Integrating Conventional System Views with Function-Behaviour-State Modelling

T.J. van Beek, T. Tomiyama
Intelligent Mechanical Systems, Delft University of Technology, Mekelweg 2, delft, 2628 CD, The Netherlands
T.J.vanBeek@TUDelft.nl, T.Tomiyama@TUDelft.nl

Abstract
The main contribution of this paper lies in the observations done in industry resulting in an approach of integrating the Function-Behaviour-State (FBS) model with user workflow and interface models to create complex system overview. On one side of the spectrum it focuses on modelling the usage of the system, while on the other side it considers the modelling and managing of interfaces. The choice for both these views is based on an industrial experience with the clinical Magnetic Resonance (MR) imaging system to manage design complexity. The paper gives a real example of the approach.

Keywords:
Design complexity, function model, overview

1 INTRODUCTION

The conceptual engineering design process of complex mechatronic systems rarely starts from scratch. Mostly an existing system will serve as a starting point for the new, changed or improved functionality. Producing high quality design solutions under these circumstances is a difficult task for system architects because they have to consider a large amount of design aspects simultaneously and make sure the solution is obtained in a timely manner. Complexity management is essential in this process but has not yet been satisfactorily addressed in literature and practice [1].

The development of a MR system is presented here as an example of a complex mechatronic system. MR systems have only been around since the early 1980's, and represent a relative young class of systems. Therefore, MR development still evolves rapidly. New functionalities are developed for each product release and compatibility of new functionalities with the existing system is. Besides the design process complexity, the MR design is characterized by a strong multidisciplinary nature (e.g. mechanics, electronics, computer science, materials science, clinical science, and fundamental physics). Managing and coordinating this multi disciplinary product development process is extremely difficult [2] and exceeds the comprehension of a single engineer who cannot understand every detail [1, 3, 4].

The research presented in this paper is a continuation of ongoing research [5, 6]. This research aims at developing a method that supports system architects in their complex design activities by giving them a clear bird's-eye view of the system architecture, see figure 1. By extending the FBS model [7] with additional system views, a consistent system architecture ontology is targeted. Creating a formalism that describes this ontology will allow for semi-automated reasoning on the large number of design concepts and address scaling issues.

The main contribution of this paper lies in the observations done in industry resulting in an approach of integrating the FBS model with user workflow and interface requirements models to create overview. On one side of the spectrum it focuses on modelling the usage of the system, while on the other it considers the modelling and managing of interfaces. The choice for both these views is based on an industrial experience with the MR system to manage design complexity. The paper gives a real example of the approach.

Lindemann and Maurer [8] recognize that controlling product complexity has become an important issue in product development and they state that although reducing complexity is purposeful, it is not favourable at any cost. Controlling complexity is not the same as managing it as this paper proposes. This paper hypothesises that by managing the design complexity with increased overview, the design complexity is decreased.

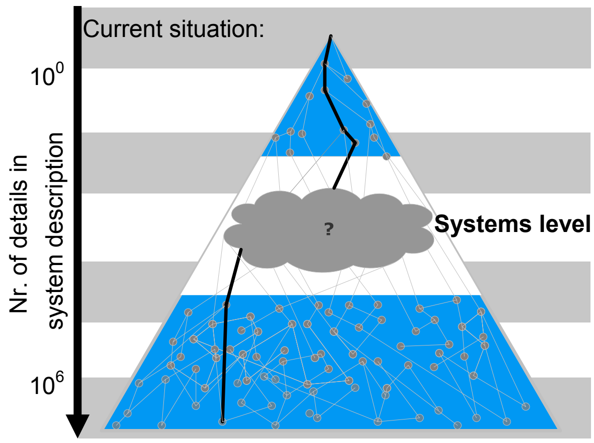


Figure 1. Schematic representation of system complexity.

The Function and Key drivers method (FunKey) [9] method proposes relating system's functions to key drivers and requirements and coupling them in a matrix. The method seeks mainly to provide an easy way of documenting a certain choice for an architecture and its performance, thereby providing the system architect with an overview of his choices. The contribution is focussed at the architect himself. This paper proposes to also facilitate communication and design knowledge sharing among architects and other stakeholders in the design process.

Boersting et al. [10] give an important contribution that relates requirements to functions and deals with complex design to gain product overview by means of models. This enables them to manage and predict change propagation in complex design. They deal with the important relation between requirements and function as where this paper tries to relate workflow to function. Workflow aims more at the modelled usage of the product.

The CAFCR model [11] also recognizes the importance of a system architecture overview. It proposes a decomposition of the architecture into five main views that capture the need of the customer, the functions the product performs, and the design of the product from the conceptual and realization points of view. The work is of importance because it presents a method to create different views, but it does not specify how to implement these methods in a design support tool.

This paper will first give Motivation and background of the research. It will elaborate on the MR system design to illustrate the complexity and it will discuss FBS as a model that can reduce complexity. In section 3 the approach of integrating workflow, FBS and interface models is proposed and in section 4 an example is given using the patient support table of the MR system.

2 MOTIVATION

2.1 Industry-as-Laboratory

The research presented in this paper is part of a research project conducted in close collaboration with the MR division of Philips Healthcare. The goal of this collaboration is to bring the academic and the industrial worlds closer together. Research driven by real industrial problems ensures relevant research topics. Proposed methods and solutions later can be tested using the industrial practice as laboratory. A short introduction to the MR system and development organization will follow to illustrate its complexity.

MR System

MR is a clinical imaging modality that visualizes small changes in the magnetism of nucleus of hydrogen atoms. The magnetic properties will temporarily change once excited by the MR's static and dynamic magnetic fields. The static field strength ranges from 1 tot 3 Tesla and is produced by a superconductive magnet constantly cooled to temperatures close to absolute zero (0K ≈ -273 C°). Combined with a dynamic magnetic field created by large amplifiers and electro magnetic coils, the hydrogen nuclei are excited according to a predetermined waveform.

After the excitation the MR turns into a highly sensitive sensor and measures the magnetic response of a specific part of the human body. Dedicated receiver coils are developed for specific parts of the body to support different clinical applications. In other words, to produce an image of the patients neck another receiver coil is used than when an image of a leg is produced.

The coil sensor signals then need to be captured and processed real time (order of nano seconds) by the data acquisition system. Powerful computers make sure that the signal is conditioned such that an image reconstruction is possible.

Besides the workstation used as an interface between the system and the operator, several computers are embedded to pre- and post-process, control and plan the scans. Three hospital rooms are needed to house the MR system, a technical room with amplifiers, control units and cooling equipment, an examination with the magnet and the patient environment and an operator room with the work station.

From the description above it can be seen that the MR system contains the disciplines; mechanics, electronics, control, physics, material science, software, clinical science and marketing.

Philips Healthcare MR development involves 400 people across all before mentioned disciplines spread over 3 main and several smaller sites all over the world. The software archive contains about 10 different programming languages, resulting in 7 million lines of code. 150 software developers work on the archive concurrently to add new functionality to system.

In preceding work of the authors [5] three main issues in the product development process have been identified:

- Lack of Design traceability
- Lack of Design Understanding
- No support in decomposing the design problem into smaller pieces.

Design Traceability

The transitions from one level of abstraction to another often are iterative processes both ways. Because of the large amount of uncoupled design information content, good traceability of the relations between design aspects in different levels of abstraction is difficult to realize in complex multi-disciplinary design processes.

Design Understanding

There is a need for better traceability of design requirements and system decomposition choices [12]. Both the size of the information embedded in the designed product and the information gathered in the design process is growing. The size of the problems has grown beyond the limits of one person's comprehension [1, 3, 4]. In our research it was estimated by architects that maybe 0.5 % of all employees have a total system overview. Not understanding the overall system is a source of uncertainty and errors in the design.

System Decomposition

System architects decompose the system into smaller sub systems. Where two sub systems meet, an interface should be defined. Creating an ideal interface description for one sub system often conflicts with the ideal interface for another sub system. The systems are highly customizable and therefore configurations exist as a sub set of all available sub systems. It was observed that navigating through the product configuration space is very difficult without models and tools that support the architects.

2.2 Bird's-eye view

To increase design traceability we need models of complex systems that connect high levels of abstraction to low levels of abstraction. Most models used now, do not span different levels of abstractions [13]. For example, a mechanical 3D CAD concerns only the geometry of components, and does not link to functional information. The link between these aspects is missing. They are not considered in parallel and connected, but sequential and only linked in the mind of the designers.

When for example changes are executed in the workflow models of the system, the designer has to determine manually where he has to change the requirement and function models.

To increase system understanding a map (shared model) is needed that communicates the system composition and outline between the architects. A MR system typically has details that reach $O(10^7)$. Other products have similar properties, for example; an aircraft has unique components of this order. Complex mechatronic machines (e.g. mobile phones, medical systems, printers, hybrid car) are controlled by software that has number of lines in the same order of magnitude. At the top level there are abstract functional descriptions. At the bottom, component details of that order are needed. At this level, descriptions are very much mono-disciplinary and their complexity is high but manageable if engineers are provided with dedicated tools. However, the middle layer is systems level multi-disciplinary. The current industrial situation lacks a good way to deal with this level.

Function Modelling

What is needed is a model that connects different design aspects (e.g. models of system usage, requirements, function, interfaces and components) at different levels of abstraction. In this paper a method is proposed based on the function modelling technique of FBS modelling [7]. An FBS approach is considered because it already integrates design concepts at different levels. The FBS model creates system overview from the early, abstract levels of functions through the concepts of objective system behaviour all the way to low level detailed state descriptions.

2.3 Function Behaviour State Model

An FBS model (see figure 2) can be divided in three connected levels; function, behaviour and state level. In the function layer an F hierarchy of the system is maintained. For a complete reference on all the FBS nomenclature and definitions the reader is referred to [7, 14, 15]. Some useful definitions are reproduced here. A function is defined as:

Function = 'a description of behaviour recognized by a human through abstraction in order to utilize it'

In another form function can be defined as 'to do something'. Functions are related to behaviour(s) by means of the many-to-many F-B relationship.

Behaviour = 'sequential one and more changes of state over time'

The behaviour or state transitions of the system are caused by Physical Phenomena (PP). And state is defined as:

State = a triplet where:

E: identifiers of entities included in this state

A: attributes of the entities

R: relations in this state

Relations can occur among entities, between entities and attributes and among attributes.

Figure 3 gives an example of an FBS model with the elements of a paper weight. The subjective function description 'to keep paper from moving' is connected to the objective entities paper weight and paper through a physical phenomenon named gravity. Figure 3 also illustrates the different possible relations.

Extend FBS

Although FBS is a good starting point for creating a multi level of abstraction system overview model, to create complex system transparency some extensions to the FBS model were proposed [5] to adequately address the problems mentioned in section 2.1 of this paper. Three areas were identified where extensions to the current FBS paradigm are needed.

1. Missing modelling entities
2. Ontology Problem
3. System decomposition support

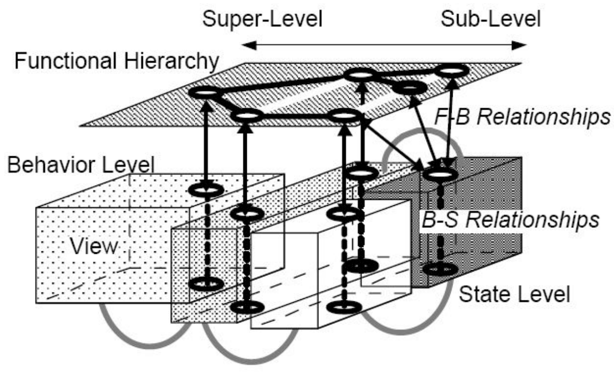


Figure 2. FBS Model scheme

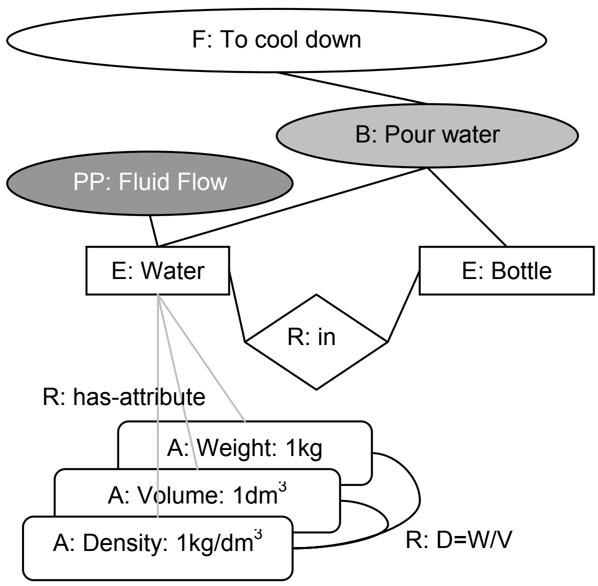


Figure 3. Illustrative FBS example for a single function

Missing modelling entities

High level functions often do not map one-to-one onto user needs and marketing requirements. Other design models, e.g. requirements and usage scenarios, are primarily used to clarify the design task and serve as input for the creation process of a function decomposition of the system.

In [10] they recognize that many methods in engineering design rely on functional models. The main problem with many of these methods is that they can be very sensitive to the quality of their input information and that 'overlooked' relations can bias the results obtained by analysing the models. Therefore, considering as many functional relations as possible is crucial for building functional models This missing link is recognized by [10]. They propose a method to link the function view and the requirements view to improve the ability to predict and manage design change.

Ontology Problem

The defined FBS ontology provides the frame in which the system is captured. If the frame is too narrow it might not allow certain functions to be included into the model. When the frame is too broad it allows all functions to be included, but it will be difficult to create a manageable design object data model since all objects are allowed to be so different. An example of a function that proved difficult in FBS was 'to facilitate cable management' in a system.

System decomposition

FBS currently does not have a facility to consider systems of systems decompositions. The design process almost always means adding or changing functionality to an existing system. This means that the models used to support the design process should facilitate adding new or changing model entities to an existing system model.

This paper addresses the first of the three discussed problem and tries to identify views to extend the FBS model.

2.4 Observations

FBS applied in industry

In the scope of this research a design project of a new functionality was observed and an FBS model was created and evaluated. The new functionality is developed to answer a new clinical application of the MR. The project involves people from the clinical science, marketing and engineering departments of Philips Healthcare MR. The developed system will add hardware and software to the existing main MR system in order to facilitate this new clinical application.

Based on design documentation, interviews and discussions an FBS model was created by the researchers and presented to the architects. The aim of the FBS model was to test whether the FBS model would support the architects with a clear overview of why the system is developed (F), what it does (B) and what it consists of (S). Communication support among architects, between architects and engineers, and other members of the design team is needed to keep overview and understanding of the system.

Observations regarding the FBS model were:

- The first impression of the FBS model is that it's too complex. It has too many nodes and edges to give an instant overview. It needs studying before the model is understood.
- Causal or ordered relations between functions are absent.
- The FBS model bridges high level system functions to detail level state, or components.

Conventional System Views

So called 'workflow' and 'interface' views were used during the design process of the new system. The workflow view described how the developed system should be used and could be used. Multiple possible workflows are identified at the start of the project. Interviews with clinical experts, a marketing expert and engineers all mentioned the workflow view as an important tool to communicate.

To asses the impact of the new system to the existing system the architects construct an interface requirement document. The document consists of a graphical node-edge view of the systems. Named edges in the graph represent interfaces between parts of the system. A worksheet is attached where the requirements on those interfaces can be looked up. The document has one zoom level. A choice was made to do it at a certain modular level and interfaces between these modules are described, but the document does not contain interface requirements for components inside the modules.

3 APPROACH: INTEGRATING VIEWS

As a result from discussions with system architects advantages and disadvantages of the FBS model application were found. FBS was found to be useful in gathering information and containing it in a human understandable manner, the graphical overview it rovides can be improved. Regarding the functional layer it was suggested by the architects to combine the F view with their workflow view, because their workflow view seemed close to the functional view. The workflow views have a sequential form. Their workflow view corresponds to a user scenario described in [16]. There a scenario is defined as;

Scenarios = 'explicit descriptions of the hypothetical use of a product'.

This definition fits the use of the term workflow in the MR development organization. It's an envisioning of the use of the product.

The discussed interface requirements document serves as a starting point for the divide-and-conquer of the design problem for the phases following the conceptual design. Based on the interface specifications engineers who develop part of the system can communicate with other engineers developing other parts. Having a clear definition of the interfaces helps reduce the amount of design errors or forgotten relations in the design.

Constructing the interface document in a flexible manner is experienced as troublesome. In this process the architects rely on their system understanding and experience. A static level of abstraction of the interfaces is problematic. A slight change in the system decomposition changes the interfaces. But a static document as it is used now does not follow these changes. The changes occur frequently in the early phase of the design process.

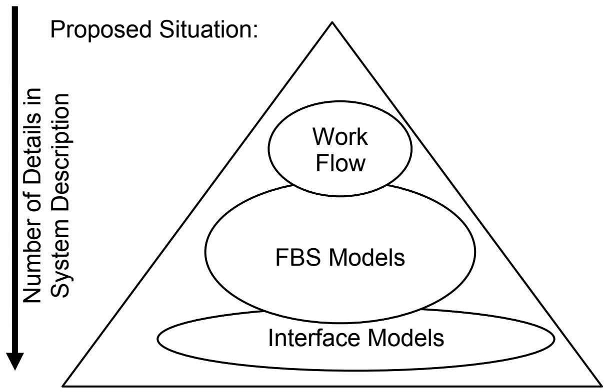


Figure 4. By combining user workflow, FBS and interface models part of the pyramid is covered.

Model:	Advantage:	Disadvantages:
FBS	Bridge high level to low level of design concepts	Time needed to get to know the model. Not intuitive
Workflow	Intuitive for people from multiple disciplines Models the use	Not connected directly to system properties
Interface	Guideline for embodiment and detail design phases. Supports Communication	Static of nature Inflexible to changing design choices

Table 1. Advantages and disadvantages of proposed views

Proposed Method

It's proposed to combine the advantage of both the FBS model and the workflow and interface views. By connecting the models the disadvantages of the individual models are reduced. See table 1 for a short summary of the advantages and disadvantages of the different models.

Workflow-F View Connection

A workflow starts with a story telling of the use of a product from the point of view of a certain stakeholder. Multiple stakeholders may be considered and multiple scenarios per stakeholder are possible. A nice graphical representation of a workflow was found in a common flow diagram. A directed graph with nodes and edges illustrates the workflow of the system. A connection between the workflow nodes and the function nodes is determined manually by using a dependency matrix with Boolean indicators for a connection, see figure 5.

The function view starts with the function tree of the existing MR system and the new functions are added onto that.

No formal workflow descriptions are used. The workflow view has no formalised ontology at this moment. A formalised workflow description could in the future facilitate (semi-)automatic reasoning of the connection between workflow entries and function descriptions.

Both views are developed by the design team in parallel. The workflow view facilitates the discussions among architects and between architects, clinical experts and marketing. Therefore fewer items are overlooked in the function view.

FBS view

When the first function and workflow views are constructed it is time to update the B and S models. Changes and additions to the function model will then be translated into changes of the behavioural and state models as discussed in section 2.3. Figure 6 shows schematically how to update the FBS model.

Interface View

Interfaces exist on different levels in a complex mechatronic system like MR. Depending on the activities, design phase and interest of the architects, different level interface requirements descriptions are needed. High level of abstraction interface requirements could for example be a description of the patient support table to the MR magnet. The patient table and magnet are complex mechatronic systems themselves but they have an interface. The engineering interfaces consist of mechanical, software and physical (e.g. contact surfaces, data streams and the strong magnetic field). A low level of detail interface could be for example a shaft-to-bearing interface inside the patient support table. Therefore, it's desirable to create a 'zoomable' Interface requirement description depending on the activities of the architect.

In figure 7 the proposed method for connecting the FBS model to the interface requirement model is illustrated. The diamond shape nodes in the state model represent FBS entity relations (see section 2.3). Entity relations typically relate attributes of different entities to each other. A relation could for example be named 'On' as with the paper weight mentioned in section 2.3. In the knowledgebase connected to the FBS model [14] it is defined that the 'on' relation allows force to be transmitted and that the magnitude of that force depends on the weight of the paper weight. The interface between the paper and the paper weight can now be described using the relations between entities and their attributes.

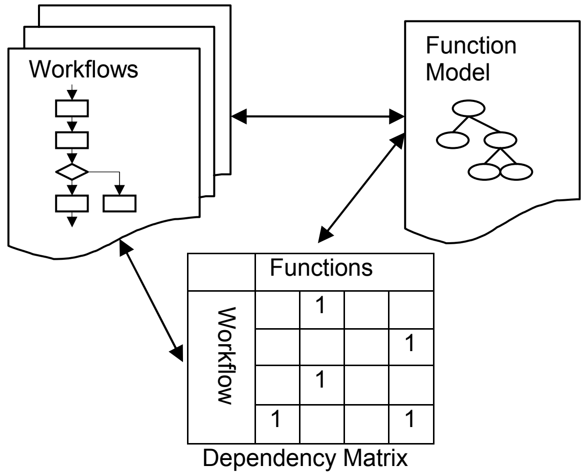


Figure 5. Connecting the Workflow and Function models by Dependency Matrix

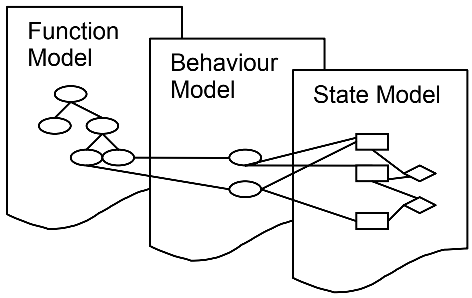


Figure 6. Updating the FBS model.

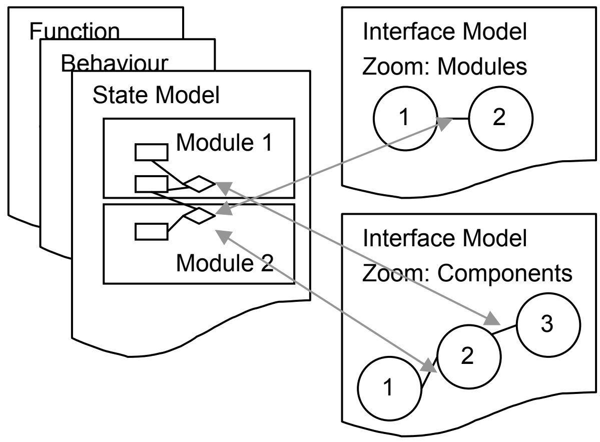


Figure 7. Connection between FBS relations and interface model.

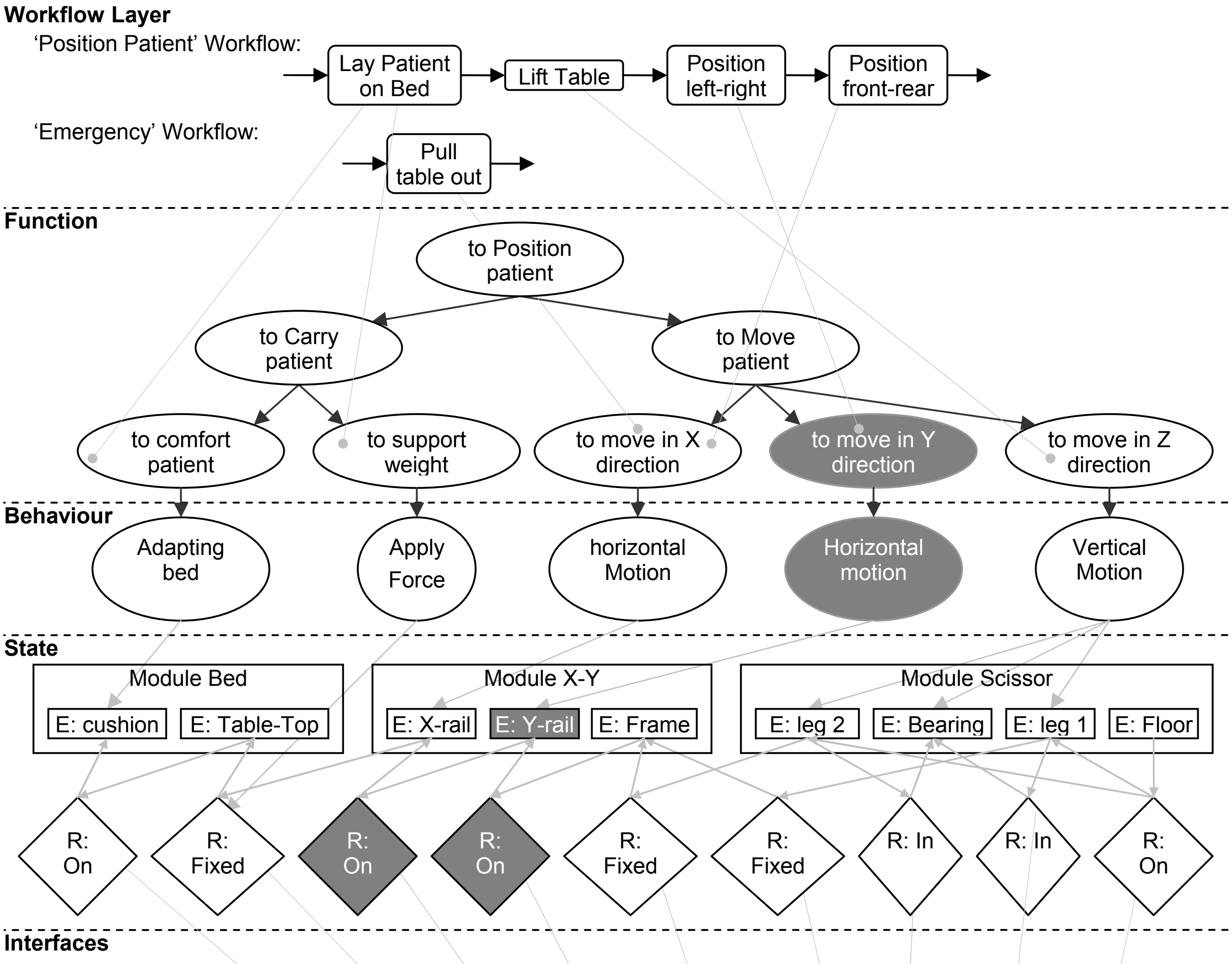
When higher level interfaces need to be described, a grouping of entities occurs. How to define these modules is outside the scope of this paper and the reader is referred to [17-19]. System decomposition and known module boundaries are assumed for now. As displayed in figure 7 with the boxes labelled 'Module 1' and 'Module 2'. Considering both modules, there is one edge that crosses the boundaries of those boxes. This apparently is the only interface between the two modules. The description of this interface would be constructed in a similar manner than that of a component-component interface.

By having the connections between the different models as live-links, a consistent view with the current design situation is realised. Responsibility of keeping the models up to date has to be assigned to specific people in the design team.

4 EXAMPLE: PATIENT SUPPORT TABLE

The patient support sub system of the MR system is displayed here, figure 8, as an example of the above described method. The presented example doesn't have a low level of detail because of the confidential nature of the data and the diagrams become too big.

As the name suggests the patient support table carries the patient during a MR exam.



Modules:	State Entities:	Bed: Cushion	Bed: Table-Top	X-Y: X-rail	X-Y: Y-rail	X-Y: Frame	Scissor: Leg 2	Scissor: Bearing	Scissor: Leg 1	Scissor: Floor
Bed	Cushion		On							
	Table-Top	On		Fix						
X-Y	X-rail		Fix		On					
	Y-rail			On		On				
	Frame				On		Fix		Fix	
Scissor	Leg 2					Fix		In		On
	Bearing						In		In	
	Leg 1					Fix		In		On
	Floor						On		On	

Figure 8. Example of Workflow – FBS – Interfaces Connection

Depending on the type of clinical exam the patient is placed on the table in the down position. Once the patient is on the bed, the table rises to the right level and the table slides into the bore of the magnet. Depending on the body part being examined the table slides in a certain distance.

The main function of the patient support table is to position the patient at a predetermined location. It realizes this function by a vertical (Z-axis) scissor-lift mechanism combined with a horizontal (X-axis) slider mechanism. Al axis are actuated and controlled by the MR host computer or a manual operator.

The Patient support table serves as a good example because it was recently upgraded with a new functionality; A second axis (Y-axis) in the horizontal motion. The new functionality was introduced for a new type of MR systems called ‘Open MR’. An open MR no longer uses one horizontal cylindrical magnet, but it uses two vertical oriented magnets. The patient is positioned in between the two magnets and experiences a more open environment. In the example we have indicated some nodes of the graphs with a different colour. These are the nodes that have been changed/added after the introduction of the new functionality.

The interface model is shown here as a dependency matrix. This matrix shows a big resemblance to the Design Structure Matrix (DSM) [18]. A DSM could be used to determine the modules of the system, but this process is outside the scope of this paper.

The entries in the interface model dependency matrix have properties determined according to the nature of the attributes and entities connected to the relations.

5 CONCLUSION AND FUTURE WORK

This paper has proposed an approach of integrating three different system views to create a better system overview for the system architects. It was found that using the FBS model as a basis and attaching the workflow and interface models an integrated system view can be created that supports the architects in reasoning from the use model all the way to the engineering interfaces.

By creating this overview the complexity of the MR system becomes more transparent and manageable. By making the complexity manageable the system complexity is reduced.

Future work of this research aims at;

- Finding more models as ‘Missing modelling entities’, first candidate being the requirements model.
- Solving the ‘Ontology Problem’. To describe a formal ontology which facilitates the proposed method and allows for semi-automated model reasoning. A tool will be developed to support the model creation process. This includes formalising workflow and interface views.
- System decomposition support. In this paper a decomposition was assumed, but in future work this process should be supported.

6 ACKNOWLEDGMENTS

This work has been carried out as a part of the DARWIN project at Philips Healthcare under the responsibilities of the Embedded Systems Institute. This project is partially supported by the Dutch Ministry of Economic Affairs under the BSIK program.

7 REFERENCES

[1] Szykman, S., et al., *A web-based system for design artifact modeling.* Design Studies, 2000. **21**(2): p. 145-165.

[2] d'Amelio, V. and T. Tomiyama, *Predicting the unpredictable problems in mechatronic design*, in *International conference on engineering design, ICED'07*. 2007, Ecole centrale Paris: Paris.

[3] Tomiyama, T. and B.R. Meijer, *Directions of Next Generation Product Development.* Advances in Design, 2005: p. 27–35.

[4] Tomiyama, T., et al., *Complexity of Multi-Disciplinary Design.* CIRP Annals-Manufacturing Technology, 2007. **56**(1): p. 185-188.

[5] van Beek, T. and T. Tomiyama, *Connecting Views in Mechatronic Systems Design, a Function Modeling Approach*, in *MESA08*. 2008, IEEE and ASME: Beijing. Accepted for Conference.

[6] Van Beek, T. and T. Tomiyama, *Requirements for Complex Systems Modeling*, in *CIRP Design Conference 2008*. 2008, Springer: Enschede, The Netherlands.

[7] Umeda, Y. and T. Tomiyama, *FBS Modeling: Modeling scheme of function for conceptual design*, in *Proc. of the 9th Int. Workshop on Qualitative Reasoning*. 1995. p. 11-19.

[8] Lindemann, U. and M. Maurer. *Facing Multi-Domain Complexity in Product Development*. in *The future of product development, Proceedings of the17th CIRP Design Conference*. 2007. Berlin: Springer-Verlag.

[9] Bonnema, G.M., *FunKey Architecting, An Integrated Approach to System Architecting Using Functions, Key Drivers and System Budgets*. 2008, University of Twente: Enschede, The Netherlands.

[10] Boersting, P., et al., *The Relationship between Functions and Requirements for an Improved Detection of Component Linkages*, in *DESIGN 2008*. 2008: Dubrovnik, Croatia. p. 309 - 316.

[11] Muller, G., *CAFCR: A Multi-view Method for Embedded Systems Architecting*. 2004, Delft University of Technology: Delft, The Netherlands.

[12] Maletz, M., et al. *A Holistic Approach for Integrated Requirements Modeling in the Product Development Process*. in *The future of product development, Proceedings of the17th CIRP Design Conference*. 2007. Berlin: Springer-Verlag.

[13] Bonnema, G.M., *Use of models in conceptual design.* Journal of Engineering Design, 2006. **17**(6): p. 549-562.

[14] Yoshioka, M., et al., *Physical concept ontology for the knowledge intensive engineering framework.* Advanced Engineering Informatics, 2004. **18**(2): p. 95-113.

[15] Umeda, Y., et al., *Function, Behaviour and Structure, Application of Artificial Intelligence in Engineering V, Vol 1: Design, JS Gero*. 1990, Computational Mechanics Publications, Boston.

[16] Anggreeni, I. and M.v.d. Voort, *Classifying Scenarios in a Product Design Process: a study towards semi-automated scenario generation*, in *CIRP Design Conference 2008*. 2008, Springer: Enschede, The Netherlands.

[17] Albers, A., et al., *A Modularization Method in the Early Phase of Product Development*, in *DESIGN 2008*. 2008: Dubrovnik, Croatia.

[18] Browing, R., *Applying the Design Structure Matrix to System Decomposition and Integration Problems: A Review and New Directions.* IEEE transition on Engineering Management, 2001. **48**(3): p. 292-306.

[19] Stone, R.B., K.L. Wood, and R.H. Crawford, *A Heuristic Method for Identifying Modules for Product Architectures.* Design Studies, 2000. **21**(1): p. 3-31.