Proceedings of the WDK Workshop on
Product Structuring
June 22-23 1995
Delft University of Technology

Designing Mechatronic Product Families

F.J. Erens[1, 2], K. Verhulst[3]

1. Introduction

Mechatronic products are complex, not only because of the use of advanced technologies, but in the first place because of the combination of diverse technologies in one system. A good example of a mechatronic product is a medical x-ray system. This example will be used throughout this paper. For medical equipment, functions to control the position of the patient, the x-ray source and often several x-ray detectors on precise locations are to be combined with functions that control the optimal functioning of an x-ray tube, an image intensifier, a film transport mechanism or video chains. Evermore, requirements are set to increase usability: user control has to reflect more the medical use than the technologies used.

Systems with an eminent quality level have to be developed and produced in an ever-shorter time and at an ever-lower cost level. This results in a price pressure leading to more integration, which is equivalent to realising more functions with fewer technologies. The increasing price pressure means also development of more systems in a shorter time [Pine, 1993]. This can be realised by designing product families with the possibility to divert variants.

The combination of technology integration and family design leads to a more complex design process. The goal of this paper is making the complexity of design more manageable. To realise this, criteria are formulated to support design decisions. It is argued that design decisions are taken from different viewpoints and that the settlement of good design information is essential for decision taking. A phased approach to development, of which an example is given in Figure 1, is not sufficient to support design decisions. Phased approaches [Pahl, 1984] describe the interaction of business functions in the development process, however without paying attention to the quality of the design (i.e. the quality of the product descriptions) at each milestone of the development process. Therefore, this paper proposes a "design cycle" that considers interactions between this development process and product descriptions.

	PHASE 1	PHASE 2	PHASE 3	PHASE 4	PHASE 5	
	FEASIBILITY	OVERALL DESIGN	DETAIL DESIGN	INTEGRATION - TEST	START PRODUCTION	

Figure 1. Example of design phases

This paper first addresses different types of product descriptions. Then, a design cycle containing four elementary design steps is discussed. These four steps are executed repeatedly in the design process, not only for systems, but also for other levels in the product hierarchy as sub-systems and components. The interactions of design steps and product descriptions will be discussed for the different project phases. Furthermore, the combination of the proposed design cycle and the product descriptions gives the opportunity to discuss issues as modularity, integration, concurrency and product variety.

[1] KPMG Industrial Consulting

[2] Eindhoven University of Technology

[3] Philips Medical Systems

2. Product descriptions

In the development process, many different product descriptions can be recognised, although most product descriptions can be classified in three categories [Albano, 1992][Andreasen, 1987]. Product models that act as a backbone for the combined product information represent these categories. They are used by different business functions and in several phases of design to settle product information in a structured way, including hierarchical and non-hierarchical dependencies:

- The *Functional Model* is a consistent description of the functionality of a medical system. It is strongly related to the purpose of the product. Usually, the Requirements Specification, created by product management, is an important input for this model.

- The *Technology Model* is a consistent description of the application of technologies to ensure the operation of the product. Development creates most of the information structured in this model.

- The *Physical Model* is a consistent description of the physical realisation of a system. It is strongly related to the construction of the product. Manufacturing sets conditions for this realisation in order to guarantee an easy assembly operation without compromising the quality level or cost level.

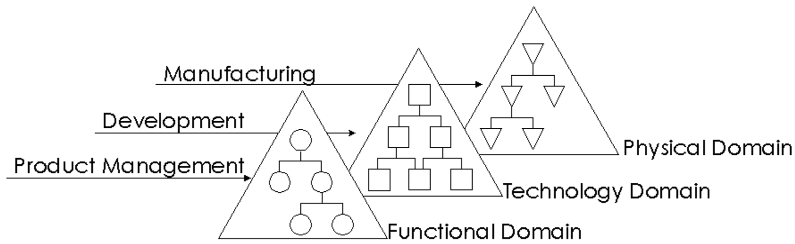

Figure 2. Product Models

Figure 2 depicts the contribution of product management, development and manufacturing to respectively the Functional Model, Technology Model and Physical Model. The depicted product models are independent of the hierarchical level of the product: these models can represent systems, sub-systems as well as components.

2.1. The Functional Model

A consistent description of the set of functions of a system is given in the Functional Model. It comprises the functions and the interfaces between functions. The functional requirements are primarily listed in the Requirements Specification (RS) in a textual form. In a second stage they are converted into a more formal description. This is to establish dependencies in an explicit form. A structured analysis checks for completeness and consistency [Hatley and Pirbhai, 1987]. This analysis of functional requirements has to separate functionality from technology. In the analysis, functionality is described in a hierarchical decomposition of functions to cope with complexity (see Figure 3).

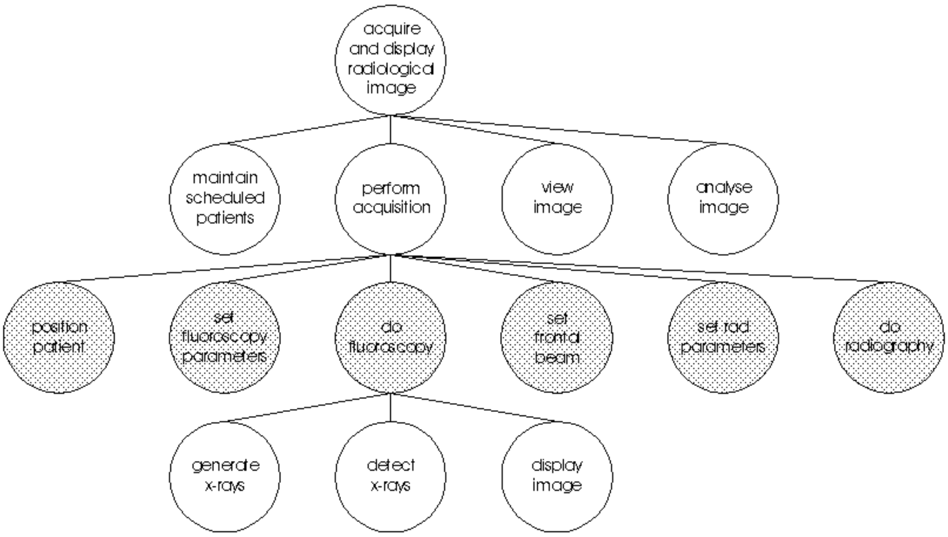

Figure 3. Functional decomposition

The Functional Model represents the intention of the requirements and guarantees an unambiguous realisation in the design process. Each level describes the functions and their dependencies and interactions that are relevant on that level. Design decisions are taken on one level of abstraction at a time. Figure 4 gives an example of one such a level, *perform acquisition*, in the functional decomposition of a medical system. Functions are depicted as bubbles. Data dependencies between functions are depicted with arrows.

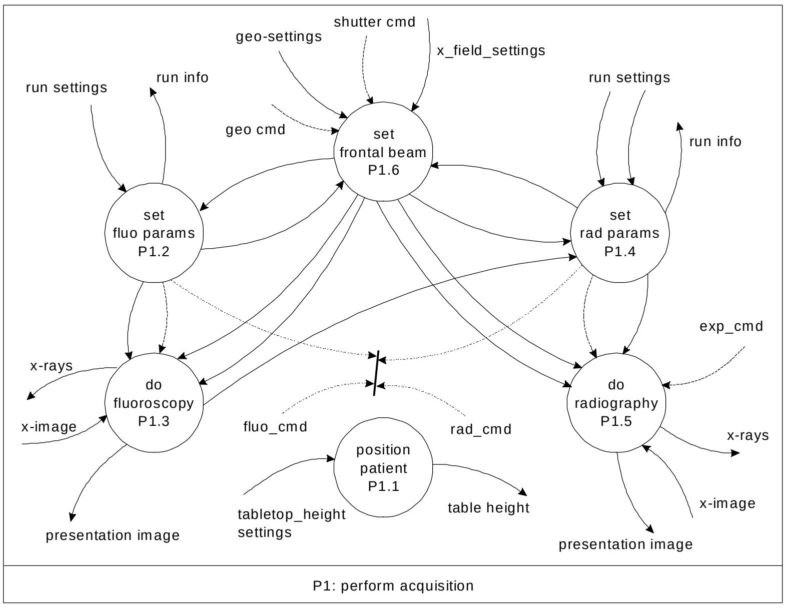

Figure 4. Functional architecture

2.2. The Technology Model

A consistent description of all technologies that are applied in a system is named the Technology Model. Again, it comprises the decomposition of technology modules and the interfaces between technology modules. A block diagram describes the technological decomposition of the system and assists the organisation of development projects. A simplified example of the technology decomposition is given in Figure 5. User functions are realised in a combination of modules.

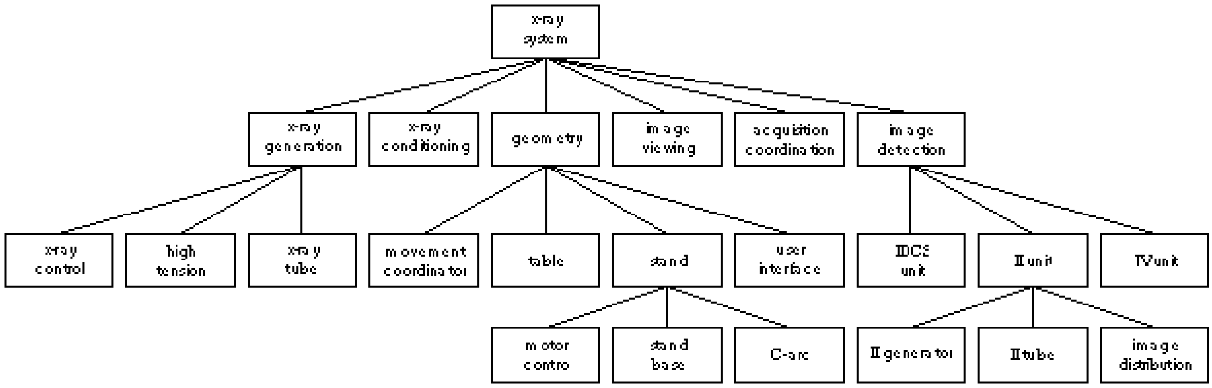

Figure 5. Technology decomposition

Similar to the design of the Functional Model, decisions with respect to the Technology Model are taken on one abstraction level at a time. The technology architecture describes the operation of a system by its modules and interfaces between those modules. In Figure 6, a technology architecture is given for the "x-ray system" on level one of the technology decomposition of Figure 5.

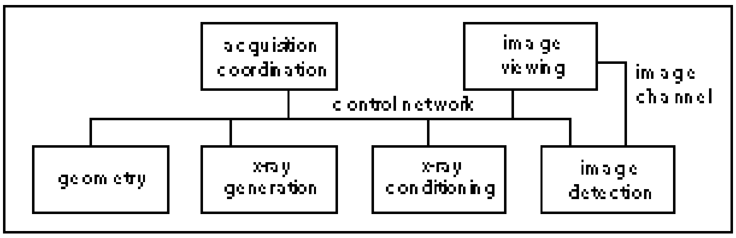

Figure 6. Technology architecture

The technology architecture consists of several modules and their interfaces. Modules are physical or logical units in which one or more functions of the Functional Model are realised, thereby taking into account constraints that have been formulated in the RS (e.g. constraints on the reuse of technologies). Examples of modules are a processor, a display, a video-board, a motor, etc. An interface is a specification of the interaction between modules. This can be a computer bus, an optical path or a mechanical connection. Interfaces in the technology domain support the functional interfaces as defined in the Functional Model.

Development is not the only user of the Technology Model. Also service makes use of the Technology Model for diagnostics of systems. Furthermore, service is involved in the innovation process in order to improve the maintainability of systems in the field. Requirements are specified to allow fault diagnosis using the Technology Model. Eventually, these faults have to be related to parts of the Physical Model.

2.3. The Physical Model

The consistent description of a system's parts and assemblies is named the Physical Model. The modules of the Technology Model are physically implemented in assemblies that can be manufactured, planned and serviced. Design decisions in the Physical Model are taken on one abstraction level at a time. The physical architecture is described in assembly drawings.

The fact that medical equipment is manufactured with some volume is the reason for a difference between an ideal Physical Model for manufacturing and the "Development Test Model" that is created in development. Eventually, the Physical Model is a compromise between the requirements of development, manufacturing, service and other parties that have an interest in the physical implementation of the product. Figure 7 gives an example of the physical decomposition, while Figure 8 positions the parts of the stand in the physical architecture.

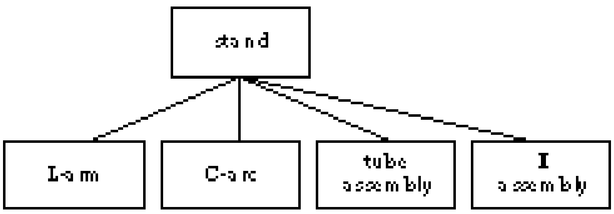

Figure 7. Physical decomposition

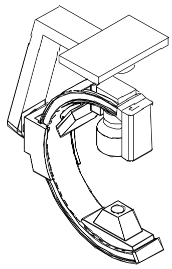

Figure 8. Physical architecture

In the production process, a bill-of-material is one of the representations of the Physical Model. For systems with such a large variety that the variants must be assembled to customer-order, the *generic product structuring* (GPS) concept is the most efficient product modelling language to describe the product family completely from the purchased components to the end-products [Erens, 1992] [Hegge, 1995][Wortmann, 1995]. Such a generic product structure makes it possible to generate a customer-order specific bill-of-material (and other documentation) out of the family description. This enables an efficient and controlled final-assembly process. The generated customer-order specific bill-of-material can also be used for "configuration management" purposes.

3. Elements of the development process

The purpose of the development process is to create a product description of sufficient quality level that can be unambiguously interpreted in the operational processes of sales, logistics, manufacturing and service. The Physical Model, which was introduced in the preceding chapter, is an important product description for these operational processes. The Technology Model is especially important as a bridge between the Functional Model and the Physical Model as it explains the operation of the system.

This chapter defines a design cycle which is similar to the productive reasoning model of March [1984] and Cross [1989]. In this cycle, four design steps are executed repeatedly. These four steps are summarised in Figure 9 for the Functional Model and Technology Model. Together, these 4 steps can be regarded as a quality cycle:

- *Decomposition* is adding detail to a product model. On the highest level, the three models describe the same product. However, on that level, product descriptions are still very generic and cannot be discriminated. Furthermore, it is difficult to map the functions of the Functional Model onto the modules of the Technology Model. For that reason, these functions are detailed until they can be allocated;

- *Allocation* is the creation of relationships between elements of different product models. For example, several functions of the Functional Model are allocated to a module of the Technology Model. In the same way, several modules are physically realised in an assembly of the Physical Model;

- *Composition* is combining elements of a product model, for example modules of the Technology Model or assemblies of the Physical Model. In a bottom-up design process, this is the first step;

- *Validation* is a check on the realised quality level of the product model, by relating it to a previous product model. For example, a composed module is validated against the original function, while an assembly is validated against the original technology module.

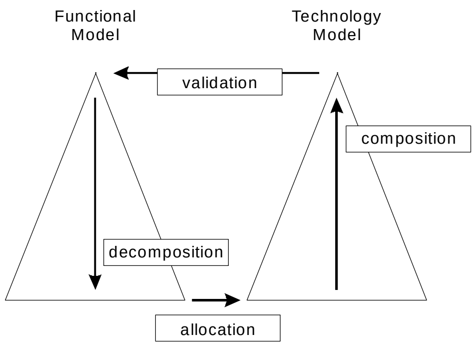

Figure 9. Four elementary design steps

The four design steps suggest a sequential approach to development. However, in reality there will be substantial informal communication between all domains to come to a solution that is acceptable from a physical, a technological and a functional perspective. For mature products or for designs which require much reuse, many modules in the technology domain or even assemblies in the physical domain will have been defined in the Requirements Specification (RS) prior to the decomposition of functional requirements.

The four elementary design steps can be applied several times, for different product models and on different abstraction levels of these models. The next chapter demonstrates that these steps can be applied for the design of both systems and sub-systems.

4. The Method

The development of a new product family starts with formulating the requirements. These requirements are described in the RS. Although there is only one RS, it covers the following three parts:

- specifications which set conditions for the functionality, e.g. requirements for the user interface;
- specifications which set conditions for the technology, e.g. requirements for reuse and reliability;
- specifications which set conditions for the physical realisation, e.g. requirements for packing and maximum size.

Functional requirements are considered to be *primary conditions*, while technology and physical requirements are considered to be *boundary conditions*. Although there are practical differences, there are no principal differences between these conditions.

Figure 10 demonstrates that a part of the RS is mapped onto the Functional Model, a part onto the Technology Model, and a part onto the Physical Model. These product models formalise the requirements and are especially used internally in the product creation process. However, some information is derived from the product models for external use, for example for commercial catalogues, user manuals and service manuals. In Figure 10, the grey areas represent the necessary decomposition activities before functions can be allocated to modules, and modules can be allocated to assemblies.

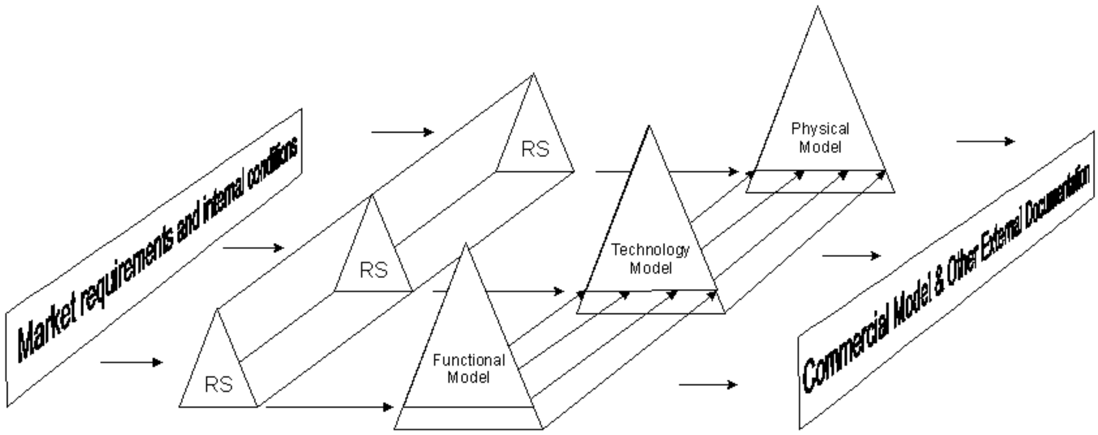

Figure 10. Design process

The first step in the design process is to compose product models from the requirements. Functions are structured in groups to reflect their dependencies. The functional requirements are made consistent, complete and unambiguous in the Functional Model.

The Technology Model is determined first of all by considering the boundary conditions that are specified in the RS. In a first step, a coarse structure with modules is made, which is based on the organisation's key technologies and states conditions for the allocation of functions. In a similar way, a first Physical Model can be composed from the requirements.

After the first creation of different product models, functions are allocated to the Technology Model. However, functions that are too big for allocation onto a module of the Technology Model are split into functions that are small enough for allocation. Then, the modules are composed to validate the technological solution with respect to the required functionality. The validation itself can be supported with simulation tools.

The process of detailing the Functional Model, allocating functions onto technology modules, and detailing the Technology Model is done in an iterative way. This iterative process can take place on different levels of the product models, for example on system level, but also on sub-system level. The allocation on system level results in combinations of modules and functions. Each of these combinations can be regarded as the design context for a next abstraction level, namely the design of a sub-system.

The iterative process of detailing the Functional Model, allocating functions, composing the Technology Model and validating this solution with respect to the required functionality is repeated for the interaction between the Technology Model and the Physical Model. Modules are implemented as physical building blocks, which are then composed into assemblies. Constructing an engineering prototype that is validated against the chosen solution concepts and the required functionality tests the resulting Physical Model. This validation could happen in phase four of Figure 1.

5. Observations

In this chapter several observations with respect to the current design process are discussed. These observations concern: (1) concurrent versus sequential design, (2) Overall Design versus Detail Design, (3) modularity versus integration, (3) single products versus product families, and (5) few technologies versus many technologies.

5.1. Concurrent versus sequential design

Recently, much attention has been paid to concurrent engineering as a means to reduce the throughput-time in development. There are two distinct ways to apply concurrent engineering in a project: (1) concurrent engineering in the small and (2) concurrent engineering in the large [Eppinger, 1994]. The first type of concurrency focuses on multi-disciplinary teams of which its members together evaluate the decisions that should be taken. This type of concurrency is sufficient for smaller projects in which the quality of the disciplines is the main determinant for the quality of the product design.

However, for larger development projects, a more formal approach to team co-operation should be undertaken as the quality of the product design is highly dependent on the quality of the project management. The three product models together with the four elementary design steps suggest to which extent the design process should be organised in a more sequential or more concurrent way.

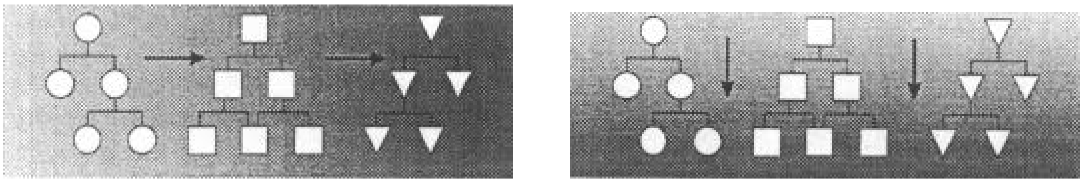

Figure 11. Sequential design versus concurrent design

Figure 11 shows that sequential design is characterised by first detailing the functional domain, after which the solution principles are determined in the technology domain and the solutions are materialised in the physical domain. In contrast, concurrent design is characterised by the involvement of all domains in all phases of the design process. Both approaches have positive and negative consequences.

The advantage of concurrent design lies in the early involvement and commitment of all disciplines in the early phases of the project. Furthermore, a concurrent design process can result in shorter throughput times than a sequential design process, however with the disadvantage of a high project risk when the decomposition strategies still have to take shape.

The advantage of sequential design is that quick progress can be made within one domain, however with the danger that a deep decomposition in the functional domain assumes certain solution concepts and can therefore not be realised in the technology domain. Similarly, this is also valid for decisions in the technology domain that affect the materialisation in the physical domain.

If the design process of medical equipment is considered, it can be noticed that this process is partially sequential and partially concurrent. In the feasibility phase, being the first step of the phased design approach (see Figure 1), all parties should be involved, in a very concurrent way, to judge the feasibility of the project with respect to functionality, technology and physical requirements. The second phase, named Overall Design, is used for detailing and allocating functions, and for composing and validating modules. This phase is sequential as can be seen in Figure 12. Detail Design is executed in the third phase. Here, in contrast, the implementation of technology modules in physical assemblies is executed concurrently. A fourth phase, integrating and testing building blocks, is a sequential activity.

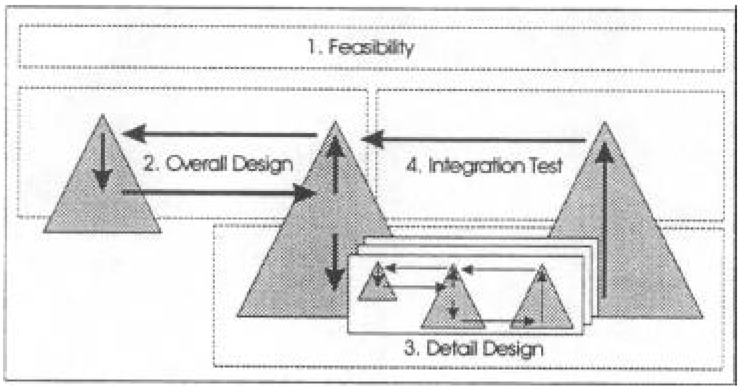

Figure 12. The phased design approach revisited

The design cycle can be executed for a system, but also for a sub-system or small device being part of a larger system. However, for several reasons, modules in a technology model are not always decomposed to such detailed level that they can be implemented as physical assemblies: they can be developed with co-designers who have their own responsibility, or they can be purchased from external suppliers. In those cases, the Detail Design phase on system level can comprise a complete external development of a part of the system. Again, this external development can have different phases with both concurrent and sequential activities.

5.2. Overall design versus Detail Design

The Overall Design phase describes those applied technologies in a system that are responsible for a correct operation of the system. The output of the Overall Design phase is a description of the system in terms of modules and interconnections. At the lowest level, modules are described that can be implemented in physical assemblies. The Detail Design phase defines the implementation of these assemblies.

The Overall Design phase is extremely important for several reasons. First of all, it determines to which extent functional requirements can be met within the technology boundary conditions. Secondly, it defines modules to be concurrently realised in the Detail Design phase. An important part of Overall Design is the systems architecture. This systems architecture is especially of importance for the development of external sub-systems and purchased parts. In those cases, the allocation of N functions onto one module creates new contexts for the design of sub-systems. An N : 1 allocation can be regarded as a synchronisation of viewpoints in the design process, after which the design of sub-systems can continue from a common perspective.

Because of the impact of the Overall Design phase, it is very significant to ensure the quality of Overall Design and to acknowledge the fact that the Overall Design phase is not completed until the solution has been validated. This often differs from today's manufacturing practice in which reviews are held after the allocation of functions, but prior to composition and validation.

5.3. Modularity versus integration

In general, a modular design is considered to be a design in which each module executes only one or a restricted number of functions [Ulrich 1990][Ulrich, 1991]. In a more integrated design, many functions are implemented in a module, for example to reduce redundancy (and costs) or to increase the system’s performance. Figure 13 shows that two functions are allocated onto a module and that two modules are physically realised in one assembly.

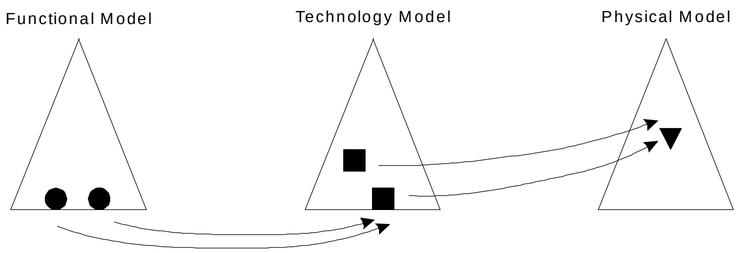

Figure 13. Integration of functions, modules and assemblies

In the Overall Design phase, it must be determined which functions are realised in a separate module. Possible reasons for modular realisation are creating variety or upgrading existing systems. Having control in the design process can be another reason for a modular product design. It is easier to manage uncomplicated relationships between functions, modules and assemblies than managing an integrated design.

In the design of mature systems, effective product decompositions in the three product models are known. Also the interactions between functions and technologies are better understood than for completely new designs. In these cases, it is possible and required to make a more fine-grained design. In the Functional Model, the decomposition becomes broader as more functions are analysed simultaneously. Then, these “fine” functions are allocated to “small” modules, thereby creating a more integrated design. This is only feasible if the interactions between functions and modules are well understood.

The above can also be reversed: when there is a need for a more integrated system, there is also a need to better manage the complexity of the development process. Because of the growing maturity and the corresponding performance and cost pressure of the medical equipment market, it can be expected that the need for integrated designs, and therefore the need for an appropriate Overall Design becomes more important for medical equipment development in the coming years.

5.4. Singular products versus product families

The most important thing in the design of a product family is that the architectures in the functional, technology and physical domain are defined such that the options that cater for the required customer variety fit this architecture. For example, a medical x-ray system uses approximately 100 sub-system variants to realise more than a million different systems. If these variants are part of a generic product structure, they can be selected on customer-order using the commercial catalogue as a representation of the Physical Model.

The abundance of variety does not only ask for a proper architecture, but also for the development of scaleable variants of functions, modules and assemblies. Scaleable variants make use of a subset of the architecture’s interfaces as can be seen in Figure 14.

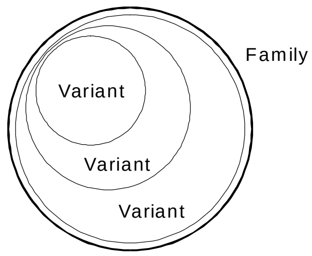

Figure 14. Scaleable variants

Scaleable variants ensure the testability of the system in phase 2 and phase 4 of Figure 1. Selecting the worst-case variants of all component families, i.e. by selecting all component variants that make maximum use of the system architecture’s interfaces creates the worst-case variant of the system family. The system test is only done for this worst-case variant, while all other component variants can be tested individually. A system family that is designed as a set of singular products requires more effort for testing and is usually of a lower quality level than a system that is designed as a product family.

5.5. Few technologies versus many technologies

The Technology Model is meant to bridge the functionality and the physical realisation of the system. Medical equipment that uses less different technologies or technologies that are better understood in their relationships can be designed without a dedicated functional analysis in the Functional Model. In these cases, it is possible to (1) directly allocate the functional requirements of the RS to modules of the Technology Model, and (2) to adapt the technology architecture to the structure of the functional requirements. A validation of the design can be done for the different technologies separately, for example for CAD-M and CAD-E, as the functions of the RS are closely related to these technologies.

The same seems to be valid for software, but the performance requirements of software can only be understood in relationship to the hardware platform (i.e. a different technology). In medical equipment, user functions require the combined operation of many technologies, e.g. x-radiation, image intensifying, video imaging, digital processing, software processing and software storage. An individual application of these technologies leads to a complex operation of the system, being more determined by the technologies than by the user functions. In those cases, a functional analysis has to be performed before sub-functions of the user functions can be allocated to the individual technologies.

6. Conclusions and recommendations

This paper tried to extend a phased development approach with three domains in which the design process is executed. These domains make use of product models to record product information. These models are (1) Functional Model, (2) Technology Model and (3) Physical Model. Each model has a specific relevance in design. The elements in these models are decomposed to add detail and composed to integrate partial solutions and to describe mutual relations.

Furthermore, the elements in a model are related with other models through allocation and validation steps. In this iterative design process, the allocation of functions onto modules is essential for synchronising the product models in such a way that the sub-systems can be designed relatively independently. Finally, this paper stated the importance of validation in the quality cycle of design. In other words, a design is not complete when functions have been allocated to modules and modules have been realised in assemblies. Generally spoken, the value of validation is yet not fully understood for earlier phases of the design process. Simulations provide powerful possibilities to validate the Technology Model W.R. to the Functional Model, i.e. to validate the Overall Design. In this way, expensive and time-consuming iterations in the design process can be avoided.

In the second part of this paper, the concept of using three product models and four elementary design steps has been used to discuss some important design questions. The following conclusions can be drawn:

- ❑ The design process cannot be described independent from the product descriptions. Therefore, design methods should unite product models and design phases.
- ❑ Concurrent and sequential design alternate in succeeding phases.
- ❑ The Overall Design phase is crucial for co-design and for purchasing larger parts of the system, especially before the Detail Design phase is commenced.
- ❑ As today's markets and products become more mature, it is important to design integrated products. This requires clear responsibilities of a system design function in development organisation.

- ❑ The product architecture of an integrated system should be such that modules fit this architecture. These modules cater for customer variety, for exchanging service parts and for upgrading a system with new features. From the viewpoint of testability, it is wise to make modules scaleable as scaleable modules have universal interfaces in predefined product architecture.

- ❑ The use of many technologies to realise user functions results in complex designs that should be supported by a structured design process.

In conclusion, in an ideal development organisation, the organisational structure supports the design method. This paper defined a design method that emphasises the role of System Design. When this method is embedded in the organisational structure of a company manufacturing complex mechatronic product families, a separate System Design function is required, being responsible for functional analysis and allocating functions onto technological modules.

Literature

Albana L.D.,Suh N.P.
Aximatic Approach to Structural Design, Research in Engineering Design, 1992, 4:171-183, Springer-Verlag Ndew York Inc.

Andreasen M.M., Hein L.
Integrated Product Development, IFS Publications Ltd / Springer Verlag London, 1987

Cross N.
Engineering design methods, John Wiley & Sons, 1989, ISBN 0 471 92215 3

Eppinger S.D., Whitney D.E., Smith R.p., Gebala D.A
A model-based method for organising tasks in product development, Research in Engineering Design (1994) 6: 1-13, Springer-Verlag London Limited, 1994

Erens F.J., Hegge H.M.H., Veen van E.A., Wortmann J.C.
Generative bills-of-material: an Overview, Integration in Production Management Systems, Pels H.J. and Wortmann J.C. (editors), Elsevier Science Publishers, IFIP, 1992

Hatley D.J., Pirbhai I.A.
Strategies for Real-Time System Specification, Dorset House Publishing, 1987, ISBN 0 932633 04 8

Hegge H.M.H.
Intelligent Product Family Descriptions for Business Applications, Thesis of Eindhoven University of Technology, ISBN 90-386-0491-2

March L.J.
The logic of design, In N. Cross (ed.), Developments in Design Methodology, ISBN 0 471 10248 2, John Wiley & Sons, 1984

Pahl G., Beitz W.
Engineering Design: a systematic approach, Springer Verlag, 1984

Pine B.J.
Mass Customization, The New Frontier in Business Competition, Harvard Business School Press, 1993, ISBN 0-87584-372-7

Suh Nam P.
The Principles of Design, Oxford University Press, 1990

Ulrich K.T., Seering W.P.
Function Sharing in Mechanical Design, Design Studies, Vol. 11, No. 4, 1990

Ulrich K., Tung K.
Fundamentals of Product Modularity, DE-Vol. 39, Issues in Design Manufacture Integration, ASME 1991

Wortmann J.C., Erens F.J.
Control of variety by generic product modelling, Proceedings of the First World Congress on Intelligent Manufacturing Processes & Systems, Puerto Rico, February 1995