality of a system. These views are interlinked through mutually shared header rows and header columns, including a process flow view, state view, actor view, use case view, effect view and interaction view (Eisenbart *et al.*, 2017). A change in the order of the framework's views (see Fig. 1) makes it more flexible for individual purposes. On the assumption that the adaptation of the IFM framework improves its usability, it is unclear yet which adaptations of the IFM framework are possible. Eisenbart et al. (2015, 2017, 2020) propose the following possibilities for adaptations for individual applications:

- Removing views
- Adding views
- Changing granularity of views
- Changing system boundaries

If one view is unnecessary or unusable, it can be removed (effect view in Fig. 1, for instance). On the contrary, more views can be added and placed as needed, based on the principle of DSM. It is possible to increase the focus by changing the granularity, e.g. by defining actors on another level of detail, describing the transitions between processes with zoom-in/zoom-out or differentiating between actual state and intended state (Krüger *et al.*, 2020). Furthermore, the system boundaries can be changed by adding or removing actors, etc. and adapting the initial conditions.

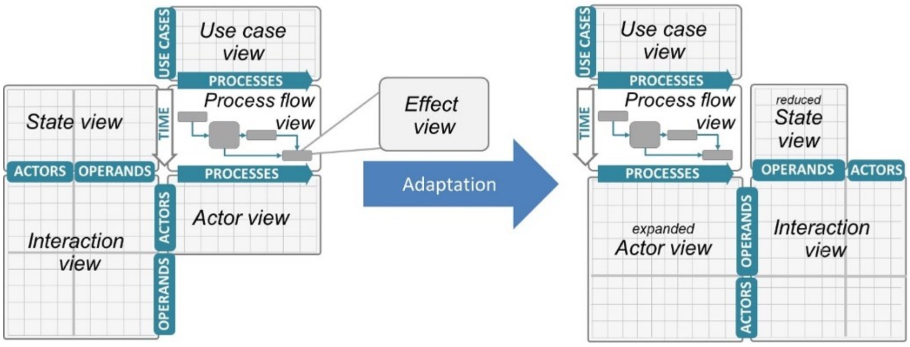

Figure 1. Integrated Function Modelling framework (Eisenbart *et al.*, 2017).

While different possibilities for adaptation of the IFM have been proposed, no guidance for specific adaptations exists. Previous applications of the IFM suggest that adaptation depends mainly on the technical system that is to be modelled. But the causal dependency between the specific technical system and its specific adaptation of the IFM framework is unclear yet. Technical systems differ in various aspects. While technical systems can be differentiated along alternative axes it is unclear which categorization is suitable for this purpose.

Ways of differentiating process types

Technical systems are the operators of transformation systems that impact in various ways in order to fulfil the defined transformation (Hubka and Eder, 1988). Further, they depend on the causal-mechanistic worldview which means, that every action has a cause and simultaneously occurs as a cause itself. The cause depends on three parts, circumstances, conditions and a source. An event is expected to occur always in the same way with exactly the same three parts. Considering a specific *type* of system as a cause for an adaptation of the IFM framework, a differentiation of system types would allow recommending adaptations of the IFM framework in order to improve its application. Further, a suitable categorization for the application of the IFM framework has to be carried out.

Hubka and Eder (1988) define different categories: such as typical engineering sciences, e.g. thermodynamics, fluid mechanics, control and regulation engineering, etc. Also, they define characteristics and criteria, such as the structure complexity, kinds of elements, system with product contact or without product contact, or the purpose of a system that is based on the model of a transformation process. Browning et. al. (2006) even consider processes as a type of system that "has received much less attention as such." They describe systems to be:

- "A regularly interacting or independent group of items forming a unified whole
- A group of devices or artificial objects or an organization forming a network especially for distributing something or serving a common purpose
- An organized or established procedure
- An integrated set of elements that accomplish a defined objective."

Furthermore, they define five systems, related to each other, thus as process system, organization system, product system, tool system and goal system. Dori and Sillitto (2017) do neither define technical system types nor focus on the processes, but they highlight the need for differentiation of various system types in general, as a prerequisite to describe the different interpretations of systems by individuals or communities, such as human-made, naturally occurring and hybrid systems. Due to the design process in medical product development, Spallek and Krause (2016) categorize three different process types "special design, specific adaptation, and standardised individualisation" for additive manufacturing as a basis for the underlying aim of the project and its individual requirements for customisation and personalisation. Albers et. al. (2016) introduced the integrated-Product-engineering-Model (iPeM), based on applications of process models, as an integrated approach, which aims to fill in the gap between process management and engineering design. They faced the challenge to capture different product generations within one model, therefore, a higher adjustment was required and achieved by using different layers to reach higher transparency and usability for various purposes. Necessary for random processes' classification in industrial control systems, Prokhorenkov et. al. (2007) relate the four classes: "processes with discrete states and discrete time; processes with discrete state and continuous time; processes with continuous state and discrete time; processes with continuous state and continuous time." Even if it depends on control systems, using the time and state as characteristics to differentiate various process types is a principle that also could be useful for the IFM framework. Most of the mentioned approaches are facing scenarios at different stages in the product development process, but they lean on a specific categorization for process types to allow adaptation for individual purposes. The approaches of Albers et. al. (2016) and Browning (2006) focus on process management, and the approach by Dori and Silitto (2017) is broadly considered. The process types by Spallek and Krause (2016) focus on reproducibility and manufacturing effort instead of impacts of the functional behaviour – the IFM framework is used for functional analysis. Hubka's (1988) differentiation of technical systems, which relates system characteristics to the model of a transformation process, seems to be a suitable vantage point for studying adaptations of the IFM framework as the central view of the IFM framework is the process flow view which is based on the model of a transformation process. While previous adaptations of the IFM allow concluding, based on the created IF model, which views were added, removed and which granularity was used and how this related to system types, these models do not allow a conclusion on the modelling process and the sequence of modelling steps.

In this paper, a case study is presented that documents the steps of adapting the IFM framework. Furthermore, the reported application of the IFM is compared with other previous applications to define different process types and capture indicating characteristics.

3 Applying the IFM framework on a Partition System – A Case Study

The case study investigates the use of the IFM framework during the development of a partition system for a heavy goods vehicle (HGV). One of the authors was involved in this project as a design engineer and was responsible for the application of the IFM. The development process aimed to reduce the manufacturing effort and costs of the whole system without affecting its functionality. Hence, the goal of the project investigated in this case study differs from the previous applications by Eisenbart et al. (2015, 2017, 2020), which focussed on weaknesses of the systems' functionality. Also, this study considers aspects of model granularity in the modelling process, contrary to Maier et. al. (2017), in a realistic project environment characterised by a short duration.

After understanding the partition system's functionality by communicating with other design team members, analysing existing products and CAD models, the system's functionality had to be visualized (not yet analyzed) using the IFM framework within one day. Afterwards, the members of the project team ought to be able to use the built IF model as a basis for functional analysis and communication (remark: an IF model represents a specific system using the IFM framework).

The Partition System of an HGV

Partition systems aim to separate the vehicle's storage room into different compartments, necessary for goods with different temperatures. In many cases, these partition systems are developed adjustable to enlarge or shrink the load compartment length. The system can be seen (and will further be described) as a mobile wall. The functional requirements are:

- Vacuum ventilation for opening and closing,
- Positioning of a locking system in the upper area,
- Positioning of a locking system in the lower area,
- Mobility of the wall,
- Various widths and heights for individual compartments *(variant construction),*
- Fixation of the wall while opening and closing,
- Using the wall to secure goods

Figure 2 shows different positions of the wall: On the first view on the left-hand side, the wall is closed. In the second view, the vacuum is ventilated. This step is necessary to move the wall from a closed position to an opened position. To ventilate the vacuum, there exist many different solution alternatives. Here, the wall is pulled up in a vertical direction for some centimetres. In the third view, the wall is opened with a circular motion from a vertical position to a horizontal position of the wall, thereby staying close to the top. While the wall is being opened, it can be pushed or pulled along the compartment length to adjust the wall to different states of compartment charge. The requirement that the whole system can be adapted to various widths and heights of the load compartment is important for the development process. It is a variant construction.

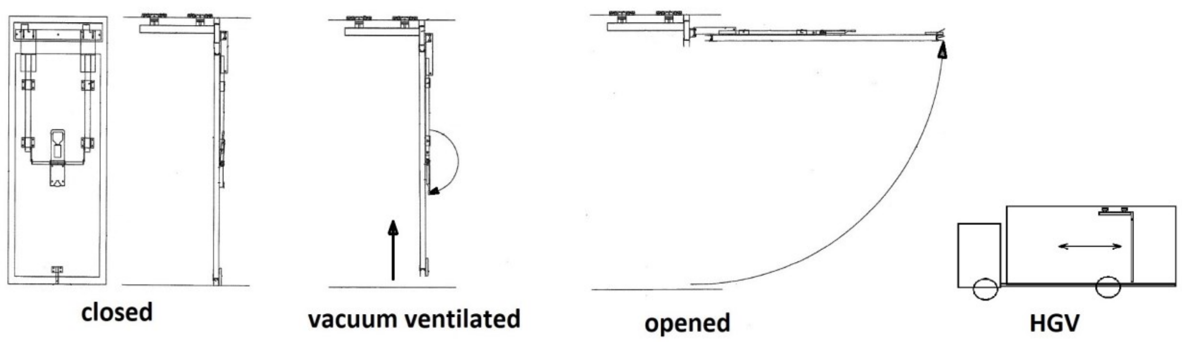

Figure 2. Sketches of a Partition System (Wilke and Wilke, 2004).

4 Case Study Results

During the development process, the authors applied the IFM framework on the given system, the results are shown in modelling scenario 1. This first application was used to analyse the technical system and to identify actors and processes with high potential manufacturing efforts. The results led to a reduction of manufacturing costs of nearly 50 percent. Nevertheless, the authors had to adapt the IFM framework for this application with more effort than expected. The application delayed over a couple of days with a total expenditure time of more than ten hours – Remind: The application was scheduled to one day. In order to compare alternative modelling strategies as a means to adapt the application of the IFM framework, the same system was modelled again after completion of the actual development process. The second IF model, which had been created by using a different modelling strategy, was completed in a total of two hours, and is presented in modelling scenario 2.

Modelling Scenario 1: Recommended possibilities to adapt the IFM framework

As some elements of the original IFM framework caused challenges, the modeller adapted the IFM framework using the recommended possibilities (see chapter 2). This resulted in deleting a substantial share of the original views and adding new views. The scheme of adaptation is outlined in Figure 3. Solid arrows mark unchanged views, while dashed arrows indicate adapted views. For instance, the *processes* have been changed to *sub-functions* for the usage of the actor view, meanwhile, for the usage of the detailed actor view, the definition of actors had been changed in the level of detail. In the actor view, the specification of influences was done using a short description instead of a symbolic indication.

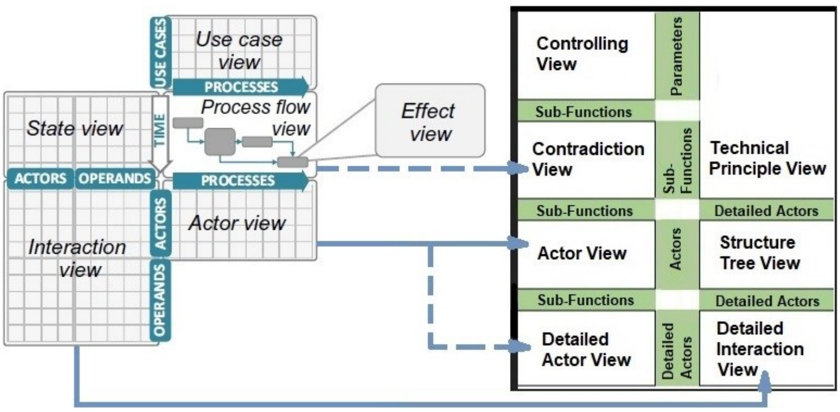

Figure 3. Scheme of Adaptation of the IFM framework for Modelling Scenario 1.

The modelling strategy and sequence of adaptions are explained in the following. At first, the user identified relevant use cases for the partition system. He decided to model one use case – the *application* of the partition system. Decomposition of this use case into multiple use cases was perceived as defining processes instead of use cases. The definition of processes was orientated to the functional requirements of the system (chapter 3). It was not possible to identify a strict process flow because of a missing time parameter of the functional behaviour of the system. The processes were reversible and could be arranged in various sequences. Hence, the process flow view was adapted into a regular DSM without indication of process duration, interpreting the processes as *sub-functions*. This view was called the *contradiction view*. Inspired by Altschuller's work on the TRIZ-method this view aimed to visualize contradictions in the behaviour of the system. The *technical contradictions* are a useful means for the synthesis of the development of a system. Because of the missing process flow view and the fact that the processes were reversible, it was challenging to define the states of actors. Also, the flow of operands did not provide much value in this case. As a result, the state view had been removed and the operands were removed from the interaction view. The challenge of the actor view was to define consistent actors. The user decided to define an actor view on the level of modules and a *detailed actor view* on the level of components. Additionally, the basic function (for instance: transmit, convert, change, combine, etc.) of each component for its dedicated sub-function was defined, based on the definition of Roth (Roth, 1994; Pahl *et al.*, 2007; Bender and Gericke, 2021, p. 248). For Instance, *Force Transmission* (Fig. 4, “Force tr.”) of carrier part 1 for the sub-function *keep wall open*. The detailed actors had been used to create a detailed Interaction-view and a *structure tree view* to visualize the module-component-relations. With the technical principle view*,* the user tried to attribute the function to the underlying working principle. At last, the user visualized the impacts of *controlling parameters* on the different sub-functions of the system. The controlling parameters were defined as descriptions that control the varying requirements, e.g. the various width of the load compartment was defined as the parameter *compartment width*. Further, this view aimed to establish a direct connection between some of the main functional requirements and related sub-functions.

	X	X	X			X	**compartment width**						
	X	X	X	X	X	X	**compartment height**						
				X	X		**height translation**						
		X					**compartment number**						
		X	X				**rail offset**						
		X					**lock offset**						
		X					**lock distance**						
tighten cylinder	**secure loads**	**upper lock**	**move wall**	**open wall**	**kepp wall open**	**ventilate vacuum**							
							tighten cylinder		Thickness			Form	Form
		X	X	X	X		**secure loads**	Form	Thickne	Width	Width	Form	Form
	X		X	X	X	X	**upper lock**	pattern					
	X	X		X	X	X	**move wall**	Length					
	X	X				X	**open wall**	Form	Thickne	Width	Width	Length	Form
	X	X					**kepp wall open**	Form	Thickne	Width	Width	Length	Form
	X	X	X	X			**ventilate vacuum**					Form	
tighten cylinder	**secure loads**	**upper lock**	**move wall**	**open wall**	**kepp wall open**	**ventilate vacuum**		**Upper Part 1**	**Upper Part 2**	**Upper Part 3**	**Upper Part 4**	**Arm-Part 1**	**Arm-Part 2**
O	O		O	O	O		**Upper Assembly**	X	X	X	X		
X				X	X	X	**Arm assembly**					X	X
		X		O		X	**Vacuum-ventilate assembly**						
			X		O		**Carrier assembly**						
	X	X	O	O	O		**upper-lock assebmly**						
X	O			O	O	O	**arm-extension assembly**						
X				X	X		**cylinder**						
	X	X	O				**lower-lock assembly**						
tighten cylinder	**secure loads**	**upper lock**	**move wall**	**open wall**	**kepp wall open**	**ventilate vacuum**		**Upper Part 1**	**Upper Part 2**	**Upper Part 3**	**Upper Part 4**	**Arm-Part 1**	**Arm-Part 2**
	Force tr.	Variety	Force tr.	Force tr.	Force tr.		**Upper Part 1**		X				
Force tr.	Force tr.			location bearing			**Upper Part 2**	X		X	X		X
	stability			stability	stability		**Upper Part 3**		X				X
	stability			stability	stability		**Upper Part 4**		X				
			Force transm	Force tr.	Force tr.		**Carrier Part 1**						
			Force tr.	Force tr.	Force tr.		**Carrier Part 2**						
			Force tr.	Force tr.	Force tr.		**Carrier Part 3**	X					
				Force tr.	Force tr.		**Carrier Part 4**						
			stability	Force tr.	Force tr.		**Carrier Part 5**						
Power introducer				function su	Positioning Power		**cylinder**		X				
	Force tr.	active	inactive			inactive	**lower locksystem**						
			Motion	Force tr.	Force tr.		**Move support**						
Function support							**tighten threat**						X

•••

Figure 4. Shortened Application of the IFM framework for modelling scenario 1.

Modelling Scenario 2: Adapting the Modelling Strategy

The second application aimed to follow contrary modelling and adaptation strategy. Here, it was tried to maintain the basic views and structure of the IFM framework. The only adaptation was to cut out unnecessary or unusable views, and change the order of views. For instance, as in scenario 1, operands in the system were neglected. Hence, the actor view and the interaction view included only actors as schemed in figure 5. Contrary to modelling scenario 1, where only one use case was defined, multiple use cases consisting of fewer processes were defined. One major problem of the first modelling scenario was caused by the definition of the sub-process *secure loads*. This process was difficult to allocate in a process flow. Hence, in scenario 2, this *process* had been defined as an additional use case. The main use-case still remains the *application* but was named as *change compartment length* to clearly distinguish it from the first modelling scenario. The third use case was defined as *preparing* and includes the handling of the wall to *tighten the cylinders* – For the assembling of the system, the cylinders are untightened, after the assembling, the cylinders have to be tightened to ensure the upper position of the wall when it is opened.

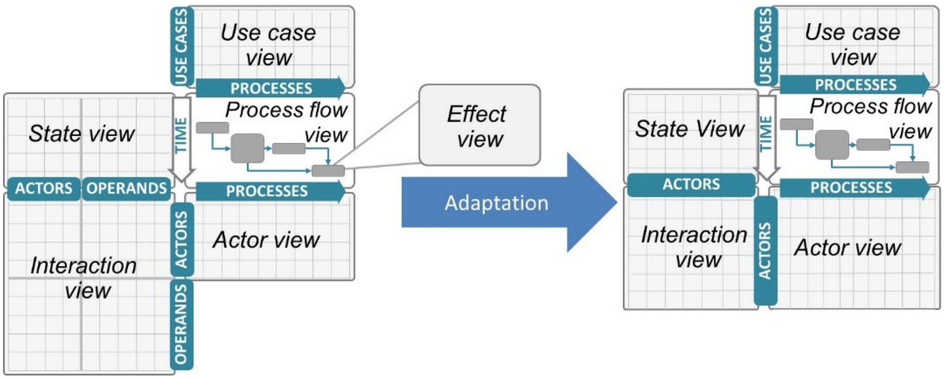

Figure 5. Scheme of Adaptation of the IFM framework for Modelling Scenario 2.

The process flow view includes all processes of the three use-cases (Fig. 6). The difficulty for the designer was to differentiate between the different use-cases because in most cases *changing the compartment length* and *securing loads* are executed sequentially. And these two use cases partially include the same processes. Visualizing this scenario in one process flow view seems difficult. Only *preparing* is executed individually - usually as the last process of the assembling. While still no time parameter exists, the chosen modelling strategy allowed to model a logical sequence of processes. Every reversible process requires defining an additional reversal process. An equivalent is the definition of states in the state view. Hence, the user of the partition system is given the ability to change the sequence of the processes at any given time.

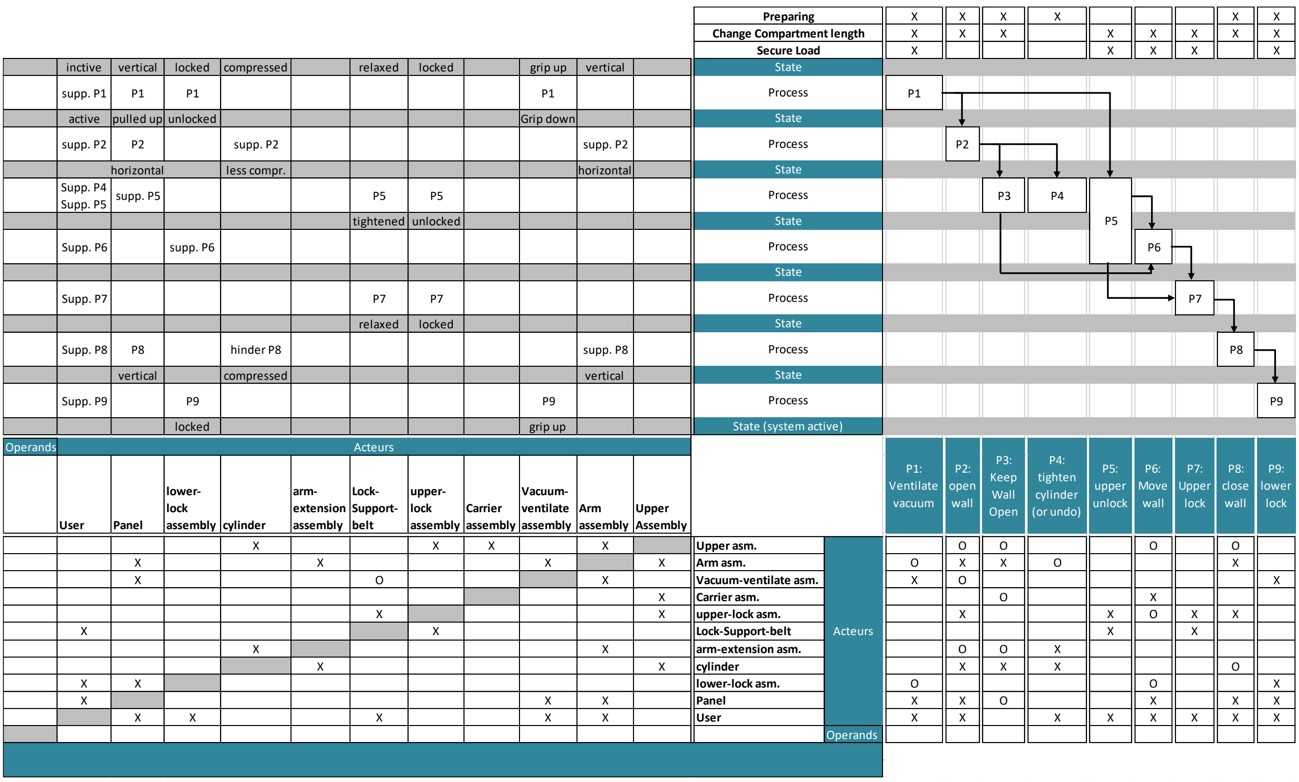

Figure 6. Application of the IFM framework for modelling scenario 2.

5 Discussion

The case study presents two different ways of adapting the IFM framework. Modelling scenario 1 draws from experiences of further applications by Eisenbart et. al. (2015, 2017, 2020) to adapt the IFM framework in its structure and views by removing or adding views. Table 1 includes reflection on the new views by the user, based on the usefulness of each view in the development process of the partition system. A positive recommendation means that the view was useful for the development process and generated a benefit for the IFM framework, a negative one means the opposite. The controlling view was based on the requirements and could have great potential for the framework. But this view only worked because there already existed a parametric CAD model of the technical system.

Table 1. Recommendation of Views in the Application of Modelling Scenario 1 ("+" positive; "-" negative; "+-" neutral).

Views	Aim	Recommendation
Controlling view	Connection between requirements and processes	+ -
Contradiction view	Show technical contradictions	+
Detailed actor view	Basic functions of components for dedicated processes	+
Structure tree view	Module-component-structure	-
Technical principle view	Show special feature of a component	-

These parameters were used to translate the functional requirements and build a connection between controlling parameters and processes of the system. Without the parametric model, this view could not have been created. Hence, the idea behind this view is promising, but the creation is too time consuming yet. The structure tree view did not give any advantage and the technical principle view is too time-consuming to create, both are not recommended. The contradiction view showed the reason for every limitation of each process of the system. Based on this view, the designers identified contradictions in the technical system and were able to develop different solutions for single sub-functions. Visualizing contradictions directly could be a benefit for the IFM framework in general. With the detailed actor view, the designer was able to combine different parts with the same basic function, and then change the geometry of the considered area of the system. For instance, the carrier parts 1 to 5 were only used for force transmission or stability (see "Force tr." in Fig. 4). Therefore, the designer was able to combine these parts and change the geometry and assembly structure of the whole area. A new kind of carrier assembly could be implemented based on the combination and without any impact on other sub-functions of the system. The procedure of combining parts with the same basic function was a very powerful means to reduce manufacturing effort and costs. The original IFM framework already has an actor view. Hence, the basic functions could replace the actual *X* and *O*. There exist different definitions of basic functions with different quantities (Roth, 1994; Pahl *et al.*, 2007; Bender and Gericke, 2021, p. 248). Suitable for this purpose, Stone and Wood (2000) filtered a high number of basic functions and synonyms by eight defined function classes. The use of function classes instead of basic functions to describe the implementation of actors in processes still allows analysing the actor view on higher detail, but also ensures the user's comprehension by not being too abstract – Instead of using function classes, the use of basic functions may be too abstract. Further, the function classes can be used as a basis for synthesis. Another benefit of using function classes in the actor view is the level of detail, because the user is forced to describe the function class of an actor, and if the actor has more than one function class, or the main function class cannot be defined, it has to be split up (defining the actors on higher detail). This supports decision-making in the level of detail for defining actors, which has been challenging yet. One argument against the implementation of function classes is that the use is still more abstract than *X* and *O*.

Differentiation of process types for adapting the Modelling Strategy

For both modelling scenarios, the user was certain in his course of action. At the beginning of the development process, the user was not able to create an IF model using the original framework, although he had former experiences with applying the method. Thus, the adaptation of the IFM framework was necessary. While modelling scenario 1 had substantial changes to the original framework as described before, modelling scenario 2 is based on an IF model which basically uses the original views and structure. The specific adaptation in this scenario concerned the modelling strategy. In modelling scenario 2, the user aimed to model the system's functionality without changes to the original structure of the IFM framework. This required rethinking previously established mental models of the system. The resulting IF model visualizes the functionality of the system well.

The authors tried to better understand the specific characteristics of this modelling task that required adaptation. Thus, they compared modelling scenario 2 with previous modelling activities they have been involved in or which they could reconstruct based on existing documentation. Modelling scenario 2 was compared with IF models of a coffee machine (Eisenbart *et al.*, 2017), a cigarette filter machine (Krüger *et al.*, 2020), and a glue gun (Gericke and Eisenbart, 2017). The application of the IFM framework on the partition system seems similar to the application on a coffee machine by Eisenbart (2017). But there is a major difference between these systems. For the coffee machine, there is a defined sequence of processes that is performed with a defined start to a defined end. The time of execution for every single process is defined as well. In contrast, for the partition system, the processes are arbitrarily sequenced with an additional speciality that every process is reversible, and the duration is not defined. Considering the application on a cigarette filter machine by Krüger et. al. (2020), there is neither a defined sequence nor a reversible sequence. The processes run in an iterative loop, without a defined end, i.e. from a repetitive sequence. This behaviour seems similar to the application on a glue gun by Gericke and Eisenbart (2017). Hence, the *type of process* can be used as a characteristic to distinguish between different applications. Based on the mentioned applications of the IFM framework the following process types can be distinguished:

- ***Sequential system*** based on a defined sequence of activities
- ***Iterative system*** based on iterative sequence of activities
- ***Reversible system*** based on reversible activities (that may follow an arbitrary sequence)

These process types affect the modelling strategy. Matching the right process type can be a helpful means for a successful application of the IFM framework. For the application of the IFM framework on a system with sequential processes, the original IFM framework needs no adaptations. For further analysis of the adaptation of the IFM framework for sequential processes, it has to be considered, how to visualize different outputs for the same basic process. Based on the application by Eisenbart, the question could be: What if the output of the process *heating water* changes between the use cases *prepare a cup of coffee (100°C)* and *prepare a cup of tea (60°)*? Two possible options are presented in figure 7. On the left-hand side, the different outputs are marked in the state view, the use of different colours in the use case view could support the differentiation. On the right-hand side, the use cases have been separated into their own process flow view, state view and actor view, based on a diagonal visualization by Krüger et. al. (2020). This could further be discussed.

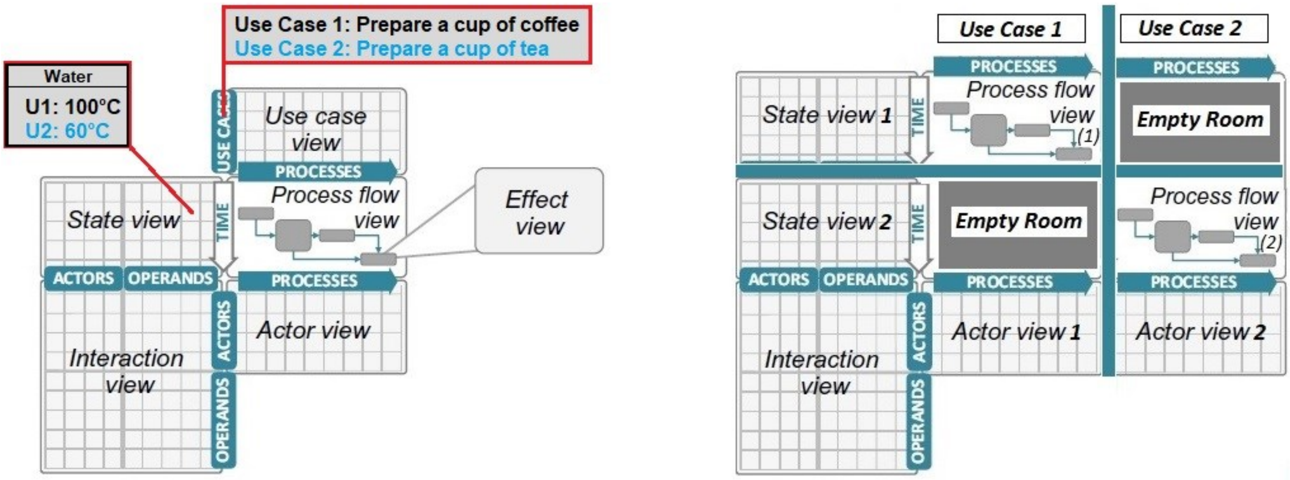

Figure 7. Possibilities of visualizing different outputs.

For the application of the IFM framework on systems with iterative processes, the specific process flow views for different use cases have to be visualized diagonally (see Fig. 7) because the processes of one use case do not take part in another use case of the system. There are multiple actor views, process flow views and state views, one of them for each use case. The iteration also leads to a higher focus on the difference between an intended state and an actual state, as well as the transition spot between processes (Krüger *et al.*, 2020).

For the application of the IFM framework on systems with reversible processes, the user has to define reversal processes for each process and as a hint, it is useful to define multiple use cases to capture the sequence. Compared to the sequential processes, reversible processes do not differ and can be visualized in one process flow view as well as the states in the state view. Because there is no defined time parameter, the user has to define only the logical sequence for the process flow view.

Keeping the original views and adapting only the modelling strategy also increases comprehension of the method in practice. The modelling strategy includes hints and rules on how to apply the IFM framework on a technical system for different process types. Actually, the categorizing is based on the characteristic of the process flow. It can be amplified by additional characteristics to describe the three process types (or more). For instance, a powerful means to the differentiation of process types would be by taking the flow of operands into account. Table 2 shows the inclusion of the basic flows of energy, information and material for the three defined process types, based on the compared applications. But it is unclear yet if this differentiation fits in general and how to interpret the basic flows for the application of the IFM framework.

Table 2. Comparison of the Process types “sequential system”, “iterative system”, “reversible system”.

Characteristic	sequential system	iterative system	reversible system
Processes	sequential	iterative	reversible
Energy	Yes	Yes	Yes
Material	Yes	Yes	No
Information	Yes	No	No

6 Conclusion

To improve the usability of the IFM framework, the method has to be adapted for individual purposes. To understand the cause for specific adaptations for specific applications, the authors analysed the steps of adaptation based on a case study. The case study has shown two different possibilities of adapting the IFM framework for individual purposes. On one hand, there are recommended ways of adapting, such as removing views or adding views. Following this approach, new views have been created that could be integrated into the framework, such as the integration of basic functions or function classes.

On the other hand, there is a way of adapting the modelling strategy. Adapting the modelling strategy reduced the expenditure time by a factor of five and also increased the user's comprehension of the method by maintaining the original structure and views for individual purposes, but it depends on a clear definition of process types. The process flow view is the main view of the method and there have been various activities identified within this view with great impact on the framework, hence, the process flow was chosen as a basis to distinguish between different applications. Existing applications of the IFM framework and the results of the case study were used to define different process types. Comparing the applications, three different process types have been identified: iterative processes, sequential processes, and reversible processes. The modelling strategy has been discussed and a necessary analysis in adapting sequential processes has been mentioned. Additionally, the integration of the different basic flows of operands has been noticed for further exploration.

References

Albers, A. *et al.* (2016) 'IPeM-Integrated Product Engineering Model in Context of Product Generation Engineering', *Procedia CIRP*, 50, pp. 100–105. doi: 10.1016/j.procir.2016.04.168.

Bender, B; and Gericke, K; (2021) 'Pahl/Beitz Konstruktionslehre : Methoden und Anwendung erfolgreicher Produktentwicklung', in Bender, Beate and Gericke, Kilian (eds). Berlin: Springer Vieweg.

Browning, T. R., Fricke, E. and Negele, H. (2006) 'Key concepts in modeling product development processes', *Systems Engineering*, 9(2), pp. 104–128. doi: 10.1002/sys.20047.

Dori, D. and Sillitto, H. (2017) 'What is a System? An Ontological Framework', *Systems Engineering*, 20(3), pp. 207–219. doi: 10.1002/sys.

Eisenbart, B. *et al.* (2015) 'Integrated function modelling: Comparing the IFM framework with SYSML', *Proceedings of the International Conference on Engineering Design, ICED*, 5(DS 80-05).

Eisenbart, B. *et al.* (2017) 'A DSM-based framework for integrated function modelling: concept, application and evaluation', *Research in Engineering Design*, 28(1), pp. 25–51. doi: 10.1007/s00163-016-0228-1.

Gericke, K. and Eisenbart, B. (2017) 'The integrated function modeling framework and its relation to function structures', *Artificial Intelligence for Engineering Design, Analysis and Manufacturing: AIEDAM*, 31(4), pp. 436–457. doi: 10.1017/S089006041700049X.

Hubka, V. and Eder, W. E. (1988) *Theory of technical systems : a total concept theory for engineering design*. Berlin: Springer. Available at: https://ebookcentral.proquest.com/lib/kxp/detail.action?docID=3089886.

Krüger, M. F. *et al.* (2020) 'The Application of the IFM Framework and FIDD Method on an Industrial Cigarette Filter Maker', (September), pp. 10–10. doi: 10.35199/dsm2020.19.

Maier, J. F., Eckert, C. M. and John Clarkson, P. (2017) 'Model granularity in engineering design - Concepts and framework', *Design Science*, 3, pp. 1–29. doi: 10.1017/dsj.2016.16.

Pahl, G. *et al.* (2007) *Engineering Design : A Systematic Approach*. 3. Edition. Edited by K. Wallace and L. Blessing. Springer Vieweg.

Pannenbäcker, T. (2001) *Methodisches Erfinden in Unternehmen: Bedarf, Konzept, Perspektiven für TRIZ-basierte Erfolge*. Gabler Verlag. doi: 10.10007/978-3-322-90156-9.

Prokhorenkov, A. M., Kachala, N. M. and Sovlukov, A. S. (2007) 'Classification of random processes in industrial control systems', *2007 4th IEEE Workshop on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, IDAACS*, (September), pp. 383–387. doi: 10.1109/IDAACS.2007.4488444.

Roth, K. (1994) 'Konstruieren mit Konstruktionskatalogen'. Berlin [u.a.]: Springer.

Schmidt-Kretschmer, M. and Blessing, L. (2005) 'Strategische Aspekte der Konstruktionsmethodik', *DfX-Symposium*.

Spallek, J. and Krause, D. (2016) 'Process Types of Customisation and Personalisation in Design for Additive Manufacturing Applied to Vascular Models', *Procedia CIRP*, 50, pp. 281–286. doi: 10.1016/j.procir.2016.05.022.

Stone, R. B. and Wood, K. L. (2000) 'Development of a Functional Basis for Design', 122(Journal of Mechanical Design).

Wilke, H. and Wilke, K. (2004) 'Isolieraufbau für ein Fahrzeug'. German Patent and Trade Mark Office, Germany. doi: DE102004061161A1.

Contact: Merlin F. Krüger, University of Rostock, Chair of Product Development, Albert-Einstein-Straße 2, 18059 Rostock, Germany, Phone +493814989171, Fax +493814989172, merlin.krueger@uni-rostock.de