INTERNATIONAL CONFERENCE ON ENGINEERING DESIGN, ICED'09
24 - 27 AUGUST 2009, STANFORD UNIVERSITY, STANFORD, CA, USA

THE NEED FOR FUNCTION PLATFORMS IN ENGINEER TO ORDER INDUSTRIES

Alex Alblas[1] and Hans Wortmann[1]
[1] Faculty of Management and Organization, University of Groningen, The Netherlands

ABSTRACT

Many industries base their innovations on product platforms. Platforms have predefined modularity with standardized interfaces. However, product platforms provide significant challenges to engineering industries that rely heavily on R&D, such as microlithography systems. Developing these systems depends on advanced technologies, that are still immature. Therefore, predefined modularity with standardized interfaces cannot be assumed. However, although platforms in the classical meaning cannot be achieved, this paper argues that *function platform*s offer a feasible alternative.

In this paper, a longitudinal case study is described at a large engineer-to-order firm delivering microlithography systems. The study analyses the platform concept in practice together with the emerging use of *function platform*s. It studies how function platforms are developed and used, and specifically what happens in the stages before the development of a physical platform; the phase where interim (functional) modules are defined. It considers the process of forming the architecture of the functional modules and examines the potential of reusing the functional clustering principles. The paper focuses on methods that model and exemplify function platforms.

This study broadens our understanding of function platform management. It exemplifies the dynamics of managing reuse and commonality in R&D–based engineering industries. The evidence from the case study shows that function platforms allow nuanced design freeze, since the platform describes the functional elements that will remain stable within the development of a product family. Therefore, function platforms enhance change management of the functionality foreseen. Also, defining an up-front function platform helps to sustain the functional decoupling of the physical product architecture. The paper finalizes with conclusions and implications for industry.

Keywords: Product platform, product model, design reuse, engineer-to-order

1 INTRODUCTION

Currently, an increasing number of industries is interested in the application of product platforms. Many examples can be found on product platforms in consumer goods industry [13] [30]. Platform strategies are often proposed as a way to reduce (internal) complexity in product development, manufacturing and maintenance, while offering variety to the market (e.g. [1], [2], [3], [5], [10], [12], [13], [14], [17], [27], [28]). In automotive industry, for instance, the platform is designed in order to deliver large product variety. In other industries, however, customers are powerful and their demands play a crucial role in the development of products, jeopardizing the idea of a platform. Although rapid progress of platform development in various industries has been made, engineer-to-order industry is still having problems in applying platforms: many applications are reported merely on products with a low level of complexity [9].

However, platforms are not a panacea. The study of Krishnan and Gupta [11] investigates the appropriateness of applying a platform strategy considering product and market conditions. Platforms could result in overdesign of low-end variants in a firm's product family when a platform is shared with high end products. A same phenomenon can be seen in overdesigned modules which can result in extra costs [11]. Accordingly, Hauser [7] found out that the platform based development approach could negatively affect profitability. Hence, product platform development also has shortcomings, especially when the number of products to leverage the extra design effort is low and when the designs change intensively. In general, several caveats in current platform strategies can be mentioned:

- *Reuse potential of platforms not fully used in engineer-to-order firms*. Many engineer-to-order firms design new products in a one-at-a-time pattern, ignoring the reuse potential of platforms [13].
- *Product platforms are not directly applicable in engineer-to-order firms*. Frequently variants are delivered through engineering. Platform theory needs to be accustomed to engineer-to-order industries producing complex products and systems [9].
- *Physical solutions are not known initially*. For mechatronic products for example, there is initially a functional architecture, but not yet a physical architecture [2]. However, companies can benefit from platform thinking by introducing early in the innovation cycle a mapping of functions to technologies (thus creating a function platform) despite of the fact that there is no grand theory joining all technologies together.

Engineer-to-order firms, producing complex products, have their own problems and opportunities in applying platform thinking. In the development of complex products a complete decoupling of subsystems is rarely feasible [31-33]. This is attributable to high levels of function sharing and geometric nesting of components to improve global performance of systems. As a consequence, performance requirements seem to conflict with decoupling of subsystems and contribute to system complexity [18]. For example a lithography machinery manufacturer is restricted by the interfaces and performance requirements of other systems in the whole chip production line. In this case, the technological principles are most of the time not known, the (functional) modules are defined, but the demanding performance criteria require often new solutions. As a consequence, physical platforms change over time and the commonality of parts is difficult to maintain due to changing solutions. A consequence of high system complexity is that the technological evolution of a platform is highly uncertain and difficult to define beforehand: solution principles, architectures and interfaces within the platform are continuously changing over time. Accordingly, it is in engineer-to-order industries difficult to determine whether or not a platform standard is affected, and consequently, whether a change implies a new platform or a platform extension. Still, in these industries producing complex products, platform development has much potential.

However, despite these caveats, most authors are convinced that platform development brings substantial benefits. Therefore, platform strategies need to be analyzed and developed further for engineer-to-order industry. This study investigates the applicability of platforms in engineering organizations that are driven by continuous stream of innovations rapidly implemented in customer solutions, in an R&D-based industry[1]. Although platforms in the classical meaning mostly cannot be achieved, this study argues that *function platform*s is a feasible alternative and postulates that:

- Function platforms *encourage reuse* of functional and technological product models.
- Innovations can be assessed in the light of the function platform strategy, i.e. is this functional change in line with the anticipated platform functionality.

The paper will first discuss the established view on platforms and its implications on engineer-to-order industries. Next, based on theory, a framework is presented that forms the conceptual foundations and frame of reference for the case study executed. After this, the research methodology is presented. Subsequently, the emerging use of the function platform concept is analyzed based on the research framework. The study examines how platforms are developed and used, and specifically what happens in the stages before the development of a physical platform; the phase where interim (functional) modules are defined. It investigates the process of forming the architecture of the functional modules and examines the potential of reusing these clustering principles. Finally, these findings will be discussed with the aim of generalization of using function platforms in engineer-to-order industries.

2 PRODUCT PLATFORMS IN ENGINEER-TO-ORDER INDUSTRIES

Many engineer-to-order firms design new products in a one-at-a-time pattern, ignoring the reuse potential of platforms. Meyer and Lehnerd [13] recognize the problems of focusing on single designs in product development. Project performance measures are based on single product calculations, without taking leverage of generic designs into account. Note that this single-product focus also can be

---

[1] R&D (science)- based industries are characterized by the short time lags between scientific discoveries and their industrial implementation [29]. Examples are pharmaceutical, biotechnology, semiconductor and fine chemical industries.

observed in the assessment of change proposals: changes are often assessed on individual technical product characteristics. Projects compete on individual designs against other projects in the corporation's portfolio, resulting in inefficiency in offering variety. *Product variety* can be defined on two dimensions: the breath of products that a firm offers at a given time and the rate at which a firm replaces existing products with new products [4]. Firms that are in between the extremes of the development of individual solutions and standardized solutions are having difficulties in using the commonality potential of product platforms. In this section the complexity of managing product variety is explained.

Architectures & platform management

Product platform development can be seen as an architectural concept [2] to reduce architectural complexity of a family of products by distinguishing the common base of the family from the elements that are subject to change. In line, Meyer and Lehnerd [13] define product platforms as a set of subsystems and interfaces that form a common structure from which a stream of derivative products can be efficiently developed and produced. Although a product platform is not the same as a modular architecture, modularity reduces the complexity to stabilize a common architecture because it isolates change within modules. A (theoretical) full modular architecture includes an one-to-one mapping from functional elements in the functional structure to the physical components of the product, and specifies de-coupled interfaces between components [18]. However, a modular product architecture approach could bound the architectural innovations (e.g. [6], [8]) which is a high risk for R&D-based industries. Also, function sharing and geometric nesting of components [18] [34] are design strategies to improve global performance which makes the product architecture less modular. So the nature of innovative changes and the product or platform architecture characteristics are intertwined with each other.

Highly innovative firms that deliberately have a broad decision freedom for customers to specify their products and allow continuous change of physical modules, require different platform strategies; platforms that capture the clustering of functional knowledge. For example in the development of microlithography systems which continually depend on highly advanced and yet undeveloped technologies, upfront predefined modularity with standardized interfaces is not realistic. An 'a priori' modular architecture is almost impossible because the tolerances of microlithography tools are much tighter than the ones for each key component [29]. In the development of end-products in microlithography the total performance specified cannot be assured in the system-design phase by simply qualifying the performance of each key component ex ante individually. Each system must be statically and dynamically fine-tuned by taking into account ex post idiosyncrasies of the individual components in actual use. Chuma [29] claims that ex ante modularity is customary only for industries with mature technologies, i.e. PCs, bicycles, package software, etc.

Complementary to the traditional ex ante modularity definition, Chuma [29] defines in his paper "interim modularity", that is an incomplete modular architecture developed during the early developmental stages. He describes the information panopticon effect, i.e. the more hierarchically visualized at a glance such a basic structure is, the more people start to discover the essence of those complex phenomena from the common perspective so that they can deeply understand the composition of the whole and part[2]. Such a basic structure can noticeably increase the visibility of complex phenomena or nonlinearities among various modules [35] and such a structure serves as an effective communication tool among the people involved. Suh [34] mentions the importance of decomposition in product development decisions and the positive effects of developing a hierarchy on decision making. He advocates that proficient use of decomposition is a prerequisite for design or organizational success. Hence, Chuma's interim modularity supports the idea of decomposing an overall design into a hierarchy. Note that in lithography the ex ante modularity is not defined beforehand and modules are defined to a degree that is possible with the available knowledge. Standardizing the product architecture in terms of standardizing component interfaces is not attainable in lithography systems. As a consequence, physical platforms are not stable but change over time. For this reason a new type of product platform is required that captures knowledge about interim modules and functions.

---

[2] This metaphor is based on a panopticon, i.e. a prison building a design concept that allows an observer to observe (-opticon) all (pan-) prisoners without the prisoners being able to tell whether they are being watched.

Functionality & Platforms

Hofer & Halman [9] propose a platform that captures knowledge about the arrangement of subsystems in the development of a family of products in order to restrict deliberately architectural choices for a market segment, i.e. the layout platform. In line with interim modularity they propose a way to structure design elements before defining or knowing the exact nature of the interfaces. Despite of the fact that the promising findings of previous authors show the potential of the platform layout concept on intangible elements, in industry there is still a high reuse potential before the development of a product layout, i.e. reuse of functions and the arrangement of functional elements to technologies.

Functions, concepts, technologies, etc. are developed in the start-up phases of development. Therefore, an interim architecture is needed to communicate preliminary concepts that define anticipated modules, especially in the startup phases of product development. Despite of its importance, the role of platforms in the functional and technology domain is not addressed in product development literature. This study attempts to fill a part of this gap by addressing the following research question: *What is the position of function platforms in engineer-to-order industries that engineer variants?*

3 CONCEPTUAL FRAMEWORK AND METHODOLOGY

In this section the conceptual framework is presented that forms the conceptual foundations for the case study conducted. Hereafter, the research methodology is presented.

Conceptual framework

To investigate the emergence of the function platform concept the data found in this study will be analyzed in the light of theory on domains [2] and lifecycle theory [21] [25]. In this section the focus will be on the development of the theoretical framework of the concept. In line, in previous studies the authors [24] claim that platforms, as separate artefacts of an enterprise, have their own lifecycle, next to the lifecycle of designs, products, projects, components, etc. Accordingly, function platforms have their own lifecycle as well. Wortmann et al. [25] developed principles that are needed to manage variety during the lifecycles of enterprise artefacts. Based on the work of Van den Hamer and Lepoeter [20], they compare product models with enterprise models and formulate five principles to manage variety of artefacts, namely:

- *View, representation and domain* where a representation is a comprehensive model in a single domain; a view is derived from a representation. According to Erens [2] a design can be characterized by several domains which are highly related, but exist separately from each other. Various authors distinguish the functional and the physical view on a product in engineering design (e.g. [2], [18-19], [34]). Erens [2] introduces, based on his studies in complex mechatronics development, the technology domain as a third domain that forms the transition between functions and physical realizations. The *technology domain* is a consistent description of the application of technological solution principles to ensure the operation of the system, independent from its final physical shape.
- *Progeny* which generalizes the version dimension. The progeny principle encompasses that new objects are created by inclusion or exclusion of other objects. Whereas lifecycles define the state transitions of the objects, progeny applies when new object are created resulting in new versions. Objects with previous versions may stop to exist or may encompass their own life cycle.
- *Life history and status;* the life history is represented by the evolution of the status. The *lifecycle* of an object describes its state transitions between initiation and end of life. State comprises attribute values of an object at a certain point in time, e.g. documents can evolve from initiation, for review, finalized and obsolete, driven by the completeness of the documents.
- *Hierarchy; mastering complexity at the level of types;* generalization versus specialization and aggregation versus decomposition. The ‘generalization and specialization’ dimension describes the range of objects that inherit the attributes on a specific level. The aggregation dimension covers the assembly of a collection of classes to form a new class. In contrast, decomposition is the division of a class in its component classes.

- *Mastering complexity by parameterization of instances;* the dimension of *variants*. It is important to note that variants differ from versions; variants are derived from generic objects by selecting between different selection attributes that define the object variant.

These principles form the basis for the theoretical framework that is used to explore the existence of and need for function platforms in practice. Next table depicts the conceptual framework.

Table 1 Conceptual framework

Domain	Functional	Technological	Physical
Life cycle	◄────	────────	────►
Progeny	◄────	────────	────►
Instances	◄────	────────	────►
Generalization	◄────	────────	────►
Hierarchy	◄────	────────	────►

◄────► = emergence of platforms in ETO

Research methodology

This study is based on a case study at a large industrial machinery manufacturer, in this paper depicted as MECHTECH, a developer of complex mechatronical products. The study first analyses the difficulties in the management of platform functionality. Its research design is based on a case study at a industrial machinery manufacturer in the lithography industry. The research project has been executed as a longitudinal case study [23] [26]. Such a research approach aims to study the phenomenon in its natural setting [22]. Because of the longitudinal character of the study, data is available from various data sources during an extended period of time. Secondly, data is of several process levels are compared to triangulate the results. During intensive site visits a large amount of minutes of meetings, procedures, design specifications, close-out reports and other miscellaneous reports where read and analyzed. Interviewees consisted of personnel involved in various engineering processes, e.g. system engineers, development engineers, supply chain engineers, production engineers and service engineers. Furthermore informal conversations were held and drawn up with a large variety of personnel. The researcher participated (passive) in relevant meetings related to product platform development. Finally, the results are validated with participants.

4 RESULTS: THE EMERGENGE OF THE FUNCTION PLATFORM

At the case company several attempts are carried out in managing the functional structure of products and platforms. A large improvement initiative at MECHTECH resulted in a new functional cluster (FC) and building block (BB) concept that stimulates reuse of functional and technology product models: the function platform. This section will briefly describe the basic principles of the function platform concept developed at MECHTECH. Note that we will briefly present the results of the improvement initiative and not zoom in on the process of forming the platform concept. For this purpose, the problems that form the basis for the need for function platforms are firstly discussed. After that, the platform concept is briefly explained.

Problems in the decomposition process

Before the introduction of the FC/BB concept at MECHTECH, the firm was experiencing difficulties in the transition of functions to modules. Changes were introduced on various aggregation levels of both the product functionality and the physical bills of material. As a consequence, it was difficult to control the commonality of systems especially in the early developmental stages, such as functional decomposition and component development. It was complicated to decide when to introduce a new component or when to reuse one. In general, the following sections describes the problems that form the basis for the introduction of a new FC/BB concept.

A common language to discuss engineering changes was not available

Physical production modules were defined on a high aggregation level comprising thousands of parts. Furthermore, the interface definitions were not available in the start-up phases of development, resulting in unpredictable effects of engineering change. Subsequently, concurrent development was difficult to realize because of varying views on product models. Manufacturing and service engineers discussed about requirements on physical installed base components and production modules instead of focusing on new developments in component functionality. Also engineering changes often described new solution principles instead of physical amendments. A common aggregation level and view was needed in order to manage requirements coming from various positions in and outside the organization.

Proper aggregation level to control modularity, commonality and reuse was not available

Even though project teams are organized based on component clusters, no method was in place to reuse and control this component cluster structure and to clearly assign responsibilities to teams organized by components. In contrast to production and service, product development was not organized with the goal to achieve modularity, commonality and reuse. Development was continuously focusing on improving, changing and adjusting component designs, whereas manufacturing was having difficulties to stabilize their parts and production modules. Although, in practice functions and technologies can be reused over product generations, a proper aggregation level that fits both downstream and upstream requirements was not available.

Generalizations of functions and technologies were not available

Development was organized in projects; with a clear start and end point. At the end of the project a product must be delivered according to the specifications. Hence, products are redesigned or reused according to project targets. As a consequence, commonality and reuse were not performance requirements of component teams. In effect, physical configurations are generic on a high aggregation level, but when looking at the components, small adjustments make the products one-of-a kind. Functions and technologies where treated on individual basis, without taking into account the generality of these functions and technologies needed to develop a series of products.

No proper freeze mechanism was available

Physical modularity and physical platforms help firms to stabilize physical configurations. However, sometimes it is difficult to freeze physical realizations because of performance requirements. New physical solutions for the same function are needed in order to attain performance. In this case the functions and technologies are stable and the physical solutions change over time. The strategic decision, which functionality and performance has to be achieved (together with the technologies needed to achieve it) in a particular development project is connected to the customer requiremens in a order. At some point in time a development manager needs to freeze parts of product in order to enable design reuse. At the heart of this decision lays the engineering strategy of a firm. Is the firm willing to give a customer complete specification freedom for a specific order or are specifications restricted to the configuration options of the available modules? In practice this decision is not this strict and firms, like MECHTECH, are in between customer-driven development and customer-independent development. Design freeze on technologies was not possible. Development managers had to choose between scylla and charibdys: either a design freeze on physical components (which often did not meet requirements) or no design freeze at all (which could bring a complete project out of control). Hence, functionality changes where proposed during the later stages of development. No guidelines where available to reject these changes resulting in high change costs.

The above mentioned problems triggered the development of a new platform concept described in the next sections.

The function platform concept

The follow section will describe the function platform concept.

View, representation and domain

Within MECHTECH functional clusters were set up with the aim to know, plan and control the development of functionality across systems. *Functional clusters* (FCs) can be seen as the coupling of functions in functional groups that is logical in terms of overlap, synergy and interfaces. Example of FC’s are wafer handling, positioning, sensing, control and infrastructures (e.g. mechanical, software, electronics). *Building blocks* (BBs) can be defined as a clustering of anticipated physical parts describing a sub function. The anticipated BBs are grouped in such a way that each FC performs a main machine-function. In order to keep a transparent structure, BBs can belong to only one FC. At MECHTECH this distinction is made by the FC/BB-structure. The BBs can be both generic and customer specific.

The basis for the scope and boundaries for the FCs within a platform is a platform system performance specification (i.e. Platform SPS), a document that specifies the design of the platform. Functional scope changes can be assessed in the light of the Platform SPS, i.e. the intended performance and functions of the platform. The platform and the BBs evolve from functional groupings to physical modules assembled by parts. By the introduction of the FC/BB concept MECHTECH clearly distinguishes between hierarchies of functions (FCs), hierarchies of technology modules (BBs) and production modules (PMs). Note that before the introduction of the FC/BB concept production modules where already in place. This distinction is represented in figure 1. At the left side of figure 1 the product functionality is broken down and allocated to the FCs. FCs are not physical units but rather a cluster of BBs that make up a product functionality and set of technologies so they utilize a "view" of the product that is most relevant to the development project needs; the functional and technology view. In this way the decisions 'what has to be achieved' and 'what is needed to achieve this' are clearly distinguished. The complexity lies in finding the right aggregation level of the right BBs that smoothly work together into FCs. When properly defined, new product development can be based upon controlled changes of the basic BBs in development rather than starting from scratch.

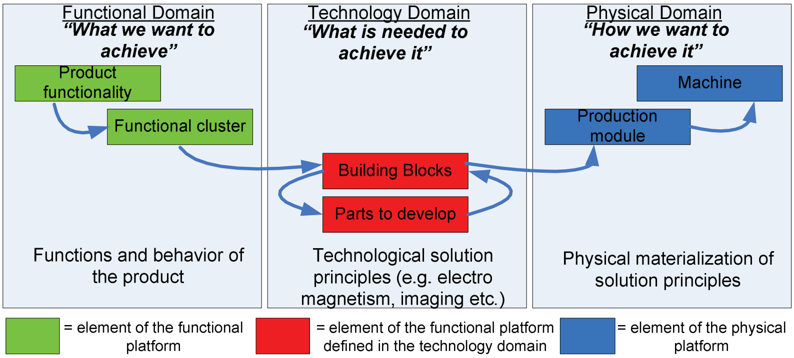

Figure 1 FC & BB structure

The BBs developed form the bases for transition of functions, via technologies, to physical realizations. In this way a production module owner can communicate his requirements to FC owners trough BBs. This was formerly done via parts or production modules resulting in the problems mentioned in section 4.1. The development organization is organized accordingly; a matrix of project leaders and FC owners. FC owners focus on modularity, commonality, reuse and technology road mapping and the project leaders focus on performance specifications. Function platforms, which is our conceptualization of the FC/BB concept, capture the knowledge about the decomposition of functions to sub functions and the allocation of those functions to technologies. During the functional decomposition of the overall system functions to sub functions, designers cluster the sub functions in perceivable modules based on past experience, i.e. not all options of functions to technologies are considered, but a subset (~cluster) is defined. Platforms are expected to capture this knowledge and to make it more explicit. Consequently, we define a function platform as the range of anticipated functions, the interfaces between the functions and mapping of those functions to technology clusters. Accordingly, function platforms can be defined as *the functional clustering principles and the generic functional and technology architecture*.

Hierarchy

The hierarchy of the function platform differs from the physical platform. Figure 2 illustrates this difference in the generic functional structure (GFS) and the generic production structure (GPS). The GFS, which forms the basis for the function platform, is the parent-child coupling of FCs and BBs.

Such a structure is based on previous developments and contains logical grouping of BBs that have overlapping or interfacing functions. So, a product can be fully configured out of a single level list of BBs, without any additional components. In the development of FCs the system engineers strive for BBs that can be independently, functionally qualified. Future BBs are grouped to form FCs that each perform a machine-function. The GPS is the structure that can be interpreted as the generic bills-of-material. Accordingly, the physical platform can be identified as the common subsystems and interfaces defined in the physical domain.

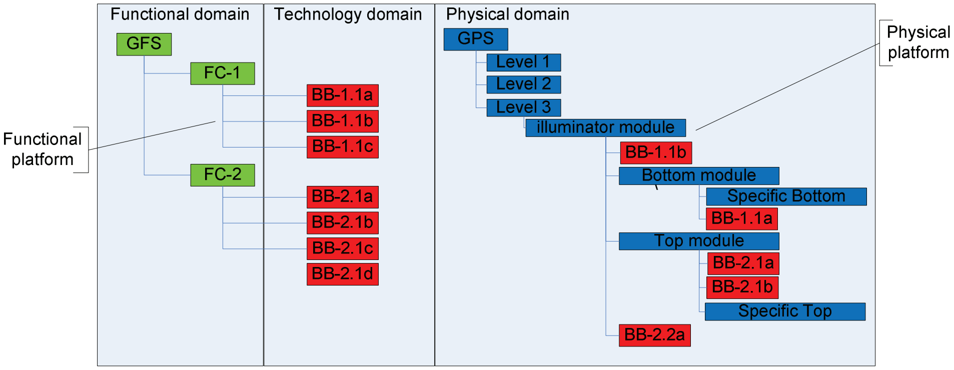

Figure 2 Generic Functional Structure and Generic Product Structure

Versions, variants and states

In the case of MECHTECH, several versions of FCs and BBs exist apart from the versioning of physical products and production modules. Hence, the progeny of the function platform at MECHTECH is captured in the versioning of FCs and BBs. Still, in MECHTECH it is unclear which version relates to which function platform. To manage the version dimension of the functionality and technology domain, the function platform must be handled as a separate artefact with its own progeny, lifecycle and variants. BBs are fully suitable for describing a machine configuration: the definition is such that any machine is a combination of BBs. Hence, the FC/BB concept will serve as the language used to describe machine configurations.

The FC roadmap process illustrates the way in which changes can be planned on BBs. The FC roadmap process is organized in order to plan changes on BBs. The process involves four elements, that is (1) Identify functional & technology requirements based on technology roadmaps (2) identify requirements and the resulting changes needed on the BBs, (3) plan changes and organize changes on BBs, (4) balance workload of BB changes over time. Consequently, BBs and FCs can have different states ranging from initiation, to frozen, to obsolete. In line, BBs and FCs can exist in different variants in parallel.

5 DISCUSSION

In general, defining a function platform can help the company in establishing an overall portfolio strategy. Engineering innovations can be assessed in the light of the function platform strategy, i.e. the question “is this functional change in line with the anticipated platform functionality?” can be answered rapidly. Note that the connection between the functions of a certain module and the range of functions anticipated by the company (i.e. the function platform), can disappear in the strategy of an individual engineer or project working on a module. So change in functions of modules need to be aligned with the function platform. In addition, in many engineer-to-order industries’ functionalities of product families evolve over time. Function platforms are intended to rationalize this decision making on the perceived functionality of platforms and support the ‘rules of the game’ on engineering change decision making on platforms. Once a change is proposed, engineers need to define whether platform standards are affected, i.e. is the platform extended or fundamentally changed resulting in a new platform. Without a function platform it is difficult to define the standard: is a product family defined functionally or physically? Which rules are adhered to with a common product platform?

The definition of a platform could differ due to the characteristics of the engineering organization; such as the extent to which engineering is independent of customer orders. In order-independent engineering with an established product architecture a product platform is defined in the physical domain, consisting of interface specifications and standardized production modules. In engineer-to-order organizations, this cannot be assumed, because configurations are engineered according to customer wishes. In these circumstances, a complete anarchy in changing designs is not an alternative. Still, it is crucial to define whether a change effects the formulated platform. Note that changes can vary from functionality changes to changes of used material. So, platform change management should be organized accordingly, based on the function platform.

5 CONCLUSION

This study presents the function platform concept that facilitates reuse of function and technology elements (FCs/BBs). Based on the function platform it is possible to generate engineering variants based on elements that are physically different and subject to engineering, but can be functionally and technologically reused. Furthermore, defining technology building blocks in the functional platform establishes a common language that enables including manufacturing and service requirements (i.e. commonality and reuse) with developmental requirements. Platforms stimulate design freeze, since within the development of a product family the platform describes the functional elements that are anticipated to remain stable. So, function platforms are expected to stimulate change management of the anticipated functionality. Although further steps need to be undertaken to manage the function platform as a separate entity, the FC/BB architecture makes management of progeny, lifecycle and complexity through hierarchies possible. Function platforms contain best practices in functional decompositions and lower possibilities of coupling between future physical modules. Although the study is limited by the number of cases observed, the results add substantially to our understanding of product platforms and indicates that (functional) platforms are applicable in engineer industries relying heavily on R&D.

REFERENCES

[1] Du, X.; Jiao, J. & Tseng, M. Architecture of Product Family: Fundamentals and Methodology, *Concurrent Engineering,* 2001, 9(4), 309.

[2] Erens, F. *The Synthesis of Variety*, Dissertation, Eindhoven University of Technology, 1996.

[3] Farrell, R. & Simpson, T. Product platform design to improve commonality in custom products, *Journal of Intelligent Manufacturing,* 2003, 14(6), 541-556.

[4] Fisher, M.; Ramdas, K. & Ulrich, K.. Component Sharing in the Management of Product Variety: A Study of Automotive Braking Systems, *Management Science,* 1999, 45(3), 297-315.

[5] Gonzalez-Zugasti, J.; Otto, K. & Baker, J. A Method for Architecting Product Platforms, *Research in Engineering Design* 2000, 12(2), 61-72.

[6] Halman, J.; Hofer, A. & van Vuuren, W. Platform-Driven Development of Product Families: Linking Theory with Practice, *Journal of Product Innovation Management,* 2003, 20(2), 149-162.

[7] Hauser, J. Strategic priorities in product development, Technical report*, Working paper*, 1999, MIT Sloan School, Cambridge, MA.

[8] Henderson, R. & Clark, K. Architectural Innovation: The Reconfiguration of Existing Product Technologies and the Failure of Established Firms, *Administrative Science Quarterly,* 1990, 35(1), 9-30.

[9] Hofer, A. & Halman, J. Complex products and systems: Potential from using layout platforms, *Artificial Intelligence for Engineering Design, Analysis and Manufacturing,* 2004, 18**,** 55-69.

[10] Jiao, J. & Tseng, M. Fundamentals of product family architecture, *Integrated Manufacturing Systems*, 2000, 11(7), 469-483.

[11] Krishnan, V. & Gupta, S. Appropriateness and Impact of Platform-Based Product Development, *Management Science*, 2001, 47(1), 52-68.

[12] McKay, A.; Erens, F. & Bloor, M. Relating product definition and product variety, *Research in Engineering Design*, 1996, 8(2), 63-80.
[13] Meyer, M. & Lehnerd, A. *The Power of Product Platforms: Building Value and Cost Leadership*, 1997 (Free Press).
[14] Moore, W.; Louviere, J. & Verma, R. Using Conjoint Analysis to Help Design Product Platforms, *Journal of Product Innovation Management*, 1999, 16(1), 27-39.
[15] Pikosz, P. & Malmqvist, J. A comparative study of engineering change management in three Swedish engineering companies, *Proceedings of DETC*, 1998.
[16] Robertson, D. & Ulrich, K. Planning for Product Platforms, *Sloan management review*, 1998, 39(4), 19.
[17] Simpson, T.; Maier, J. & Mistree, F. Product platform design: method and application, *Research in Engineering Design*, 2001, 13(1), 2-22.
[18] Ulrich, K. The role of product architecture in the manufacturing firm, *Research Policy*, 1995, 24(3), 419-440.
[19] Ulrich, K. & Eppinger, S. *Product design and development*, 1995 (McGraw-Hill New York).
[20] Van den Hamer, P. & Lepoeter, K. Managing design data: the five dimensions of CAD frameworks, configuration management and product data management, *Proceedings of the IEEE*, 1996, 84(1).
[21] Van de Ven, A.H. & Poole, M.S. Explaining development and change in organizations, *Academy of Management Review*, 1995, 20(3), 520.
[22] Voss, C.; Tsikriktsis, N. & Frohlich, M. Case research in operations management, *International Journal of Operations & Production Management*, 2002, 22(2), 195-219.
[23] Eisenhardt, K. Building Theories from Case Study Research, *The Academy of Management Review*, 1989, 14(4), 532-550.
[24] Wortmann, J. & Alblas, A., Product platform life cycles: a multiple case study, *International Journal of Technology Management,* 2008, 48(2), 188-201.
[25] Wortmann, J.C.; Hegge, H.M.H. & Goosenaerts, J.B.M. Understanding enterprise modeling from product modeling *Production Planning & Control*, 2001, 12(3), 234-244.
[26] Yin, R. *Case Study Research: Design and Methods*, 2003 (Sage Publications Inc.).
[27] Zandler, M.; DSouza, F.; Erens, F. & Verhulst, K. Architectures for product families, *Computers in Industry*, 1997, 33(2), 165-178.
[28] Zhang, W. & Britton, S. Managing modularity in product family design with functional modeling, *International Journal of Advantage Manufacturing Technology*, 2006, 30, 579-588.
[29] Chuma, H. Increasing complexity and limits of organization in the microlithography industry: implications for science-based industries, *Research Policy*, 2006*,* 35(3), 394-411.
[30] Sanderson, S. & Uzumeri, M. *Managing product families*, 1997 (Irwin).
[31] Simon, H. *Administrative Behavior: A Study of Decision-making Processes in Administrative Organization*, 1976 (Collier Macmillan).
[32] Hobday, M. Product complexity, innovation and industrial organisation, *Research policy*, 1997, 26(6), 689-710.
[33] Hobday, M. The project-based organisation: an ideal form for managing complex products and systems?, *Research Policy*, 2000, 29(7), 871-894.
[34] Suh, N. *The Principles of design*, 1990 (Oxford University Press).
[35] Watson, R.A., Pollack, J.A. Modular interdependency in complex dynamical systems. Artificial Life, 2005, 11(4), 445-457.

Contact: A. A. Alblas
University of Groningen
Faculty of Organisation and Management
Department Business & ICT
Nettelbosje 2
9747 AE Groningen
The Netherlands
a.a.alblas@rug.nl

Alex Alblas is a researcher at the Faculty of Organisation and Management within the University of Groningen (RuG). His research mainly aims at managing product diversity in complex and customer driven innovation processes. He is conducting research in close cooperation with several prominent industries, in particular in equipment manufacturing for semiconductor industries.

Hans Wortmann is a Full Professor in Information Management at the Faculty of Organization and Management within the University of Groningen (RuG). His special field of interest is in enterprise information systems. He is the Editor-in-Chief of Computers in Industry. Before joining RuG, he was employed at Baan Company, a vendor of standard enterprise software, as Vice President of R&D.