Linking Modularity and Cost:
A Methodology to Assess Cost Implications of Product Architecture Differences to Support Product Design

by

Sebastian K. Fixson

Diplom-Ingenieur, Mechanical Engineering
University of Karlsruhe, Karlsruhe, Germany, 1995

Submitted to the Engineering Systems Division
in Partial Fulfillment of the Requirements for the Degree of

Doctor of Philosophy in Technology, Management and Policy

at the
Massachusetts Institute of Technology

June 2002

Signature of Author ....
Technology, Management and Policy Program
Engineering Systems Division
April 23, 2002

Certified by ....
Thomas A. Roemer
Professor of Operations Management, Sloan School of Management
Thesis Supervisor

Certified by ....Joel P. Clark.........................
Joel P. Clark
Professor of Engineering Systems & Materials Science and Engineering
Engineering Systems Division

Certified by ....
Frank R. Field
Senior Research Engineer
Center for Technology, Policy, and Industrial Development

Certified by ....Susan R Helper.........................
Susan R. Helper
Professor of Economics, Weatherhead School of Management
Case Western Reserve University, Cleveland

Accepted by ....Daniel Hastings.........................
Daniel E. Hastings
Professor of Aeronautics and Astronautics and Engineering Systems
Director, Technology and Policy Program
Chairman, Committee of Graduate Studies

Linking Modularity and Cost:
A Methodology to Assess Cost Implications of Product Architecture Differences to Support Product Design

by

Sebastian K. Fixson

Submitted to the Engineering Systems Division
on April 23, 2002 in Partial Fulfillment of the Requirements for the Degree of
Doctor of Philosophy in Technology, Management and Policy

Abstract

Reaching saturation levels, many markets in modern industrial societies tend to fracture into smaller 'niche' markets, and create a need for greater variety. At the same time, increasing product variety in non-growing markets results in decreasing production volumes per model, which tends to increase costs.

Modularity as a design concept has been suggested to be able reconciling these opposing effects. Most descriptions of modularity characterize products through idealized extremes, such as 'modular' versus 'integral.' While conceptually powerful, this notion is very difficult to operationalize. Consequently, it has been very problematic to determine the economic consequences of modularity. This thesis presents a methodology to overcome this problem. The development of the methodology is split into three parts: what is modularity, what costs are considered, and how can the link between the two be established?

First, to operationalize modularity, an in-depth analysis of the phenomenon was conducted and an alternative framework developed. The multi-disciplinary analysis revealed that modularity is a bundle of product characteristics rather than an individual feature, and that different disciplines and viewpoints emphasize different elements of this bundle. Consequently, the descriptive product architecture framework developed in this thesis encompasses all dimensions identified in the analysis, but simultaneously enables one to comparatively measure those characteristics along individual dimensions.

Second, to improve the understanding of the multitude of costs that occur over a product's life, a product life-cycle view has been used to investigate the cost effects of early design decisions with respect to product architecture. In addition, a review of the cost modeling literature identified the gap that exists between some empirical work identifying particular product features' effects on particular costs, and the more general design guidelines such as design-for-manufacturing (DFM) or design-for-assembly (DFA).

Finally, the thesis constructed a link between modularity and cost by applying the product architecture framework and technical cost modeling to experimental case studies. Case study subjects were four different car door structures. The case studies demonstrate the cost consequences of individual product architecture dimensions by isolating their effects from competing explanations. Enabling the translation of business goals into focused design advice, the proposed methodology represents a tool to reconnect management and engineering worlds.

Thesis Supervisor: Thomas A. Roemer
Title: Professor of Operations Management

Acknowledgments

Although a dissertation's title page bears only one name, it never is the work of only one person. Many individuals have – knowingly or unknowingly – contributed to the successful completion of this dissertation.

I would like to thank my advisors for their guidance, support, and friendship throughout my years at MIT. First of all, I would like to thank Professor Joel Clark. By taking me on as a graduate student at the Materials Systems Laboratory (MSL), he provided the invaluable support required to make this dissertation project possible. His support stretched far beyond financial backing, encompassing intellectual debates, trust, and confidence. Interdisciplinary work is always endangered by having no real departmental home, but Professor Clark's encouragement to follow through with this work was instrumental for the design, progress, and completion of this project. In many hour-long discussions about 'modularity,' Professor Thomas Roemer provided instrumental help in shaping the arguments put forward in this thesis. Additional benefit of theses debates has been a growing friendship which I hope endures many other collaborations, in research and otherwise. I would like to thank Dr. Frank Field III, whose insights helped more than once to regain momentum when the project had to surmount another hurdle. In addition, his computer wizardry helped on multiple occasions to prevent true disaster. Despite being physically distant in Cleveland, Professor Susan Helper has contributed significantly to this work. Her perspectives and suggestions always challenged my thinking and were essential for making this dissertation truly interdisciplinary in building a bridge between the engineering and management worlds.

I would like to thank the International Motor Vehicle Program, and especially John Paul MacDuffie and Donna Carty, for partially funding this research and welcoming me to the IMVP community. Many people from industry have supported this work with data and experience.

My MIT experience has vastly surpassed my expectations. I believe that the people I met here have been the prime reason for my personal growth. At MSL, I had the opportunity to meet and work with dedicated, smart, and interesting people. Drs. Rich Roth and Randy Kirchain helped to provide an intellectually stimulating, respectful, and diverse work environment. All MSL students I had the pleasure to work with over the years, deserve my sincere thanks for making this lab the special place it has been for me. Two friends have made the often long working nights memorable. Patrick Steinemann's expertise for Swiss cheese fondues and component classification schemes filled many evenings, just as Francisco Veloso's for Portuguese wine and automotive supply chain structures. I would like to thank Ashish Kelkar and Alex Cosquer for their friendship and Squash challenges. The TPP community provided inspiration and support – thank you, Gail Hickey and Sydney Miller.

A special thanks goes to my fellow doctoral TMP comrades. Their diverse research projects opened new views onto this world for me; views I would have never had had I been in a single department. And their friendships have been a constant source of encouragement and joy. In particular, I would like to recognize Chizuru Aoki, Tina Ghosh (TMP h.c.), Carlos Martinez-Vela, and Terry Vendlinski.

I would like to thank my parents, who many years ago planted the seeds for this accomplishment by instilling trust in their children on the most fundamental level.

Finally, I would like to thank my fiancé Stefanie. She motivated me when I procrastinated, encouraged me when I lost focus, and always believed in me. She turned our years in Cambridge into an experience of personal growth, warmth, intellect, and love.

Table of Contents

Detailed Table of Contents

List of Figures

List of Tables

1. INTRODUCTION

1.1 Background and Motivation

With many markets reaching saturation levels in modern industrial societies, a trend towards greater market heterogeneity can be observed. Formerly large mass markets tend to fracture into smaller 'niche' markets. Product customization is one of the answers developed by companies to provide individualized products to many different customers. This development has been recognized for various products: from sneakers with the customer's own name imprinted on them to customized beauty products to personalized food where, for example, each customer picks the ingredients for his cereal (Mirapaul 2001). The automobile market shows increasing numbers of niches as well as increasing numbers of models in these niches, e.g., sports cars (The Economist Intelligence Unit 2000a, The Economist Intelligence Unit 2000b).

The market trend to demand more products in greater variety or products that are 'customizable' to a larger extent creates a problem for many companies. Individualization of products comes with a cost. Economies of scale are the key reason that mass-produced products can be produced at costs where many can afford to buy them. Craft production, the production theme prevalent before mass production, manufactured only individual products, but for costs that made it impossible for many to afford the products.

Several different, but closely related ideas have been put forward to overcome this apparent contradiction, centered on individualizing only the features the customer values when being different while keeping other parts standardized to achieve scale economies. Concepts like 'mass customization,' 'platform planning,' or 'build-to-order' all promote the re-use of some fraction of the product across product families or generations while 'customizing' the remaining fraction.

A product concept suggested to make these strategies possible is 'modularity.' Promoted with phrases like 'interchangeable components,' 'mix-and-match capabilities' or 'standardized interfaces,' the idealized opposite extremes have been named 'modular' versus 'integral.' While conceptually powerful and compelling in principle, the concept is very difficult to translate into real world products. For example, Lear, a company supplying interior products to automotive OEMs, bought United Technologies Automotive, a wiring specialist, in 1999. As a result, people speak about the 'interior module' as the new combined output (Anonymous 1999c), although it is rather a packaging of components. Does this mean process-ownership determines what a module is? On the other hand, the plastics industry has labeled a strategy to expand its markets by providing the components that are in the vehicle adjacent to those it already supplies, and consolidating them into fewer parts as 'modular marketing' (Anonymous 2000c). In the dichotomous framework of integral and modular architectures, however, this approach represents an integration process rather than modularization. Or, modularity is used as another term to describe 'subassemblies' (The Economist Intelligence Unit 2000f). But pre-assembled units do not necessarily guarantee the customizability of the end product.

One consequence of the confusion around the term 'modularity' then is that the viability of business strategies using modularity is also unclear. While modularity has been identified as beneficial for specific problems, like enabling faster product development (Thomke and Reinertsen 1998), or to allow the customer to (re-)configure his product according to his wishes (Pine 1993), there are contradictory arguments with respect to the potential success of some of the strategies associated with modularity. For example, while some promote the build-to-order strategy for the automotive industry to reduce the inventory that exists in the supply chain (Holweg and Pil 2001), others are highly critical of this idea because of its costs in the production stage, and alternatively have suggested moving the customization point into the distribution phase, i.e., locate-to-order (Agrawal et al. 2001). Yet others challenge the idea of 'mass customization' more fundamentally as being too expensive for many products and point out that customers are not always willing to pay for it (Zipkin 2001).

1.2 Research Question and Structure of the Dissertation

If *modularity* is understood as a product characteristic, it is a product characteristic that is established early on in the design and development stage, i.e., during concept or system design. At this early stage, the level of influence exercised by design decisions on downstream activities (production, logistics, use, etc.) is very high. Consequently, a better understanding of the effects of these early design decisions on the economics of the downstream operations has high leverage potential. What is needed is a method to provide the designers with a way to assess the effects of these early design decisions and ultimately to develop an understanding of the potential and cost of modularity. In fact, others have called for the development of advanced cost models to approach this task (Hervey 2000).

Therefore, the research question that this dissertation addresses is formulated as to develop a methodology that allows one to assess the cost consequences of modularity (Figure 1-1).

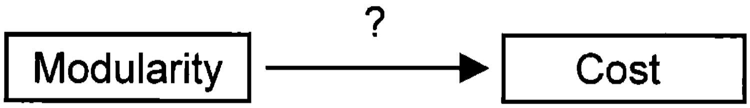


Figure 1-1: Research question: How to establish the link between modularity and cost?

In order to develop a methodology that allows one to assess the economic impact of modularity, the question is split into three portions. First, a deeper understanding of what actually constitutes *modularity* is absolutely necessary. Most current modularity definitions are either very product specific or too coarse to be operationalizable (Chapter 2 of this thesis). There is a need for a conceptual model that can bridge product classes and industries, and simultaneously provide measurable dimensions. In this thesis, I argue that the current ambiguity of *modularity* can be explained by viewing modularity as a bundle of product characteristics, and different disciplines and viewpoints emphasize different pieces of this bundle. To overcome this problem, I develop a descriptive product architecture framework that allows one to measure product characteristics along multiple dimensions (Chapter 3). Since the framework covers all features typically subsumed under modularity, it can serve as a broad practical description of the operational elements of modularity.

Second, the question of what costs are considered or incorporated in the analysis also requires much closer attention. To do this, I propose a product life-cycle view for the analysis and investigate the cost effects that early design decisions with respect to product architecture cause in product life stages after those decisions are made (Chapter 4). I also review the cost modeling literature and identify the gap that exists between some empirical work identifying particular product features effects on particular costs, and the more general design guidelines like design-for-manufacturing (DFM) or design-for-assembly (DFA) (Chapter 5).

Finally, constructing the link between modularity and cost is the third task of this dissertation. Building on the first two parts, i.e., the descriptive product architecture framework and the analysis of costing models and techniques, I approach the third part, i.e., constructing a link between modularity and costs, with help of experimental case studies (Chapters 6 & 7). The case studies allow one to demonstrate the cost consequences of individual product architecture dimensions by isolating their effects from competing explanations. I conclude with a discussion of the case findings, an interpretation of the results' meaning for the research approach and its practical applicability, and suggestions for future research (Chapter 8). Figure 1-2 shows the overall structure of the dissertation.

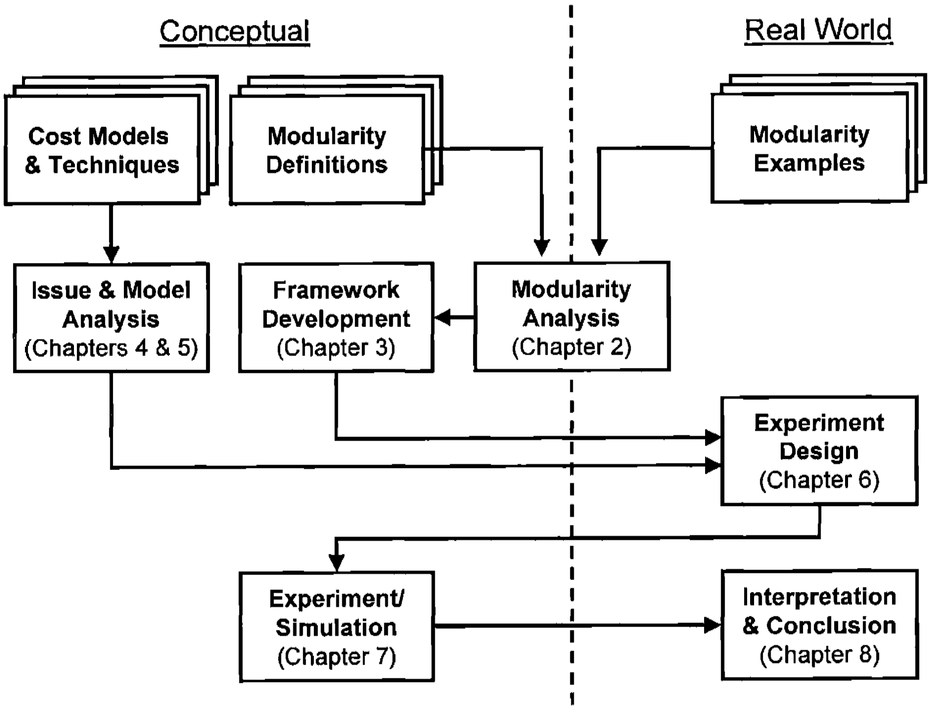


Figure 1-2: Structure of the dissertation

2. MODULARITY ANALYSIS

2.1 Chapter Introduction

Over the last decade, there has been a growing interest in *modularity* – both in academia and industry. For example, modularity has been described as enabling faster product development (Thomke and Reinertsen 1998) and allowing production of a large product variety at low cost (O'Grady 1999). Modularity is supposed to provide the customer with almost endless opportunities to customize his product (Pine 1993), and modularity has been identified as harnessing unparalleled innovation rates (Baldwin and Clark 2000).

Beyond the ubiquitous example of the personal computer, recent product examples that claim to be modular, range from small electronic devices to entire subsystems of the automobile. For example, Handspring designed its PDA (personal digital assistant) with a slot to fit in modules that turn the handheld device into an MP3 player, a camera, or a telephone (Biersdorfer 2001). In the automotive industry, cockpits (Anonymous 1999e) or front-ends (Anonymous 2001b) are to be delivered as modules.

But what exactly is *modularity*? Are there different levels of modularity? Can products be more or less modular? Does a product consisting of 'modules' exhibit 'modularity'? And what determines a 'module'? While this chapter does not claim to find the final answer to all these questions, it undertakes the task of analyzing the recent streams of literature on the topic of modularity and attempts to relate the various terminologies and interpretations of modularity to each other.[1] Figure 2-1 illustrates the role of this literature analysis in the context of the dissertation.

The literature - taken from engineering and management fields, published in academic journals or the trade press - offers many different definitions for modularity, often overlapping, yet slightly different. For example, some focus on technical function containment as the characteristic module feature; for others the option for the user to be able to re-configure the

---

[1] Sometimes interchangeably used, sometimes understood as a consequence of modularity, but in any case closely related are concepts like *mix-and-match, variety, (mass) customization, product platform,* and *product family.* This chapter's goal is not to write a handbook about all possible uses of these terms. It does attempt, however, to put these concepts in perspective to its main focus, i.e., modularity, whenever necessary.

modules, and thus the product, is the key point for modularity; and yet others emphasize complexity reduction during assembly as representative for modularity.

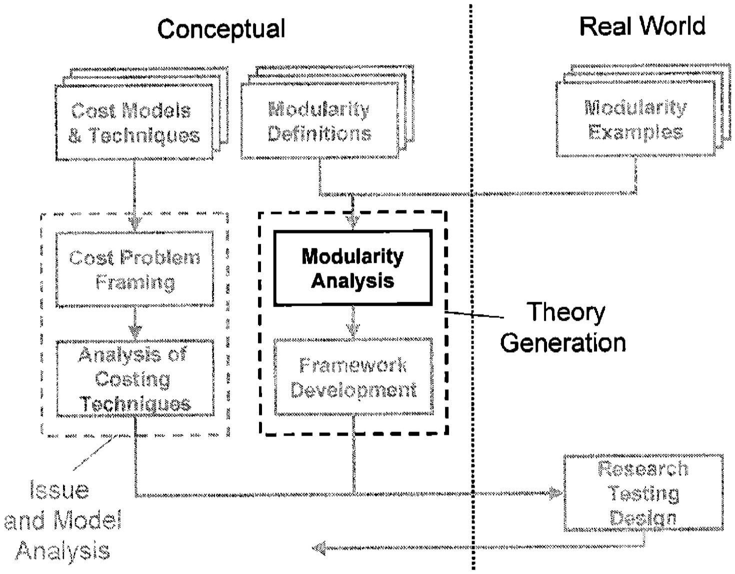


Figure 2-1: Relation of modularity analysis chapter for theory generation

To extract the essence of modularity, i.e., to find the common elements used across disciplines and to improve the understanding of the remaining differences, I develop a three-lens analysis tool to analyze the literature along the dimensions *system, hierarchy,* and *life cycle*. The three perspectives help to illustrate that *modularity* really is a bundle of product characteristics, and different views emphasize different pieces of this bundle. As a consequence, it may be advantageous to link the individual product characteristics themselves to the desirable (or undesirable) effects rather than employing an aggregated concept like *modularity*.

Two boundaries define the scope of this chapter. The first boundary defines the literature considered. Although it has been found that the concept of modularity (or parts of it) is used in disciplines as diverse as psychology, biology, American studies and mathematics (Schilling 2002 forthcoming), this analysis is focused on the literature bodies in engineering and management of technology. The stream of literature that applies modularity concepts to organizational designs and institutional structures often exhibits some overlap to product modularity and is considered where appropriate. Furthermore, although the emphasis of this chapter is placed on academic literature, some industry definitions of modularity are also included to consider the practitioners' viewpoints. While this literature analysis does not claim to be perfectly exhaustive, it analyzes a selection that I feel can be understood as representative for the current state of research on modularity. The second boundary defines the subject of analysis. This literature analysis is concerned with modularity concepts and ideas for industrially manufactured and assembled hardware products. While a number of similarities between hardware and software products exist, some fundamental differences do remain. For example, in software design it is possible to construct hierarchies that do not exist in the physical world. Therefore, this chapter restricts its analysis to hardware products.

The remainder of this chapter is organized as follows. The next section presents the analysis tool and explains its three perspectives. Sections three through five apply the three perspectives *system, hierarchy,* and *life cycle* to the literature body. The systems perspective discusses how differences in importance placed on elements (modules) and/or relations (interfaces) can result in different understandings of modularity. The hierarchy perspective investigates how the understanding of modularity depends on its development path, i.e., whether the modularity definition arose from a focus on the product technology, or whether it was achieved by decomposing a market driven, top-down approach. In the fifth section, the life cycle perspective explores how the different loci along the product's life cycle that individual researchers choose can affect the understanding of modularity. Finally, section six summarizes the findings and concludes with a discussion of possible interpretations and future research opportunities.

2.2 The Analysis tool: Three Perspectives on Modularity

This chapter analyzes works from different thought worlds and from different occupations. The thought worlds include engineering, management and operations management[2] and the occupations encompass academia and industry. Each of these worlds typically has its own way

---

[2] Of the engineering thought world, the majority of the references are taken from design engineering since *product modularity* is the focus of this chapter. The management literature reviewed is centered in the management of technology arena. Finally, I consider a small set of the operations research/operations management literature, which, although small, does cover some important assumptions for the understanding of modularity that are common in this field. The appendix presents all references in two lists: one for engineering literature and one for management literature (including operations management).

of looking at problems and definitions, as is the case for *modularity*. This thesis argues that herein lays one of the sources for the confusion that often accompanies the term *modularity*. For example, the more management-oriented literature often describes 'modularity' on a relatively abstract level as having standardized and interchangeable components: "modular design [is] a unit or group of standardized elements or parts that may be used within a number of different products" (Galsworth 1994, p.195), or "a modular system is composed of units (or modules) that are designed independently but still function as an integrated whole" (Baldwin and Clark 1997, p.86). In contrast, the literature based in the engineering world mostly focuses on developing designs or design guidelines for specific purposes. As a consequence, the resulting module definitions focus on particular applications, like product functions (Stone et al. 2000a), production requirements (Siddique et al. 1998), or material contents (Newcomb et al. 1998).

While these differences reflect to some extent the origins of the works (engineering works tend to focus on technical details, management articles on market relevant aspects), there are additional differences that can be found across and within these literature bodies. For example, while some sources focus on the subsystems' definition ('customer's choice to mix-and-match components'), others discuss in detail the interfaces ('must allow non-destructive separation'). Yet others consider both aspects important ('one-to-one mapping *and* decoupled interfaces').

Finally, more differences across and within the literature bodies are represented by the understanding of modularity with respect to the life cycle phase under consideration. For some, modularity allows the optimal execution of design tasks, for others the efficient organization of production or distribution, and for yet others it allows the customer to re-configure her product. Interestingly, these 'modularities' (and their accompanying modules) can be very different.

How can all these different viewpoints be reconciled? Is there a way to improve the coherence within and to bridge the gap between the different thought worlds? In other words, can the modularity descriptions from the management literature be operationalized and can the prescriptive models from the engineering literature be generalized?

If definitions and descriptions of modularity are made with various backgrounds and in various contexts, it seems worthwhile to use multiple perspectives to search for common elements and remaining differences. For this reason, a multi-dimensional analysis tool is proposed to distill the common aspects of modularity and to understand when and why additional, perspective-specific aspects occur. Three perspectives represent the lenses through which the often overlapping yet slightly different modularity descriptions can be investigated (Figure 2-2).

The first perspective views the product like a system. Analogous to a system, a product can be described via its elements and the relations between them. This view helps to clarify some of the underlying features of interchangeability. I call this view the systems perspective. The second perspective views modularity from opposing ends of the product development path. Modules based on functionality from a technical viewpoint can differ considerably from those defined from a market viewpoint. This view is named the hierarchy perspective. The third perspective investigates how the choice of one phase of the product life cycle over another can result in stressing some aspects of modularity while pushing others to the background. This third view is the life cycle perspective.

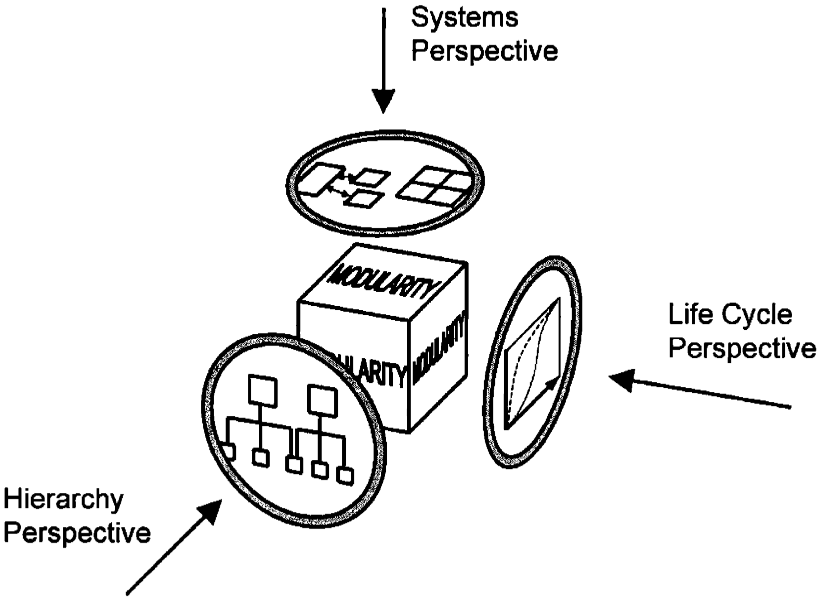


Figure 2-2: Three perspectives on modularity

In contrast to other literature reviews that cover an entire field of research (see, for example, Finger and Dixon (1989a, 1989b) for an extensive review of research in mechanical engineering design) the analysis presented here is rather guided by a phenomenon (modularity). Consequently, this analysis tool does not cluster the literature into groups but rather uses the perspectives of the tool as lenses through which it reviews the literature. Because more than one aspect might be considered in a single reference, some sources will be discussed from more than one perspective. The ultimate goal is to grasp the underlying assumptions and elements of product modularity that are used throughout the literature and to develop an understanding of how the different definitions and viewpoints relate to each other.

2.3 Systems Perspective: Do Modules determine Modularity?

Trying to capture what *modularity* is, or how various scholars and practitioners use the term leads quickly to the related question of what is a *module*. The lowest common denominator of most descriptions of modules is probably the notion that they exhibit relatively weak interdependencies between them and relatively strong interdependencies within them (e.g., Alexander 1964, Ulrich 1995, Baldwin and Clark 2000, Schilling 2000). The attempt to operationalize this notion, however, leads to a number of additional questions. For example, how are gradations of interdependencies between modules distinguished as representing different levels of modularity? And how are they compared to various levels of dependencies within modules? In addition, if modules are a precondition for modularity, are products with more modules more modular than products with fewer modules? Or, is the level of modularity also affected by the modules' own characteristics (size, function, etc.), and if so, are the averages or the extreme values of the characteristics establishing the whole product's level of modularity?

In the systems literature, a system is determined by (a) its elements and (b) the relations between these elements. Seen through this lens, there seem to be two fundamental dimensions which many product descriptions and analyses employ: (1) the elements the product consists of and (2) the relations (i.e., interfaces) between these elements. Researchers and practitioners have used and interpreted these dimensions in different ways, emphasizing different aspects. Three sets of this research are discussed below: the works which focus primarily on the relations a product exhibits between its elements, the works which concentrate on the elements themselves, and those works which combine both dimensions (Figure 2-3).

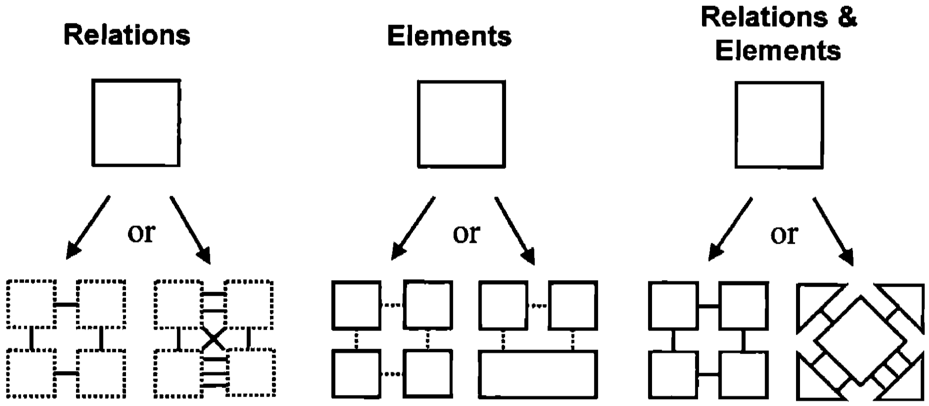


Figure 2-3: Systems perspective: three different foci for product decomposition[3]

2.3.1 Relations only (Interfaces)

In the first group, most researchers focus exclusively on the dimension *relations*, arguing that any determination of the elements (i.e., their content, functionality, size, location, etc.) unnecessarily constrains the analysis. Baldwin and Clark, for example, avoid the dimension *elements* altogether but rather focus on the implications of interface specification for the design process (2000).[4] Often, the dimension *element* is only implicitly addressed while the main focus is on the interface dimension: "Production of components conforming to standard interface specifications also leads to modularity." (Garud and Kumaraswamy 1995, p.94)[5] or "a *modular*

---

[3] As an abstraction, assume that the three boxes in the top row represent three cases of a product and the area of each box represents the product's functionality. For each of the three cases, the bottom row suggests two ways of decomposition into smaller elements. Elements are represented by boxes and interfaces by lines. The difference between the two decompositions in the first case affects only the interfaces (solid lines) and assumes identical elements (dashed boxes). Conversely, the decomposition in the second case neglects the interfaces (dashed) but focuses on the elements (solid) instead. The third case's decomposition considers both elements and interfaces.

[4] However, they implicitly introduce an upper bound for a module's complexity: "A complex system can be managed by dividing it up into smaller pieces and looking at each one separately. When the complexity of one of the elements crosses a certain threshold, that complexity can be isolated by defining a separate abstraction that has a simple interface. The abstraction hides the complexity of the element; the interface indicates how the element interacts with the larger system." (Baldwin and Clark 2000, p.64) While 'complexity' can have multiple dimensions, it is conceivable that there is some relation between a module's complexity and other measures for its role within the product, i.e., its size, its functionality, etc.

[5] Garud and Kumaraswamy add that a mix-and-match capability is at the root of their definition: "Modularity allows components to be produced separately and used interchangeably in different configurations without compromising system integrity." (Garud and Kumaraswamy 1995, p.94) While this implies that levels of functionality are somewhat defined for the components, nothing is said about the characteristics of this functionality.

product architecture [..] is a special form of product design that uses standardized interfaces between components to create a *flexible* product architecture" (Sanchez and Mahoney 1996, p.66, *italics* theirs). Standardized interfaces (for component exchange) have also been the centerpiece of Starr's concept of modular production: "It is the essence of the modular concept to design, develop, and produce those parts which can be combined in the maximum number of ways" (Starr 1965). This focus on interfaces with respect to modularity is particularly strong in assembly-dominated industries. For example, the automotive industry's heavy emphasis on modules' roles in assembly[6] results in a view that almost neglects the elements' (modules) role for the product function: "'Modules' are groups of components arranged in close physical proximity to each other within a vehicle, which are often assembled by the supplier and shipped to the VM [vehicle manufacturer] for installation in a vehicle as a unit. Modular instrument panels, cockpit modules and door modules are examples." (Delphi 1999).[7]

Common elements for the set of researchers focusing on the relations between elements (i.e., modules) are the notions of "standardized" and "interchangeable." While technical details still differ along other dimensions (e.g., interface design), these notions imply the existence of a certain number of alternatives for the elements with equal connection points.

2.3.2 Elements only

Researchers in the second set have placed their emphasis on the dimension *elements*. Often, this emphasis stems from the process of aligning the product's functions and requirements with its physical components (e.g., Erixon et al. 1996). On a conceptual level the idea of product decomposition seems straightforward, as Alexander quotes Plato: " ... the separation of the Idea into parts, by dividing it at the joints, as nature directs, not breaking any limb in half as a bad carver might." (in Alexander 1964, preface). To operationalize this concept, however, is much more difficult and researchers have chosen various approaches which can be clustered into three sub-groups. These sub-groups can be distinguished by the extent to which they consider

---

[6] Common in the auto industry, is a distinction between modules and systems. The latter focuses on product function, while the former is mostly associated with assembly (Mercer 1995).

[7] This definition is taken from Delphi's 1999 10k-report. Other automotive suppliers and OEMs use similar definitions.

architectural changes in the way functions are allocated to the product's elements (Figure 2-4). In the simple case, the elements' functional boundaries are fixed and only predetermined sub-units can be exchanged. The medium case introduces the option for the designer to 'collect' smaller elements into larger ones to 'form' modules. Finally, the fundamental case permits a complete re-allocation of functions to the elements. Each of these cases is discussed in turn.

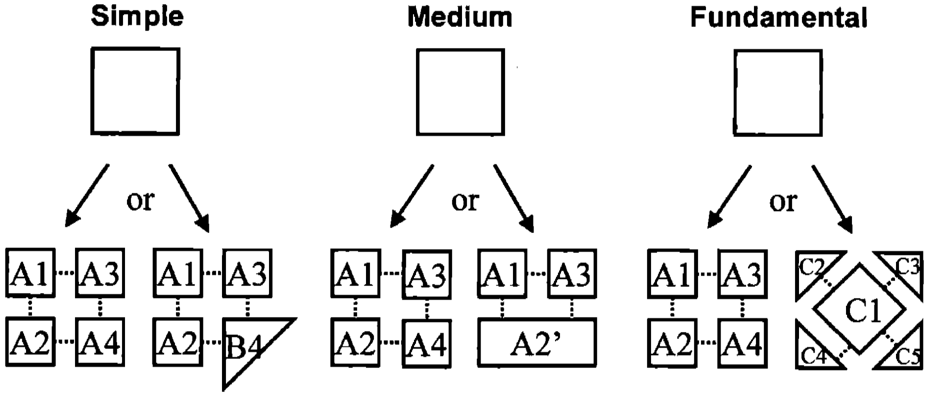


Figure 2-4: The dimension 'elements only' can vary in multiple ways[8]

2.3.2.1 'Simple' (Element replacement)

First, in its most simple case, the architecture is predetermined and only elements that contain a certain function or feature can be varied (A4 or B4, simple case in Figure 2-4). Examples are color changes of face-plates (e.g., on cell phones) or the use of different power sources in otherwise identical products (e.g., power tools). Coulter et al. also follow this idea to determine the optimal material choice for each component to achieve best recyclability of an automotive center console (Coulter et al. 1998). They apply an optimization approach that alters the materials for each component to minimize the number of different materials per pre-selected module (component group). Characteristic for these 'element replacements' is that they cannot differ to an extent that the product functionality is endangered, i.e., they must contain, or consistently contribute to, the function or feature that is to be changed (or varied).

---

[8] Again, assume that the area of the squares in the top row symbolizes product functionality, i.e., all three cases represent identical levels of functionality. Then the different ways of decomposition illustrate variations in the way the functionality is allocated to the product's elements (sub-units, components, chunks, modules, etc.).

This approach can also often be found in the operations management world. For instance, models developed to identify potential gains from parts commonality implicitly assume interfaces that guarantee total interchangeability of components or modules. Using this simplification, some models investigate the effect of parts commonality on safety stock levels (Collier 1982, Baker et al. 1986), how parts commonality affects supply chain costs (Ernst and Kamrad 2000), or how matching supply chain structure to variety type affects firm performance (Randall and Ulrich 2001)[9]. Other works assume components as interchangeable but allow them to differ along a performance dimension or quality to allow for creating product variety. For components that affect the product quality only weakly or indirectly[10], the analyses focus on balancing cost penalties from overdesign with cost savings from commonality. For example, Fisher et al. investigate the factors that determine the number of different brakes across a car family (Fisher et al. 1999), or Thonemann and Brandeau develop algorithms to find the optimal level of commonality for automotive wiring harnesses (Thonemann and Brandeau 2000). For components whose quality level does impact product quality, Desai et al. model how to balance the revenue and cost effects of commonality for the different quality levels of the components (Desai et al. 2001).[11]

Another research approach that fits into this subset of approaches is 'group technology.' It advocates "to exploit similarities and achieve efficiencies by grouping like problems" (Hyer and Wemmerlov 1984, p.4). Primarily focused on forming part families, this grouping is suggested along multiple dimensions such as design, material, manufacturing process planning & cell

---

[9] Randall and Ulrich distinguish two types of variety: production-dominant variety and mediation-dominant variety. In case of the former the increase of production costs associated with increased variety outweighs the increase in market mediation costs, in case of the latter vice versa. In either case, however, the variety is provided by a change in an attribute. Their case products, bicycles, have four attributes: frame material, frame geometry/size, frame color, and components. As a consequence, the product architecture does not change with an 'exchange' of an element with another element with a different attribute level.

[10] A component's quality affects the product quality only indirectly if the component quality level (above a certain threshold) does not differentiate the product from the customer's perspective.

[11] Fisher et al. (1999) categorize a product's components into two groups. One encompasses all components with a strong influence on product quality and the other includes all components with a weak influence on product quality. In their analysis Fisher et al. focus on the latter category to model cost trade-offs. Thonemann and Brandeau 2000 follow the same idea. In contrast, Desai et al. (2001) model explicitly the impact of quality differences on both cost and revenues. Even so, they also model the quality difference as confined to the element (component) itself, and assume perfect component interchangeability.

design, or purchasing criteria (Suresh and Kay 1998). From a product perspective, this also argues for interchangeable components.[12]

With respect to modularity, the 'simple' case of element exchange mirrors what has been termed parametric design, i.e., the product architecture is fixed and characteristics are varied only within the boundaries of the elements (e.g., material, quality, color, etc.). This approach almost always includes two assumptions: the exchanged elements provide identical interfaces and the replacement must not compromise system function.

2.3.2.2 'Medium' (Packaging problem)

The second sub-group of decomposition approaches assumes the smallest building block of the architecture, the basic elements, as fixed, and produces the product architecture by arranging (and re-arranging) these components into (larger) modules (A2+A4 or A2', medium case in Figure 2-4). For instance, for a vacuum cleaner, should the motor and the fan jointly form one module or two separate ones? In essence, this approach presupposes existing, basic elements, and the architecture definition is reduced to the determination of how these elementary elements are grouped into larger ones (i.e., the modules).

The criteria used to group the elements into modules vary across research fields and along the product's life. For instance, for products where the expected innovation rates of the underlying technology differ across components, it has been suggested to group components with similar innovation rates into modules (Langlois and Robertson 1992, Martin and Ishii 2000). Others have focused on reducing time or expenses of product development (Roemer 2000) or improving the product's end-of-life environmental performance (Newcomb et al. 1998) as criteria driving the module formation process. An important class of tools that have been developed to help in this module formation process is the category of interaction matrices.[13]

---

[12] While group technology strives for commonality along these different dimensions, their effect on commonality from a functional perspective may vary. For example, if a common manufacturing process is the goal, the part function is of only secondary concern. I am thankful to Dan Whitney to pointing this out.

[13] Many variations of matrices most current day authors use to determine how to form modules go back at least to some extent to the work of Steward (1981). His design structure matrix (DSM) is the basis for many derivatives. Browning categorizes the many different types of what he calls Dependency Structure Matrices into four groups: (1) Component based or Architecture DSM, (2) Team-based or Organization DSM, (3) Activity-based or Schedule DSM, and (4) Parameter-based or (lowlevel) Schedule DSM (Browning 1998). The first deals with functional interactions while the product is in use, the second with development team interactions. Both cases have no time component and most optimization algorithms applied to these problems attempt to distribute the product's

Some matrices document types and importance of interactions (Pimmler and Eppinger 1994), others indicate the components' levels of suitability to belong to the same module along multiple criteria (Huang and Kusiak 1998, Kusiak 1999). In most cases, columns and rows are re-arranged to minimize the unwanted interaction or to increase the desired 'similarity.' Genetic algorithms have also been suggested for this clustering process (Gu et al. 1997).

The 'medium' approach is also used in works that measure the impact of shifting the point of module formation in the production process on production costs (Ishii et al. 1995) or in the investigation of the differences in service costs for multiple configurations of a document handling system of a copy machine (Dahmus and Otto 2001).

This approach's underlying assumption is that functions are clearly defined on the level of the lowest, basic elements. Returning to the vacuum cleaner example, this means that the motor and the fan have distinctly separate functions. They can be combined, but they are not divisible. The possibility that some fraction of one element's function, say the motor, is delivered by another component, does not exist. In other words, building a matrix and filling it with the product's basic elements, establishes already the first layer of product architecture.

Common for these 'configuration' processes is that modularity is defined in approaching an optimum that combines elements into modules according to pre-set criteria. Although module boundaries vary according to the different criteria, in general, the goal is (a) to group 'similar' elements and (b) to transform interactions between modules into interaction within modules.

2.3.2.3 'Fundamental' (Re-arranging of function to components)

While the second subset was constrained by the pre-definition of sub-module level components, the third subset relaxes this constraint. This approach attempts to capture truly distinct product structures – designs that differ fundamentally in the way functionality is allocated to the elements (see fundamental case in Figure 2-4). As an illustration, consider the example of a computer. The medium approach would take basic elements and group them into modules like display, CPU, hard drive, energy unit, keyboard and mouse. In contrast, the

---

complexity to some extent evenly (1), or try to align functional product interaction with development personnel interaction (2). Groups (3) and (4) include an order or sequence of information, and optimization algorithms used for this type of DSM strive to reduce the amount of iterations during the development.

fundamental approach allows to describe the architectural difference if, for example, the data input function (typing) is re-allocated from the keyboard to, say, the display ('touch screen').

One way to find new function-component allocations is to map the functions on *potential* modules and then assess the viability of these potential modules along various criteria (O'Grady 1999). While this approach might create a new allocation scheme, it does so within the constraints of existing components. To overcome this problem requires a higher level of abstraction. Using customer needs and fundamental, basic functions, McAdams et al. compare different products to identify possible common modules (McAdams et al. 1998). They abstract the product functions required by customers into fundamental functions (e.g., convert electricity to rotation, import human hand and import human force, etc.) and analyze similarities between small household appliances like icetea-makers, coffee-makers, and palm grip sanders. Following a similar idea, Dahmus et al. compare function structures for common and unique functions across a product family to define possible product architectures (Dahmus et al. 2001). Obviously, these approaches offer some unique challenges. For example, how are functions compared and weighted with each other? Currently, most researchers use some sort of weighting scheme – either implicit or explicit. Research work that proposes optimization procedures (or design guidelines) often recommends interdepartmental negotiations to agree on these weights.

Compared to the previous 'medium' sub-group, this 'fundamental' sub-group uses a higher level of abstraction (physical function instead of basic components) to create the product architecture. To some extent, this abstraction also implicitly carries conditions for the module formation and interface definition (for example, 'convert electricity' requires certain materials and excludes others). It does so, however, on the least specific level of the three sub-sets.

On the whole, it seems that the level on which the elements are formed and the way in which they represent the product, carry information needed to determine differences in modularity. Therefore, if one were to operationalize modularity, information of the elements with respect to the product's total functionality would be needed.

2.3.3 Relations and elements

The third research set of the systems view combines the ideas of the two dimensions *elements* and *relations*. Its proponents argue that both dimensions are required for a complete description of modularity. One research approach employing such a composite definition suggests using the product architectures as a means to describe and determine levels of modularity. In his influential 1995 article, Ulrich defines the product architecture as "the scheme by which the function of a product is allocated to its physical components." He distinguishes two archetypes of product architectures: "A modular architecture includes a one-to-one mapping from functional elements in the function structure to the physical components of the product, and specifies de-coupled interfaces between components. An integral architecture includes a complex (non one-to-one) mapping from functional elements to physical components and/or coupled interfaces between components." (Ulrich 1995, p.422) This model of a modular-integral dichotomy has been employed in a broad range of fields, ranging from engineering (Allen and Carlson-Skalak 1998), to strategy (Chesbrough and Kusunoki 1999), to theory building (Schilling 2000).

Martin and Ishii have proposed another method that also considers the arrangement of functional elements, the function structure mapping and the interface specifications, i.e., both elements and relations. To support product family development, they suggest to measure (a) the innovation rates of components (both technology and market driven) and thus their likelihood to change, and (b) the extent to which changes in one component trickle through the rest of the product and propose to make qualitative assessments of both (Martin and Ishii 1996, Martin and Ishii 2000).

Some practitioners also use module definitions that include both dimensions, i.e., elements and relations: A module is a "complex assembly forming a closed function unit which permits specific differentiation and which, as a consequence of defined interfaces (function, geometry), can be developed, manufactured and assembled independently" (Wilhelm 1997).

This third set of research can be considered as a more comprehensive modularity description compared to the previous two. It appears that information about both relations (interfaces) and elements (modules) are required if differences in modularity are to be measured. However, the approach combining both dimensions still is difficult to operationalize because it remains unclear how different feature combinations along the two dimensions result in different levels of modularity. Section 2.6 will return to this question.

2.3.4 Contrasting elements and relations

Figure 2-5 summarizes the findings of the literature analysis through the lens of the systems perspective. The figure allows two major observations. First, there is no clear separation of the literature into different camps – neither along thought world boundaries nor otherwise. A few clusters can be identified, however. For example, the rather management oriented literature tends to assume predefined elements, i.e., it employs the simple decomposition differences. Another cluster is formed by those references taken from the engineering literature that use abstract functions to define the system's elements. They can be found in the lower right corner of the graph, i.e., allowing 'fundamental' differences in decomposition. Second, the literature bodies analyzed here cover the whole range of possible product decomposition differences (simple to fundamental) as well as the whole range of importance placed on interface considerations (low to high). The consequence of this for a generalizable modularity understanding is discussed further in the final section of this chapter.

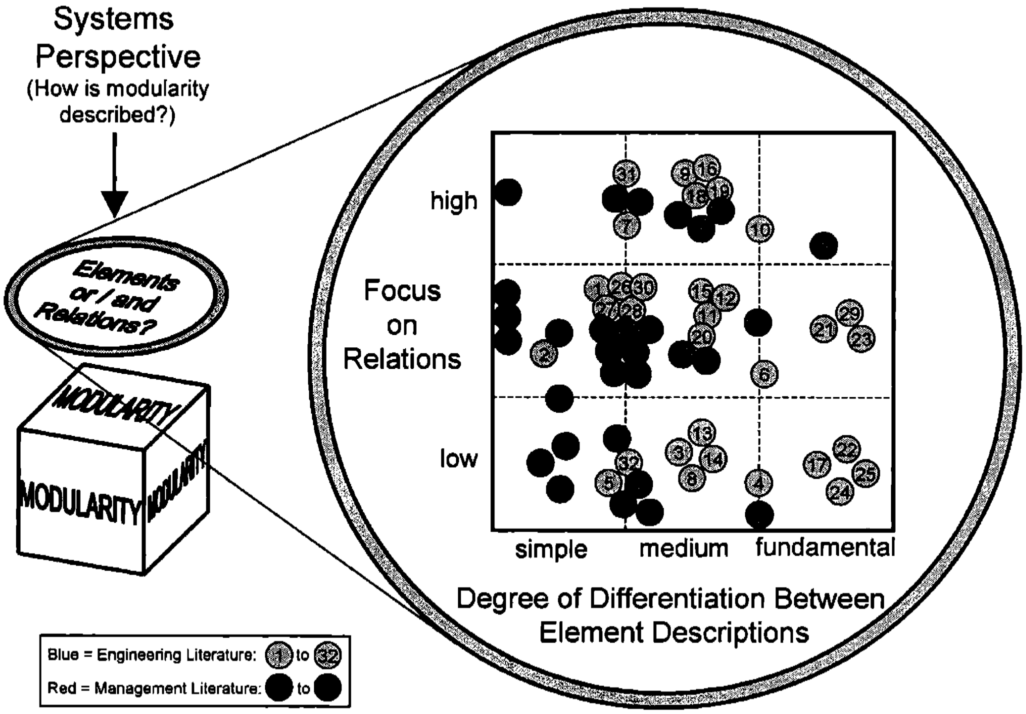


Figure 2-5: Modularity seen from the systems perspective: both elements and relations matter

2.4 Hierarchy Perspective: Technology Modularity or Business Modularity?

Almost four decades ago, Herbert Simon noted that complex systems tend to organize themselves in hierarchies (Simon 1962).[14] Others have found that almost all products are themselves part of 'nested hierarchies,' i.e., while exhibiting an internal hierarchy they are simultaneously part of an upper-level hierarchy (e.g., Christensen 1992a, Gulati and Eppinger 1996, Baldwin and Clark 2000, Schilling 2000).

However, the product's own hierarchy is not the only hierarchy that is associated with the product. Clark has pointed out the existence of multiple hierarchies. He makes a "distinction between hierarchies and associated resources linked to product and process technology, and those linked to customers and markets." (Clark 1985, p.249) In analogy to this distinction, I will present two hierarchies as vantage points from which the modularity issue has been pursued (Figure 2-6). The first one is a technology driven approach grounded more in the engineering world, and the second represents a market driven approach based in the business world. It is not so much the level of analysis that differs between the searches for definitions and descriptions of modularity, but rather the viewpoints from which researchers and practitioners begin their analyses and descriptions.[15]

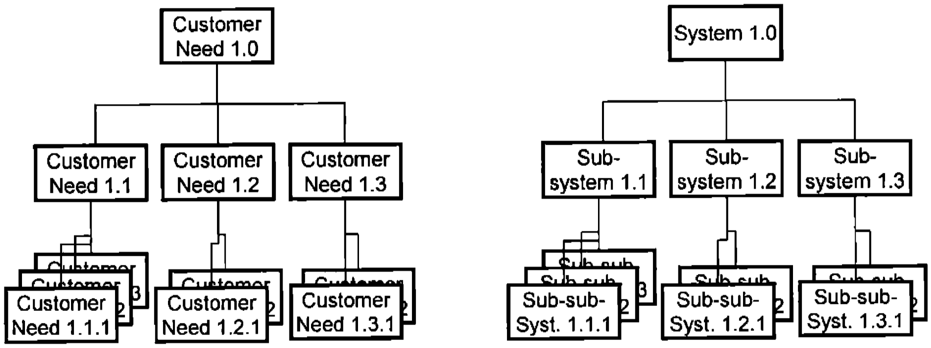


Figure 2-6: Business and technology hierarchies

---

[14] Simon defines a complex system as "one made up of a large number of parts that interact in a nonsimple way." (Simon 1962, p.468)

[15] There is some overlap with the possible clustering along a unit-of-analysis distinction, like component level vs. product level, or product level vs. product-family level. To cluster the literature along this distinction, however, is often difficult, especially when authors later extend their work to include additional levels of analysis. In contrast, the original approach is almost always different. (Nevertheless, the tables in the Appendix list in the 'hierarchy' section the unit-of-analysis for each reference in addition to the technology/business assessment.)

2.4.1 Technology modularity: How to build a product

The mental framework of the bottom-up approach is rooted in the engineering world. Engineers all over the world are trained and educated to break up problems that are too complex into smaller ones that can be solved. Engineers want to create products 'that work.' This implies that there is something that products 'do' and this 'doing' is nothing other than the function of the product in technical terms. Solving problems in the engineering world is finding ways to create mechanisms that function as desired. Pahl and Beitz, for example, recommend the following four steps for conceptual design: (1) abstract to identify the problem, (2) establish function structure, (3) develop a working structure,[16] (4) evaluate and select best combinations. In subsequent design stages, i.e., embodiment design, the design is completed (Pahl and Beitz 1996). Function structures, the part of interest here, refer to the 'flow' of energy, materials, and signals that 'travel' through the system. They are themselves hierarchically structured (Figure 2-7).

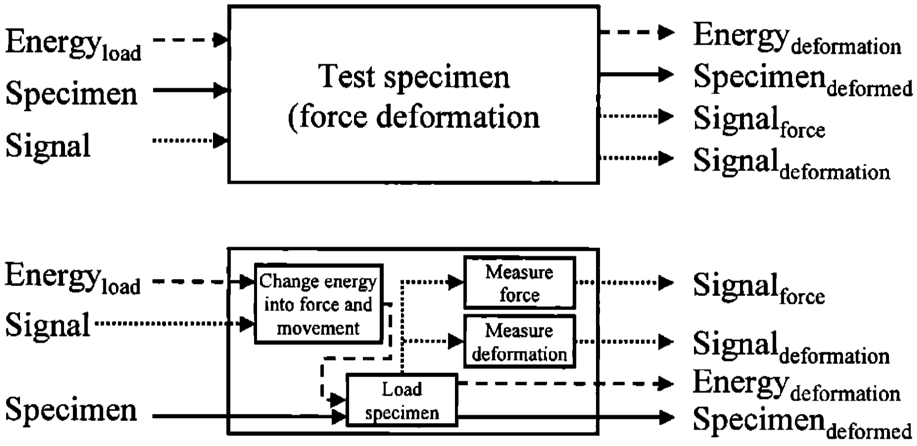


Figure 2-7: Overall function (top) and sub-functions (bottom) of a testing machine[17]

Having defined functions on these fundamental levels, engineers 'assemble' the products in their mind. That is, functions that are similar, or use the same working principles, can be

---

[16] Working structures describe working principles together with geometric information, such as location and direction. Working principles are physical effects, such as gravity, friction, etc. (see Pahl and Beitz 1996).

[17] Pahl and Beitz 1996, p.152

combined. Precisely this approach has been used to support developing modular products. Stone et al., for example, develop three heuristics to identify possible modules (Stone et al. 1998). The three heuristics they suggest take on the engineers' perspective on functionality: dominant flow, branching flow, and conversion-transmission are all technical views of what the product does.[18] Building on this idea, Stone and other researchers have extended it to increase its applicability to other products (Stone and Wood 2000, Stone et al. 2000a), to include product family considerations (Stone et al. 2000b, Dahmus et al. 2001), or brand considerations (Sudjianto and Otto 2001). Function based module definitions have also been explored to accommodate recycling goals (Allen and Carlson-Skalak 1998).

The characteristic common to all technology approaches is a detailed study of the product's technical functionality, followed by assigning functions or set of functions to physical elements. Finally, elements are (re-)combined into complete products or product families. The modules formation process, and, therefore, the modularity definition, takes place at a rather low, technically detailed, level within the technology hierarchy linked to product and processes.

2.4.2 Business modularity: How to serve markets

While engineers view a product as "a complex assembly of interacting components," the marketing perspective sees a product as "a bundle of attributes" (Krishnan and Ulrich 2001, p.3). Most often, researchers that choose a market-driven approach, start with the product's potential or existing market(s), divide the market(s) into categories or segments, and propose architecture(s) to simultaneously serve these market segments. Two conflicting objectives drive this process: (a) the need to offer the customer as much variety as she wants and (b) the need to reduce the variety for cost reasons, i.e., to strive for commonality. The fundamental question is how to translate different customer needs and expectations into product (family) architectures.

The way the variation of customer needs is treated is key for this mapping from customer needs to product architectures. Some approaches focus entirely on the extent to which

---

[18] Dominant flow refers to the highest ranking (from customer needs) non-branching flow (e.g., the specimen in Figure 2-7), branching flow refers to modules defined by branching function chains, and conversion-transmission refer to conversions of energy or material of one form into another form of energy or material. Note that Stone et al.'s approach also introduces customer needs to evaluate the modules. Basic starting point, however, are the functions in engineering terms.

commonality is achieved, others consider different types of customer need variations, and yet others model the tradeoff between commonality and distinctiveness.

In pursuit of commonality, the use of identical parts has received different labels, depending on the level within the product hierarchy and the location in the value chain. For instance, an approach that seeks to reduce variety on the component level, primarily on the shop floor, has been termed group technology (e.g., Hyer and Wemmerlov 1984). Others have focused on the extent to which an existing product family accomplishes the use of common parts and components. Kota and Sethuraman, for example, develop a product line commonality index that measures how far a given product family is away from the (manufacturing) ideal to have identical components (Kota and Sethuraman 1998, Kota et al. 2000). Similarly, MacDuffie et al. have developed composite variables that reflect, among other things, levels of part commonality for statistical analyses (MacDuffie et al. 1996). On higher levels of the product hierarchy, i.e., if a larger fraction of a product is 're-used' in other products of the product family, the term *product platform* has received considerable attention. A product platform is described 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." (Meyer and Lehnerd 1997, p.39) Some understand the platform as offering a configuration space within which a customer variety can be produced. For example, Siddique et al. developed a product family reasoning system that identifies candidate sets of platforms out of a set of existing products, subject to constraints imposed by other products or assembly facilities (Siddique et al. 1998, Siddique and Rosen 2000).

For a more detailed consideration of customer need variations, Yu et al. suggest a customer need analysis that represents customer need target values as probability distributions across market segments and over time (Yu et al. 1999). They also introduce three categories of what they call portfolio architecture: fixed, adjustable, and platform. They find that if the customer need distribution is stable over time and narrow in its distribution, a single, fixed architecture is sufficient. If a need distribution exhibits ergodicity, i.e., the need distribution across the population at a single point in time is equal to the distribution of every customer over time, they recommend an adjustable architecture. The requirement for legroom in a car is an example of such a customer need. It is served with a single but adjustable architecture. If the target values of customer needs are not stable over time or across segments, they suggest to isolate the corresponding feature in a module and to use a platform architecture for the rest of the product. As an example, they use the cover of a toaster to indicate a need that changes with trends. As another way to offer the customer variety, it has been suggested to create 'optimal' building blocks and let customers 'customize' their products themselves (Tseng and Jiao 1996, Tseng and Du 1998). To design the building blocks, clustering of design parameters is suggested. This approach seems to work well for products where the differentiation is one of scale, i.e., the same component with a different performance level (e.g., power supply switches).

Finally, others model the opposing forces for variety and commonality as a trade-off. For example, Robertson and Ulrich propose a method to balance distinctiveness with commonality. They propose to define the number of chunks (physical pieces) of a product as roughly equal to the number of differentiating attributes (Robertson and Ulrich 1998). Acknowledging that the importance of various factors going into this tradeoff might differ, they suggest an iterative approach as a correction mechanism. The difficulty in handling this multi-factor trade-off is also recognized in other research. For the case of a spacecraft family based upon a common design, a negotiation process is suggested to agree on common parameters across all missions (Gonzalez-Zugasti et al. 2000).

In sum, the market driven approaches begin with the understanding of a need for product variety and suggest methods to identify commonality on various levels of the product family and, subsequently, of the product. Modularity definition occurs on a rather high level of the hierarchy linked to customer and markets.

2.4.3 Contrasting technology and business modularity

As previously stated, the hierarchy perspective allows one to investigate the location where the module creation occurs. This 'location' has two aspects: its position in the product hierarchy, and its position in the timely order of the design process. The first aspect, the position in the product hierarchy identifies a hierarchy level such as component, module, product, or product-family. The second aspect, which is the focus of this section, describes whether the module creation occurs before or after the customer requirements are translated into technical specifications. Consider the zig-zagging between functional and physical domain during the design process as described by Suh (Suh 1990, p.36ff). During the design process, what Suh calls the functional requirements (FRs) on one hierarchy level need to be conceptualized on the associated physical domain via design parameters (DPs), before the next lower level of FRs can be decomposed (Figure 2-8).

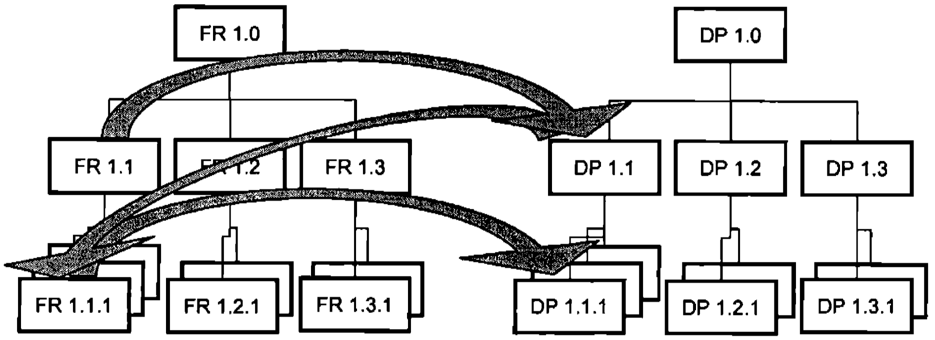


Figure 2-8: Zigzagging between functional and physical domain

Similarly, what I call business modularity is the module creation before the customer needs are translated into technical specifications, i.e., the designer combines or separates customer needs on rather high levels of hierarchy. In contrast, technology modularity happens when the designer creates modules from the set of technical specifications. Here, the translation of customer needs into design parameters precedes the module creation and the module creation occurs on rather low levels in the hierarchy. Thus, depending on where in the zigzagging process the module formation takes place, the resulting 'modularity' may differ. Figure 2-9 summarizes how the literature analyzed for this chapter can be catalogued in the two categories 'business modularity' and 'technology modularity.'

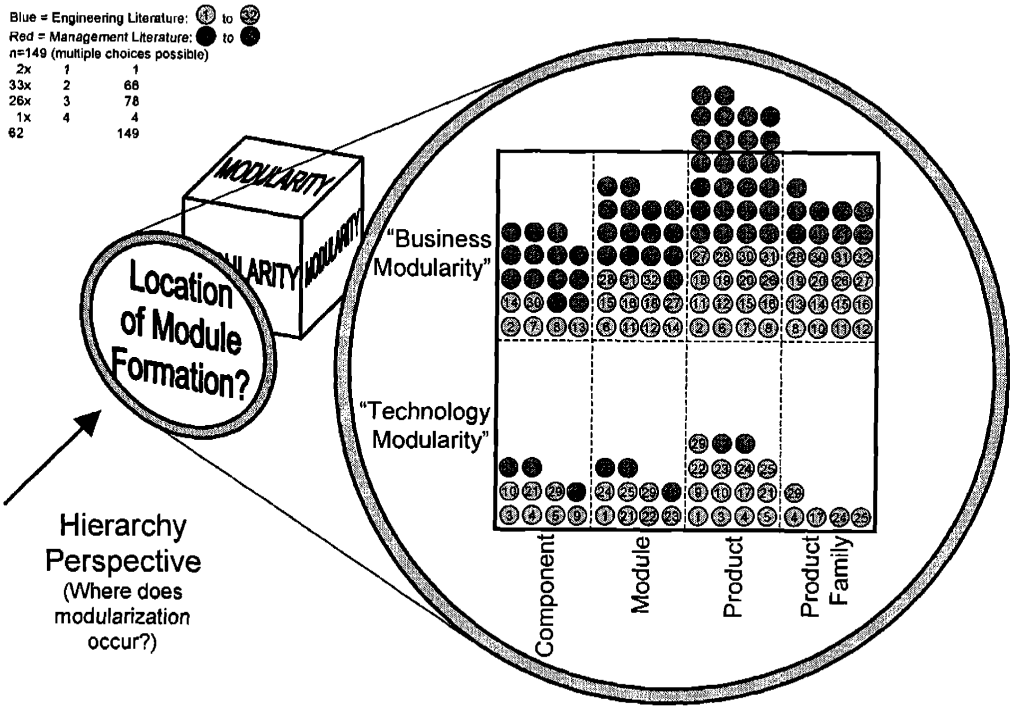


Figure 2-9: Modularity seen from the hierarchy perspective: business or technology modularity?[19]

Not surprisingly, the management literature is found almost completely in the business modularity section. In contrast, engineering literature can be found in both categories. With respect to the product hierarchy level, most references in both categories emphasize the product level, with module and product-family coming in second for the business modularity, module and component levels for the technology modularity.

Understanding the differences in where and when the module creation occurs during the design process is important for grasping the meaning and intention of the selected module, and thus, its role in the product and in the product family.

---

[19] While the figure separates 'business modularity' and 'technology modularity' unambiguously, the categorization along product hierarchy levels is less clear-cut. For this reason, multiple occurrences *within* a modularity category are possible. For example, reference no. 22 discusses module-level and product-level issues, but both in the technology modularity category.

2.5 Life Cycle Perspective: Modularity for Whom?

The third perspective discussed in this chapter takes on various positions over a product's lifetime. Every product runs through different phases in its life (Figure 2-10). Each life phase sets different performance goals for the product. For this reason, a product that is optimized for one phase is not necessarily optimal for others. Since this is a general optimization phenomenon, it is not different for developing optimal 'modularity.' Just as numerous approaches have been developed to optimize products for various purposes (e.g., see the DFX literature), a number of methods have been proposed to develop modular products. Depending on the field of application, these techniques arrive at different definitions of what a 'module' represents.[20]

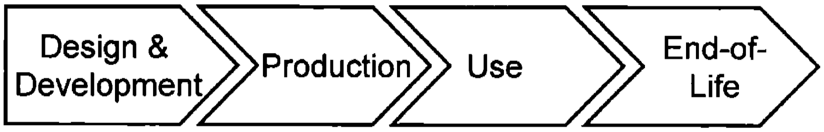


Figure 2-10: Generic product life-cycle phases[21]

2.5.1 Design and development

Researchers working on design and development (D&D) processes are typically concerned with the question of how to reduce the resource consumption (cost & time) for D&D, condition to a certain level of product functionality and quality.[22] Since today's complex products are already beyond what a single human mind can work on, the development of these products is split into work packages, which are assigned to various people and teams in the organization. Organizational structures tend to mirror the structure of the products the organization makes

---

[20] Some have suggested defining 'modularities' for different phases: Modularity-in-Design (MID), Modularity-in-Production (MIP), and Modularity-in-Use (MIU) (Baldwin and Clark 2000, Sako and Murray 1999). As this section will show, this distinction still appears too coarse to make *modularity* operationalizable.

[21] Product life cycle focuses here on the individual product as the unit of analysis, in contrast to a product population.

[22] Another concern could be to create product architectures whose development processes allow designing *better* products.

(Henderson and Clark 1990).[23] Organizational structures create a need for communication and efficient communication maximizes the resource productivity. Thus, the question is: what are the architectural characteristics of a product that minimize the resources required to develop it?[24] What are the modules that facilitate the development?

Consequently, researchers have proposed methods that 'modularize'[25] the product, and in turn the design process, such that the communication effort is minimized. The most fundamental account is, that a task that exhibits a low level of interdependence with other tasks, has a higher probability to be successfully solved than a task that has a high degree of interdependence with other tasks (von Hippel 1990).[26] Based on the design structure matrix (Steward 1981) researchers have developed several modeling techniques to predict the impact of product architecture choices via organizational structure on development time and cost (Eppinger et al. 1994, Baldwin and Clark 2000, Ahmadi et al. 2001).[27] On a very generic level, module definitions in these works aim at minimizing the communication effort and at reducing the risk level within larger development efforts.

Another method to describe the ease (or difficulty) with which a product design can be changed in one dimension without affecting other dimensions has been developed by Suh (1990). Motivated by his central question, "as we map DPs [design parameters] in the FR [functional requirement] space, are there certain rules or axioms that are satisfied by a good design?" (Suh

---

[23] Researchers with different foci point out different consequences of this effect. While some argue this mirroring is optimal for an efficient development process (Goepfert 1998), others point out the strategic disadvantage it can cause for situations with technological change (Henderson and Clark 1990). In fact, for complex multi-technology products it has been argued that companies need to maintain a technology base that is broader than what they actually produce (Brusoni and Prencipe 1999).

[24] There actually is a two-way relationship between product architecture and organizational design (Gulati and Eppinger 1996). Nevertheless, for the purpose of this analysis, I focus on the effect product architecture/modules have on the organizational performance. In addition to the product architecture, organizational decisions alone, like sequential iteration or overlapping, also influence the efficiency of development processes (e.g., Smith and Eppinger 1997b, Krishnan et al. 1997).

[25] The literature often prefers terms like 'partitioning' and 'task blocks' rather than 'modularization' and 'modules.'

[26] There are some counterarguments. For example, Sosa et al. find that design teams of what they call integrative subsystems have a better understanding about 'their' cross-subsystem interfaces than design teams working on modular subsystems (Sosa et al. 2000). It is unclear, however, (a) to what extent this is a function of how well the product is understood in general, and (b) whether this performance deficit of the modular teams is outweighed by their performance increase with respect to 'regular' task when compared to highly integral architectures.

[27] Individual studies employing DSMs often search for an optimal way to organize product development for a *given* product architecture. Taken together, however, they point out differences in product architectures that allow, or hinder, efficient product development processes.

1990, p.46) Suh proposes two axioms to guide the designer: (1) Maintain the independence of FRs (The Independence Axiom) and (2) Minimize the information content of the design (The Information Axiom). The first axiom then allows one to distinguish three types of designs: uncoupled, de-coupled, and coupled. While 'good' design is not explicitly defined, the degree of coupling describes the likelihood for further re-design when one design parameter is changed. In other words, higher degrees of coupling increase the risk of design iterations.

2.5.2 Production

If one understands component interchangeability as modularity, then the idea of a simplifying concept in the world of production is already a century old. What Henry Ford accomplished for components *across multiple instances of a single product* (identical, interchangeable parts), was proposed by an automotive engineer in 1914 *across multiple different products*: standardized wheel sizes, hubs, bearings, axles and fuel feeding mechanisms (Swan 1914).[28] Half a century later, in 1965, Starr proposed modular production as a new concept to provide product variety. His emphasis on "maximizing the combinatorial variety of assemblies from a given number of parts" (Starr 1965, p.138) implicitly requires function containment within the interchangeable components in order to not compromise the product function as a whole. Thirty years later, Pine suggested a similar approach for mass customization (Pine 1993). Although he argues that mass customization targets individual customers while producing variety alone does not necessarily do so, the tools behind it are very similar. Building on Ulrich and Tung's work (1991), he proposes six categories of modularity: component-swapping, component-sharing, cut-to-fit, bus, sectional and mix modularity. Again, these definitions implicitly carry some features for the modules: function containment (otherwise some functionality of the product would be lost), a limited number of different interfaces, and some notion of the ease with which interfaces can be physically connected and disconnected.

In production, 'modules' are predominantly understood as assembly modules. Typical characteristics are the collection of components in/on them and the 'ease' with which the connection can be made. For products that differ only along a few performance dimensions,

---

[28] Swan had much larger components in mind than Ford with his move to interchangeable components.

combinatorial design has demonstrated its advantages (Whitney 1993). Similar views are often found in the automotive industry (Wilhelm 1997, Delphi 1999).

Another argument for modular design is made with respect to logistics. The literature promoting late customization or postponement strategies to reduce inventory and shorten lead times often advocates modularity: "A product with a modular design provides a supply network with the flexibility that it requires to customize a product quickly and inexpensively." (Feitzinger and Lee 1997, p.117). To allow a late customization can require rethinking of the order of the production processes for a particular product. Researchers have developed models to determine how supply chain characteristics like value added per stage or choice probabilities per stage and option can be used to guide operations reversal to reduce the variance in the supply chain (Lee and Tang 1998).

In sum, most module definitions concerned with the product's production phase are aimed at lowering production and logistics costs, and reducing lead times. Major ideas behind this are economies of scale for modules that can be used across product families, complexity reduction throughout manufacturing and assembly, and inventory reduction through risk pooling and postponement.

2.5.3 Use / Operation

Two aspects fall under the use phase category. First, many module definitions implicitly use the use phase, because they build on the product functionality, i.e., the function the product will perform while it is in use or operation. Most 'module' definitions that originate in the engineering world follow this idea (see discussion in section 2.3.2 and Appendix). Consequently, function containment is of major importance, albeit sometimes only implicitly. For example, if motor power is a distinguishing characteristic for different products in a product family, than the function 'propulsion' should be contained in the module.

Second, and even more so than for assembly, the 'module' idea for the use phase is focused on a single interface characteristic: the effort it takes to separate it. This is a result of the idea that modularity-in-use allows product re-configuration on an effort level lower than the original production, often enabling the *user* herself to customize or re-configure the product. Similarly, up-grades or maintenance require (a) interfaces that are easy to separate and (b) functionality containment in order to have the desired results. For instance, for service purposes components with equal lifetime or similar failing frequency should be located in one module (Newcomb et al. 1998). In general, due to the similarity to the assembly portion of the production phase, very similar concepts are underlying modularity-in-use with respect to function containment. Differences with respect to the interface separability may remain.

2.5.4 Retirement

The final phase of a product's life is its retirement. Two major paths exist for the product after its first life, depending on the post-life intent. First, it could be refurbished as a unit or its components could serve as spare parts; and second, it (or parts of it) could be transformed into other use. For assembled products, the former always includes a disassembly process, the latter only if either material value makes it economically viable or legislation requires the separation of hazardous materials.

The post-life-intent, for example, can be expressed as material recycling, which makes modules desirable that contain as few different materials as possible (Allen and Carlson-Skalak 1998, Newcomb et al. 1998). To improve an existing design's environmental performance, a procedure has been suggested that identifies the constraints, that – if changed – would offer the greatest improvement towards a more environmentally friendly design (Coulter et al. 1998). In their example, an automotive center console, the authors change the materials of the components, but not the modules' boundaries. To summarize, the requirements of a post-use phase are often entirely different from those in design, production or use. As a result, module definitions vary again.

2.5.5 Contrasting multiple life-cycle phases

Figure 2-11 summarizes the literature analysis with respect to the life-cycle phases. As expected, the engineering oriented literature focuses mostly on the use phase: designers think about what products are supposed to do for the user / customer. The management literature has its focus more on the design and production phase. This is partially caused by the operations management references that often have these two phases as their focus. Overall, production and use phase are the most populated phases, followed by the design phase, and retirement has only recently become a field of research with respect to modularity.

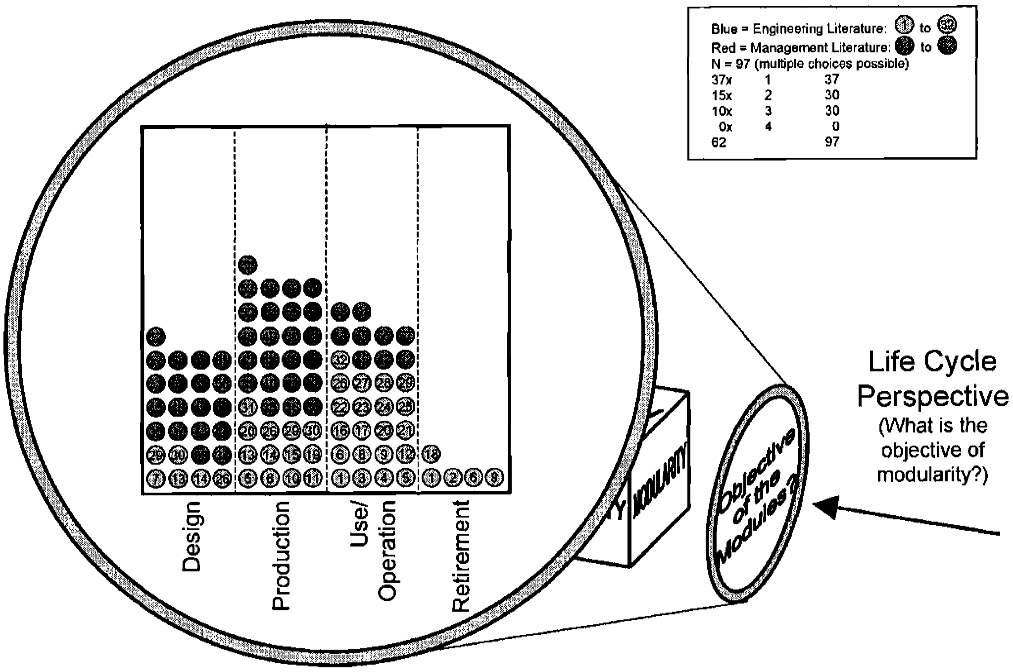


Figure 2-11: Modularity seen from the life-cycle perspective: what is modularity's objective?

Similar to the first two perspectives, the third also makes the differences in the understanding of what constitutes modularity and modular products visible. The differences here originate from different foci along the product life cycle chosen in the different articles. The objective of a module drives its characteristics, i.e., its size, form, and relations with the rest of the product. As a consequence, modularity for one life-cycle phase can considerably differ from modularity for another life-cycle phase. Furthermore, even within phases, modularity can differ. Consider, for example, the production phase: standardization of manufacturing rests on different modularity features than late customization does.

2.6 Discussion and Chapter Conclusion

The foregoing discussion has analyzed product modularity concepts from three different perspectives. This section recaptures the findings and interprets their meanings with respect to a general concept on product modularity. The ultimate goal behind this approach is to find a way to operationalize modularity, because an operationalization is necessary to identify and investigate modularity's effects.

The first perspective, systems, has indicated that a product description focusing only on elements or only on relations limits itself unnecessarily. While this may be appropriate for some applications, in search of a more comprehensive understanding of product modularity, the research approach that includes both dimensions appears to be advantageous. Despite being conceptually powerful, however, this approach in its current form is also difficult to operationalize for two reasons. First, while the simultaneous description of elements and relations presents two layers of information, the level of dependency between the two is unclear: Do they always change simultaneously? In other words, can a product architecture without a one-to-one mapping from functional elements to physical components have de-coupled interfaces? Or can one with a one-to-one mapping exhibit coupled interfaces? Consider the example of attaching the MP3 module from the PDA example with adhesive bonding instead of a plug. The one-to-one mapping would still exist, but the interface characteristic was changed. It seems that some interface characteristics can change (or be changed) without simultaneously changing the function-component mapping, and vice versa.

Second, and this is somewhat related to the first argument, both dimensions are themselves multi-faceted. The function-component allocation can have different results at different places throughout a product (e.g., one-to-one mapping at one portion of the product, a non one-to-one mapping at another). How does this affect the modularity assessment of the product architecture? Likewise, interfaces can differ in multiple dimensions. Does 'coupling' mean the interface's role for the product function? For its design? For its manufacturing? Or for its disassembly?

The second perspective juxtaposed two hierarchies (product and market). While the product design process follows a zigzag process that connects both hierarchies, the module formation can take place at different 'locations' in this zigzag process. At one end, technology modularity, the product is built by finding solutions for elementary problems, these technical solutions are then combined into chunks, modules and ultimately into products, i.e., modules are 'formed' by a combination and aggregation process. In contrast, at the other end of the zigzagging, business modularity divides markets into segments and identifies product features that need to be separate and others that can be common. The modules, or the product's modularity, are developed from there. In other words, business modularity focuses on product variety while technology modularity focuses more on commonality.

What the analysis of the literature from this perspective demonstrates, is that whatever is seen as modularity depends on when and where in the zigzagging process the module formation occurs. One reason for this possible non-congruence of modularity between the market related hierarchy and the one related to the product hierarchy is the translation process of customer requirements into technical specifications.[29]

Finally, the third perspective has investigated modularity concepts for different life cycle phases. All design techniques guiding module creation for a particular phase necessarily prioritize design goals, either implicit or explicit, just as any other optimization approach. While this is a legitimate goal for a particular design task, it makes the term 'module' alone less useful to distinguish various types and levels of modularity.

In sum, the literature analysis offers three major insights. First, product modularity appears to be a bundle of product characteristics rather than a single condition. Both function-component allocation schemes and several interface characteristics are required for a complete description. Second, while function containment is, explicitly or implicitly, part of most modularity descriptions, what is understood as a function, however, can vary with the viewpoint taken. Therefore, a modularity description should identify its own viewpoint. Third, module and interface characteristics are interpreted differently, depending on the life cycle phase. For example, designers emphasize low levels of functional interaction, producers prefer easy installation, and users vote for easy disconnection.

---

[29] There are powerful and tested tools to translate market research into technical specifications (e.g., House of Quality, etc.), that focus on converting market demands into performance requirements in engineering metrics. However, the process by which the interpretative gap between 'feature' (market) and 'technical function' (technology) narrows over time seems to be dependent on the actors themselves. The emergence and stabilization of product market categories has been described as social construction process through ongoing interactions between producers and consumers (Rosa et al. 1999). Perhaps, the understanding of what constitutes product families, and thus, products and ultimately modules, undergoes a similar stabilization process.

Apparently, there is not a single definition for *modularity* that holds under all circumstances, and is simultaneously operationalizable. Acknowledging the multi-faceted character of module definitions along a product's life and across various participants' viewpoints, multi-perspective approaches might offer more comprehensive understanding of the phenomena. One possibility are simultaneous assessments of different perspectives (Tseng and Jiao 1998, Jiao and Tseng 1999), of different phases (e.g., product, production and sales in Du et al. 2000) or weighting procedures to accommodate the to some extent conflicting objectives (Gu et al. 1997).

An alternative approach is to un-bundle the product characteristics normally subsumed under *modularity* and to tie them individually to viewpoints and life cycle phases. In the next chapter, I will develop a product architecture taxonomy that permits comparative multi-dimensional product architecture descriptions. This descriptive approach then enables one to link architectural feature differences individually to effects and consequences (e.g., costs, time, etc.) in different life cycle phases.

3. A FRAMEWORK TO MEASURE MODULARITY

3.1 Chapter Introduction

Based on the findings of the previous chapter, in this chapter I develop a framework that will allow capturing multiple aspects of modularity while being applicable across different disciplines.

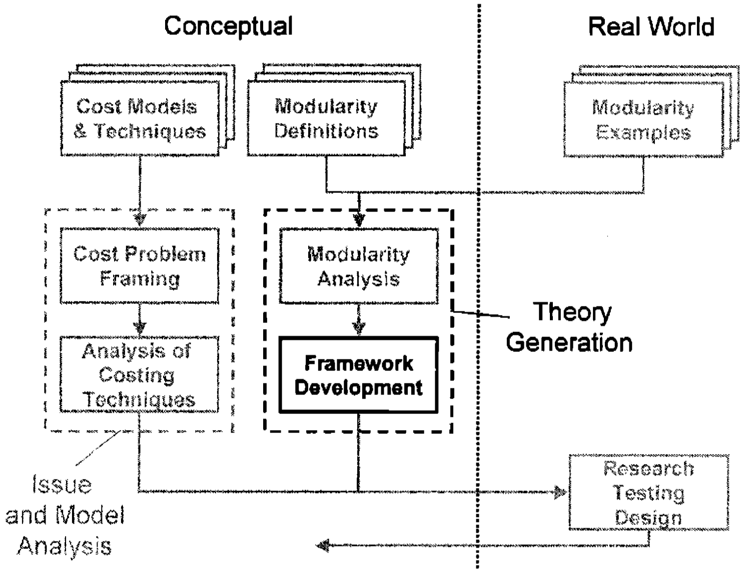


Figure 3-1: Content of chapter 3 in relation to the other chapters

The chapter is organized as follows. The next section revisits the discussion of optimization vs. description with respect to the fundamental goal of the method to be developed and points out why the method concentrates on the artifact 'product' as the unit of analysis. The third section lays the foundation for the replacement of *modularity* with *product architecture*. The fourth and main section of this paper develops the product architecture description methodology and discusses it step-by-step with a detailed example.

3.2 How to Operationalize A Multifaceted Concept

Having identified 'modularity' as a bundle of product characteristics that are prioritized differently by various approaches and definitions, I propose in this chapter to develop a product architecture description methodology that can capture all of these characteristics along their multiple dimensions. This approach will be descriptive and will focus on the artifact 'product.'

3.2.1 Optimization vs. description

As the previous chapter has demonstrated, the way 'modularity' is used in the literature is not independent from the researcher's viewpoint (see section 4 in the previous chapter), nor from his objective with respect to the product life-cycle phase (see section 5 in the previous chapter). As a result, different analyses arrive at different module definitions, i.e., they define modularity differently. A large number of variables and diverse set of interests make optimization always problematic.[30] While every optimization necessarily introduces objectives and constraints, the overarching goal of the research presented here is to investigate the economic effects of a multidimensional concept (i.e., modularity) on multiple participants (i.e., designer, maker, user, etc.). Once the relationships are better understood, optimization procedures can be applied for individual perspectives. Consequently, at this exploratory stage of the research it appears advantageous to separate the description of products from their evaluation. For these reasons, a descriptive approach is chosen here to depict modularity.

3.2.2 Focus on artifact 'product'

Two reasons guide the selection of the product itself as the unit of analysis. First, in contrast to organizations or tasks, an artifact represents itself to multiple investigations in the same fashion. Most of its attributes are relatively easy to measure. In other words, if we understand the modularity information as the input parameter, then the choice of the product as the carrier of the information increases the reliability of the input parameter. Second, while precise

---

[30] In their work on systems architecting, Maier and Rechtin conclude regarding this problem: "This primacy of complexity in system design helps explain why a single 'optimum' seldom exists for such systems. There are just too many variables. There are too many stakeholders and too many conflicting interests. No practical way may exist fro obtaining information critical in making a 'best' choice among quite different alternatives." (Maier and Rechtin 2000, p.6)

engineering data is not always easily available, the products themselves are often easier to access than internal company data.

3.3 Product Architecture as a Replacement for Modularity

If we understand 'modularity' as a collection of product characteristics or features that are differently prioritized by different stakeholders, and if we want to operationalize 'modularity' to measure its impact, we will need a construct that can cover all aspects of modularity simultaneously.

Chapter 2 has laid out in detail why this goal cannot be accomplished by one of the existing modularity definitions. All of the existing definitions suffer from either one of the following problems, some of both. First, since the modularity definition is based on a specific perspective, it needs suffices such as *in-design, in-production*, or *in-use*. In fact, the added specification needs to be much finer, such as indicating if manufacturing, assembly, or logistics is the focal point of *production*. In addition, even if this extra specification would be added to the term *modularity*, most of the existing definitions actually are constraint to the comparison of very similar architectures. The reason for this is that most definitions simply count components and cannot distinguish between those having different roles in a product.[31]

As a consequence, I argue to replace the ambiguous term *modularity* with the *product architecture* as the description of fundamental product characteristics, and to determine the differences along architecture's individual dimensions that resemble modularity ideas. To do so, I will build on a widely used definition of product architecture suggested by Ulrich. He called it "the scheme by which the function is allocated to its physical components." (Ulrich 1995, p.422) In addition to the allocation, his definition also includes a "coupling of interfaces." In his typology, he distinguishes between modular and integral architectures. Because of its conceptual power and its widespread acceptance across disciplines, and because it is one of the more comprehensive approaches to encompass different kinds of decompositions as identified in the previous chapter, I choose Ulrich's definition as a starting point.

---

[31] The previous chapter discusses in detail that most existing frameworks allow only product decompositions that differ only on simple or medium levels.

Ulrich points out that his typology describes ideal types: "The types shown are idealized; most real products exhibit some combination of the characteristics of several types. Products may also exhibit characteristics of different types depending on whether one observes the product at the level of the overall final assembly or at the level of individual piece parts and subassemblies." (Ulrich 1995, p.424). As a consequence, in order to operationalize 'modularity,' what is needed is a way to determine where in between these extremes a particular design is located in the space of possible architectures, or at least how two or more architectures compare to each other with respect to their locations in this space. To develop a framework that allows this type of 'position determination,' I will build on Ulrich's work and expand it in essentially three dimensions. These expansions are grounded in the findings of the literature analysis presented in the previous chapter.

First, for his ideal cases Ulrich defines jointly how functions are allocated to physical components and how interfaces are coupled. I will relax this joint requirement. While I agree that both dimensions are necessary for a complete architecture description, I argue that these two dimensions are to a large extent independent from each other and, consequently, should be treated independently.

Second, both dimensions *allocation* and *interfaces,* are themselves multi-dimensional constructs. Therefore, the framework to be developed needs to be capable of capturing all aspects of these multiple underlying dimensions.

Third, since the notions of *modular* or *integral* are associated with an allocation of the functionality in the product, it is unclear whether a term for the entire product is a reasonable aggregation or whether it hides too much of the information of interest.[32]

With these requirements in mind, the following section develops a framework to identify and measure architectural differences usually subsumed under the headline of modularity.

---

[32] To my knowledge, only a few authors have begun to discuss this issue. For example, Whitney 2002 forthcoming discusses the delivery of Key Characteristics (KCs) and determines four different possibilities for an architecture: modular, chain, integral, or chain-integral. Although these assessments differentiate between more than two combinations of functions (KCs) and parts, they are still aggregated for the entire product.

3.4 Multidimensional Product Architecture Description Method

The framework to compare architectural differences follows a systems idea by proposing that every product description consists of two major dimensions: its elements and the relations between them.[33] For the purpose of the development of the framework, I will call the elements *components* and the relations *interfaces*.[34] Both are necessary to describe a product comprehensively. I shall discuss the procedure to determine both dimensions in all aspects below.

3.4.1 Elements and their role in the product: the function-component allocation scheme

To build on the definition that a characteristic feature of a product architecture is the way in which functions are allocated to components, a mechanism is needed to determine and measure this dimension reliably. In other words, for all three pieces of the function–component allocation (FCA) scheme a rule-based procedure is needed to ensure repeatable results: How is a function determined? What is a component? And how is the allocation determined?

3.4.1.1 Functions: What are they?

In order to determine 'function' for the proposed product architecture taxonomy we distinguish between three different kinds: technical functions, features, and attributes. The most obvious description of a function is the one where it is used in a technical sense. For example, *acceleration* is an intended function of an automobile. We define that a product provides a certain function when it is capable of delivering a function at a determined performance level.[35]

---

[33] Others have used the chains that deliver so-called Key Characteristics (KC) to identify differences in product architecture. For example, Cunningham and Whitney propose a qualitative method to capture the chains that deliver the KC under investigation and evaluate their impact on integration risk (Cunningham and Whitney 1998). The authors argue that the height and the span of the ends of the chains are characteristic for the risk to deliver the KC and that shorter and 'lower' chains offer less risk. Generally, this could be achieved in two ways: (a) the chain is 'isolated' in a lower –level unit (e.g., subassembly) or (b) by reducing the number of subassemblies and component through parts integration. While the former suggests a modularization process, the latter rather describes an integration process on the module level.

[34] The term *components* serves here as a pure placeholder. It can include subsystems, modules, components, and parts. What constitutes *interfaces* is detailed in section 3.4.2.

[35] Determining functionality often raises the question regarding performance. In some cases the performance is directly measurable and varies along a continuum (e.g., MB per disc area for disc drives). Then further inquiry is needed to determine on what basis to compare products and their architectures. In many other cases, however, the functionality has already passed a certain performance threshold. If the functionality then is perceived more as a binary variable (e.g., there are power windows in a car door or there are not), rather than on a continuum (e.g.,

As feature we understand product characteristics that escape a technical function description, but still provide a valuable product characteristic from the customers' perspective. For example, the *color* or *surface structure* of an appliance or the aesthetic appearance of an automobile are features. Features' performance levels are typically either hard to determine ("Does a coupe look 'better' than a sedan?") or are entirely based on individuals' perspectives ("I prefer a red car over a blue one."). Finally, attributes are characteristics of the physical components themselves. *Weight* and *mass* are examples of these kinds of characteristics.[36]

3.4.1.2 How to determine the appropriate functions

Generally speaking, every product's function can be decomposed into sub-functions, just as most physical products exhibit hierarchical product structures with respect to their physical assembly.

Two thoughts should guide the determination of the appropriate level in the functional hierarchy: the comparability and the applicability. First, since this framework is developed as a tool to compare different products with respect to their differences in product architectures (i.e., with respect to their different levels of modularity), the function selection should consider choosing a level on which the variance in function allocation between the products becomes visible. In its most simple case, this means choosing functions that are in full or to the same degree delivered by the different architecture candidates.

Second, the function selection should neither choose the highest level of the functional hierarchy nor the lowest in order to be meaningful. The function on the highest level is necessarily provided by all components, for otherwise there would not be a reason for them to be there at all. For example, assume a hair dryer is the product under investigation. Its main (and highest level) function is to *dry hair*. If *to dry hair* were selected as a function, the result would be the allocation of this function to all components, for no component of the hair drier would

---

how long does it take to open and close the window), then the assumption of equal functionality – at least within a certain range - is a good starting point to compare products and their architectures.

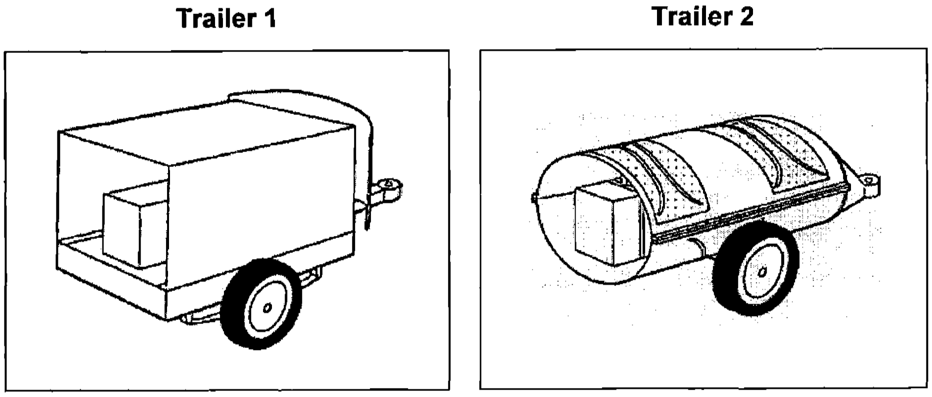
[36] Because mass and weight are closely related to a number of performance dimensions for mechanical products, Ulrich discusses them under the heading product performance. In most cases, the direction of performance improvement is typically clear: smaller is better. (Ulrich calls the corresponding design approach function sharing (for mass reduction) and geometric nesting (for size reduction). (Ulrich 1995, p.433). While in certain cases the exact distribution or mass or size affects the performance in a non-linear way, we restrict the implication to a linear consideration. As such, mass and size become component attributes that and reflect the physical embodiment of the components.

exist in the first place if not contributing to the product's functionality. On the other hand, if the function is chosen too low, e.g., "hold part A in position x relative to part B with force f," then exactly one and only one component delivers this function, i.e., the function tends to become idiosyncratic to a particular design. In other words, if the function description is too detailed, i.e., on a very low level of the function hierarchy, it is likely to predetermine its realization with parts and components.[37] In either of these extremes the function-component allocation schemes would be trivial. In contrast, if one begins to define functions like "generate air flow," "heat air flow," "control heat," "control air flow," and "supply energy," then it becomes meaningful to investigate how they are mapped to parts and components. When in doubt, I recommend to choose a level higher rather than lower in the functional hierarchy, because the higher a function is located in the functional hierarchy the closer it is to an actual user need and perception. Users often care about that a function or feature is provided, but not necessarily how it is provided by certain technical solutions on lower hierarchy levels. Furthermore, the notion of variety – often named to be one of the main drivers of modularity – is tied to customer perceived variety, and not to variety of certain technical specifics.

For the example in this section to demonstrate the method, I will use only the product's technical functions to map the product's functionality to its components. Two products with identical functionality are chosen as examples for the comparative product architecture analysis. This constraint of requiring perfectly identical functionality will be relaxed later. I employ the example proposed by Ulrich, two trailers with different product architectures (Figure 3-2).

---

[37] Kirschman and Fadel, who have developed a function taxonomy for mechanical designs, also recognize the problem that entirely new concepts are unlikely to be found, if only on the lowest functional description level would be searched. Again, the reason is found in the pre-determination of solutions by defining functions too narrowly. (see Kirschman and Fadel 1998, p. 477)



Source: Ulrich (1995) " The role of product architecture in the manufacturing firm"

Figure 3-2: Two trailers with different product architectures

Ulrich identifies trailer 1 as modular and trailer 2 as integral. The method proposed here will add to this and locate both designs in the space of possible product architectures. Ulrich suggests the following six functions as the major functionality both trailers have to deliver:

1. **Protect Cargo from Weather**
2. **Connect to Vehicle**
3. **Minimize Air Drag**
4. **Support Cargo Loads**
5. **Suspend trailer Structure**
6. **Transfer Loads to Road**

Source: Ulrich (1995) " The role of product architecture in the manufacturing firm"

Figure 3-3: Function list for two example product architectures (trailers)

3.4.1.3 Components: What are they?

Just as the previous section asked for a reliable way of determining what is considered a *function,* the framework also needs to provide a repeatable way of determining what is considered a *component.* As stated earlier, this framework considers every unit, i.e., part, subsystem, module, etc. on the determined level of the physical hierarchy as a component. The inner structure or complexity of the components is at this point irrelevant.[38] The major advantage of understanding components in this way is that it does not require a pre-determined definition of what constitutes a *module*. What this also implies, however, is that a function-component allocation scheme on one level says nothing about function-component allocation schemes on lower levels of the hierarchy (within certain *components*) or on higher levels (i.e., what role the product plays as a whole in an upper level system).

In fact, almost all physical assembled products simultaneously display some sort of inner hierarchy and are also part of a larger system and its hierarchy. Others have called this phenomenon *nested hierarchies.*[39] Consequently, then any description of its architecture with respect to the function-component allocation scheme must necessarily be tied to the hierarchical level under consideration. For instance, consider a disc drive as part of a personal computer. Investigating the role of the disc drive for the function-component allocation scheme of the computer will produce a different result than the analysis of the disc drive's own FCA scheme. In sum, FCA schemes have to be defined anew for each level of the hierarchy, and different products can have similar schemes on some levels while quite different on other ones.

3.4.1.4 How to determine the appropriate components

Similar to a product's functionality, a physical product can be decomposed into a hierarchical tree – at least with respect to its assembly. It is for this reason that I suggest to use the assembly information to construct a hierarchical tree that can serve as a guideline to determine what constitutes a component for a particular hierarchy level.

The assembly process and sequence tells what units (e.g., components, sub-assemblies and modules) a product consists of and what the order is in which the pieces have to be put together. The source of this information can be a tear-down analysis in which a product is physically disassembled to understand its structure and parts.[40] Alternatively, design drawings and process descriptions can also be used to determine the assembly structure and sequence. Figure 3-4 shows the lists of components for the two example trailers.

---

[38] The only exceptions are minor parts such as screws, rivets, etc. They are excluded from the analysis.

[39] The notion of 'nested hierarchies' appears in a number of earlier works (e.g., Christensen 1992a, Rosenbloom and Christensen 1994, Gulati and Eppinger 1996, Christensen 1997, Baldwin and Clark 2000).

[40] Ulrich and Pearson, who use the same starting point, i.e., to extract data from physical artifacts, call this approach *product archaeology* (Ulrich and Pearson 1993, Ulrich and Pearson 1998)

Trailer 1	**Trailer 2**
1. Box	1. Upper half
2. Hitch	2. Lower Half
3. Fairing	3. Nose piece
4. Bed	4. Cargo hanging straps
5. Springs	5. Spring slot covers
6. Wheels	6. Wheels

Source: Ulrich (1995) " The role of product architecture in the manufacturing firm"

Figure 3-4: Component list for the two trailers

Additional tools to establishing the component lists are fishbone assembly diagrams. They can help to establish and communicate hierarchy levels based on the assembly sequence. Figure 3-5 illustrates this approach with simplified fishbone diagrams for the two example trailers.

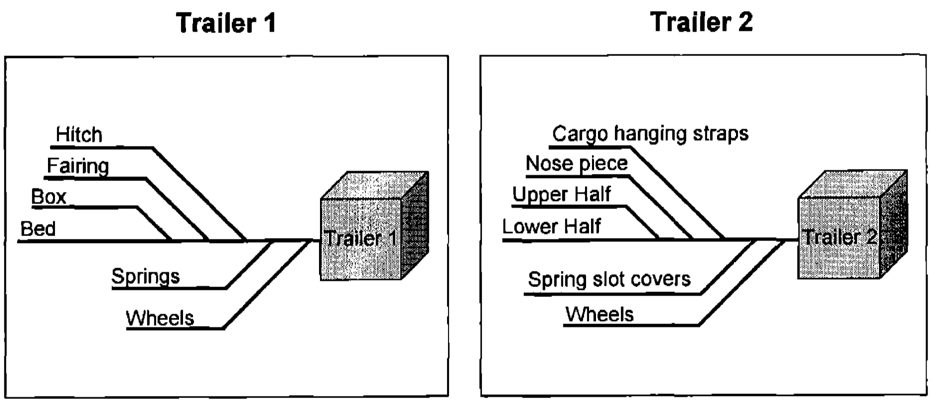
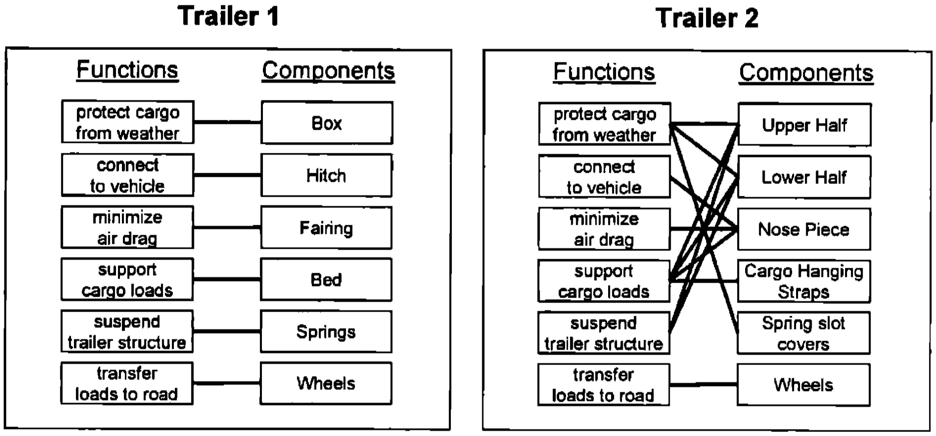


Figure 3-5: Fishbone assembly diagrams for two example trailers

3.4.1.5 Allocation process: How to allocate functions to components

In his work, Ulrich uses the two trailers as extreme examples. Trailer Number 1 he identifies as having a modular architecture and trailer number 2 as having an integral architecture, because the first exhibits a one-to-one relationship for each of the six function-component pairs, while the second demonstrates non-one-to-one relationships (Figure 3-6). Ulrich calls the latter type of relation *complex.*



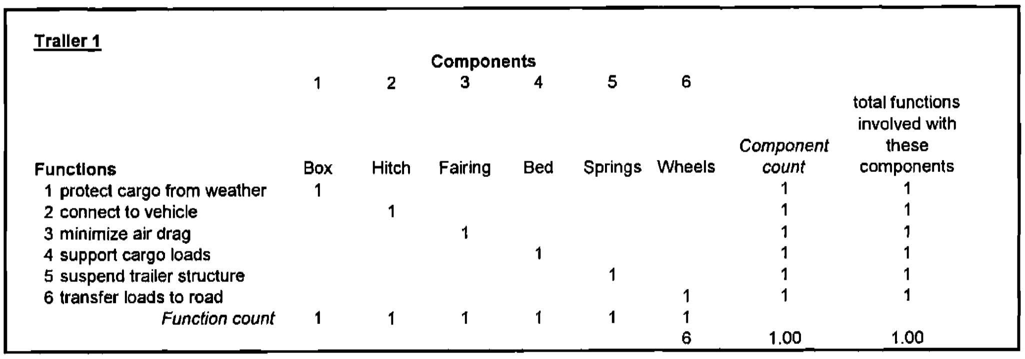
Source: Ulrich (1995) " The role of product architecture in the manufacturing firm"

Figure 3-6: One-to-one and non one-to-one function-component relations

It seems, however, that there are multiple ways in which a product architecture can deviate from the ideal case of having one-to-one relations with respect to its function-component allocation scheme. One major goal of the product architecture description methodology described here is to allow a more precise examination of architectures of this *integral* type.

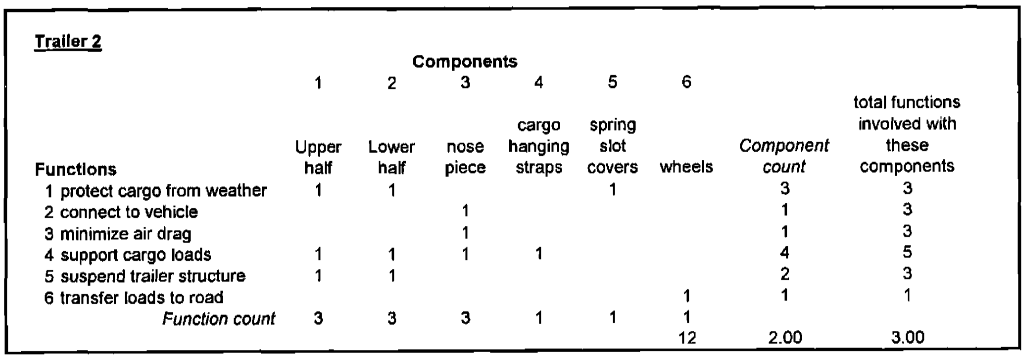
The first step to do this is to construct matrices to locate, visualize, and measure the allocation of the products functionality. The first column contains the function identified earlier, and the first row lists all components identified for the hierarchical level under investigation. For the allocation of the functions to the component, two general options exist concerning a metric. Either a participation of a component to a particular function is indicated in a binary fashion (i.e., yes/no or 1/0), or percentages of a function are allocated to components that contribute to this function. Both approaches have their advantages and disadvantages. Whether or not a component plays any role in a function is usually easier to identify than its proportional contribution. In addition, for a number of product planning considerations such as to offer variety it is often sufficient to know what components are involved to deliver a certain function or feature. On the other hand, for some functions, and attributes like size or weight in particular, it may be worthwhile to assess the contribution that is provided by individual components. One way to produce more detailed data is per interviews of multiple experts in the field (e.g., designers). If the products to be compared are similar in functionality, it is often possible to find agreements on weighting factors other than the binary contribution.

For the example of the trailers, I will only indicate whether or not a component contributes to a function. If it does, the cell in the matrix is filled with a '1,' otherwise its value remains '0.' Figure 3-7 and Figure 3-8 show the two matrices.



Trailer 1			**Components**					
	1	2	3	4	5	6		
Functions	Box	Hitch	Fairing	Bed	Springs	Wheels	*Component count*	total functions involved with these components
1 protect cargo from weather	1						1	1
2 connect to vehicle		1					1	1
3 minimize air drag			1				1	1
4 support cargo loads				1			1	1
5 suspend trailer structure					1		1	1
6 transfer loads to road						1	1	1
Function count	1	1	1	1	1	1		
						6	1.00	1.00

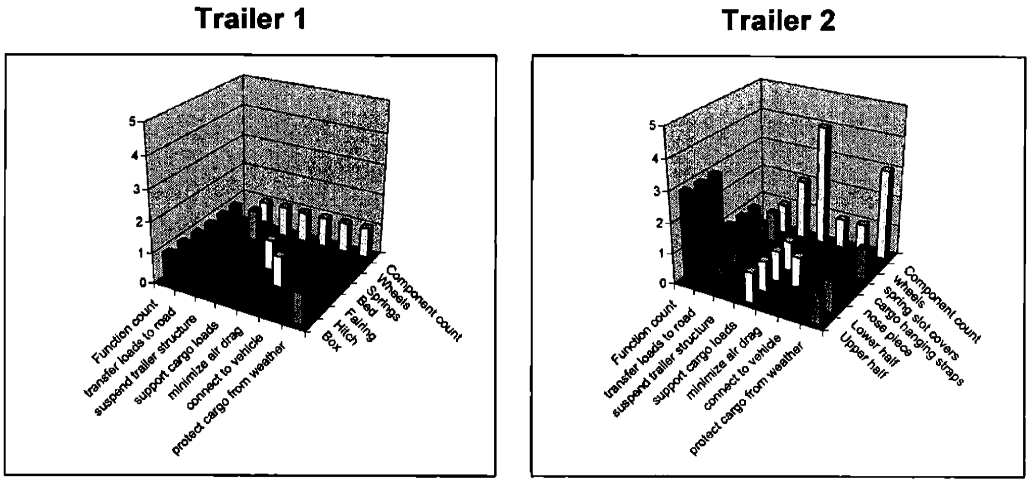
Figure 3-7: Function-component matrix for trailer 1



Trailer 2			**Components**					
	1	2	3	4	5	6		
Functions	Upper half	Lower half	nose piece	cargo hanging straps	spring slot covers	wheels	*Component count*	total functions involved with these components
1 protect cargo from weather	1	1			1		3	3
2 connect to vehicle			1				1	3
3 minimize air drag			1				1	3
4 support cargo loads	1	1	1	1			4	5
5 suspend trailer structure	1	1					2	3
6 transfer loads to road						1	1	1
Function count	3	3	3	1	1	1		
						12	2.00	3.00

Figure 3-8: Function-component matrix for trailer 2

The row named *Function count* at the bottom of the matrix counts the number of functions a component is 'involved in.' Equivalently, the column titled *Component count* to the right of the actual matrix sums the number of components that are 'involved' in delivering a particular function. Three-dimensional views of the two matrices are shown in Figure 3-9. The rows in the back of the graph represent the sums of function per component (dark blue – on the left) and component per function (magenta – on the right).


Trailer 1

Trailer 2

Figure 3-9: 3-dimensional views of function-component matrices

If modularity has something to do with a one-to-one relation between a functional element of a product and one of its physical components, then there are multiple ways for being distant to this 'perfect' state. For one, a component could deliver more than one function. Vice versa, a function could be provided by multiple components. Finally, both cases could occur simultaneously, i.e., each of several components that deliver jointly one function is also involved in other functions. The consequences of the way in which a function is provided are important for product planning purposes. If a function or feature is decided to be variable (e.g., to create a product family), then it is important to understand to what degree this function or feature can be 'isolated' given the FCA scheme of the architecture. In other words, the extent to which a product architecture is modular is an information that should be available for each individual function for it may be that one function exhibits a perfect one-to-one relation while another of the same product shows a many-to-one relation, for example.

To extend the notion of different architectures I argue that an architecture for an entire product is likely to show simultaneously different aspects of different function-component allocations. I further argue that an aggregation often risks loosing the valuable information of the functional level.[41] For these reasons I suggest to map the degree and the direction in which each function deviates from the one-to-one ideal individually. To construct these FCA-maps, two indices are required. The first counts the number of components that participate in providing a particular function (column *component count* in the function-component matrix). The second index takes the set of components just identified for a function and searches for the total number of functions this set of components is involved in (rightmost column in the function-component matrix). These two indices together show the nature of a function with respect to its architectural situation within the product. Mapping the values for each function onto a two-dimensional plane creates the FCA-maps. See Figure 3-10 and Figure 3-11 for the FCA-maps for trailer 1 and trailer 2.[42]

The maps are divided into four regions. Every function can be located in one of these four regions. Functions that are located in the lower left corner region of the map close to the 'ideal' one-to-one relation, I call modular-like. They resemble closely the modular notion, albeit may require two components or are provided together with another function. If a function is provided by a larger set of components, which individually are not involved in other functions, then the function is located in the integral-fragmented region (lower right sector). In contrast, if one component delivers several functions, these functions will be located in the integral-consolidated region of the map (upper left sector).[43] Only if multiple components provide multiple functions in such a way that most (every) component(s) participate at most (every) function(s), the functions would be located in the integral-complex region of the map (upper right sector).

---

[41] In addition, a weighting scheme would be required to weight not only the functions against each other but also the ways in which they deviate from the 'modularity ideal' of exhibiting a perfect one-to-one relation between function and component.

[42] The FCA maps are read as follows: First, the horizontal axis indicates for each function the number of components involved in its provision. Next, the vertical axis indicates the total number of functions that is provided by the set of components just identified.

[43] Designs or architectures that use a strategy *function-sharing* exhibit a function concentration in this sector (integral-consolidated).

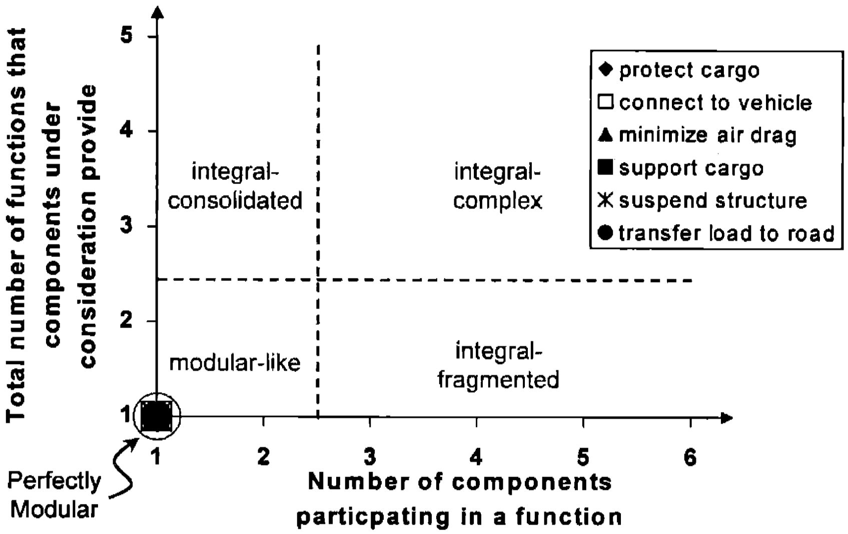


Figure 3-10: FCA-map for trailer 1

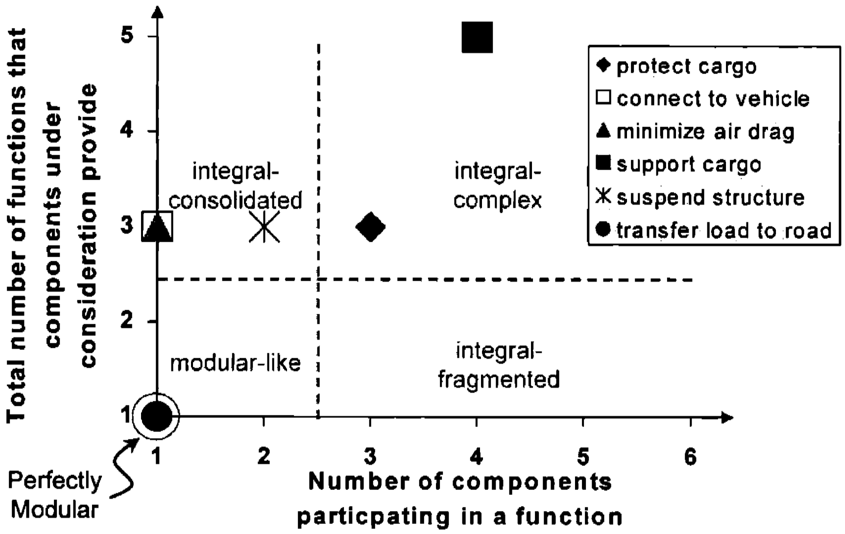


Figure 3-11: FCA-map for trailer 2

The architecture of the entire product is then represented by the pattern of all functions. What this mapping tool allows is to display how various architectures differ from each other with respect to the idea of exhibiting one-to-one relations between functions and components. For example, in case of trailer 1 all functions can be found in the lower left corner. It exhibits a perfectly modular function-component allocation scheme. In contrast, while trailer 2 has complex relations between functions and components, it does so to varying degrees. The function *transfer load to the road,* for instance, is exactly on the same modularity-level as it is for trailer 1. Conversely, the functions *minimize air drag* and *connect to vehicle* are entirely provided by the same component. This component is also involved in providing *support cargo loads.* Each position makes it easier or harder to isolate a particular function for variety, upgrade, or one of the many other reasons often associated with modularity.[44]

3.4.2 Interfaces and their role in the product: Three Characteristics

Once the elements and their functions of a system have been described, the next task is to specify the second major dimension of the product architecture: the relations between the elements, i.e., their interfaces. Ulrich uses the term *coupling* to describe the interfaces and focuses on the interfaces' effect on the design process: "Two components are coupled if a change to one component requires a change to the other component in order for the overall product to work correctly." (Ulrich 1995, p.423) He departs from the notion of having a physical connection in order to qualify for being an interface: ".. coupling may also be based on other physical phenomena, such as heat or magnetism." (p.424)

Similar to the dimension *elements,* Ulrich's work serves as a stepping stone to extend the work. Based on the findings of the previous chapter, an interface plays different roles during various stages of the product's life. I group the information into three categories of interface characteristics: (1) the interfaces' role for the product function, (2) their role for making, changing, and unmaking of the product, and (3) their role with regard to substitutes.

---

[44] The framework proposed here points also to an explanation for another phenomenon: the development of product architectures over time. It has been voiced that more and more products become modular over time. In contrast, I conjecture, that most products or systems are born in the integral-fragmented mode and become more integrated over time, and only some functions move towards higher level of modularity while others do not. To be more specific: if one controls for functionality, i.e., keeps a product's functionality constant, products begin their life in the *integral-fragmented* region, the area in the lower right of the FCA-maps. In the beginning, products or systems are often only crudely defined and consist of many different parts to make them work at all. Later, once the technology becomes better understood, its definitions become clearer, and products and process get improved. As a consequence, over time the design moves to architectures with fewer parts (parts integration), i.e., towards the left side of the map. Only those functions where variety becomes important, they move to architectures where they can be better 'isolated,' i.e., they move to the lower left corner, the *modular-like* region. Simultaneously in the same architecture, however, those functions that are perceived as stable across customer segments and/or over time, move to higher levels of *integral-consolidation,* i.e., they move to the upper left of the map.

3.4.2.1 Number and nature of the interfaces

The interfaces' roles for the product function are determined by their number and their nature. Obviously, the simple number correlates to some extent with the number of components the product consists of (at the hierarchy level under consideration). If the number of components is n, then the number of interfaces is at least n-1 and at most (n*(n-1))/2; the former would be a string of components, the latter a web of connections where every components forms an interface with every other component.[45] Taking our examples, trailer 1 has 5 interfaces out of 15 possible, while trailer 2 exhibits 6 out of 15 possible interfaces (see Figure 3-12 and Figure 3-13).

In addition to the number, the distribution of the interfaces provides additional information about the product architecture. For example, if a component interacts with more than two other components (see for example the component *upper half* of trailer 2), there are two possible explanations. First, the component may play a pivotal role in the product architecture, for example, as a central component with high internal complexity, as one that supplies general or central functionality. This resembles what most researchers understand as a platform.[46] Second, the product architecture under consideration may not be very modular, but rather fragmented. It is more likely for a fragmented product architecture than a modular one to have many components that show interactions with many other components. The example trailer 1 exhibits more a stack-like architecture, whereas trailer 2 has the upper half as its central element.

Depending on the functionality of the components participating in the interface under consideration, the interfaces also vary in their nature. Nature of an interface is here comprised of its category and its intensity. To determine an interface's category and intensity I build on work by Pimmler and Eppinger. They have developed a methodology for the analysis of product design decomposition (Pimmler and Eppinger 1994).[47] After decomposing the system into

---

[45] This rather simple calculation assumes that between two components only one connection or interface exists. This abstracts multiple or multidirectional connections into a yes/no relationship.

[46] The platform concept has been discussed at length in Meyer and Lehnerd 1997, Robertson and Ulrich 1998, or Sanderson and Uzumeri 1995.

[47] Pimmler and Eppinger's methodology helps to describe the interactions between components and allows to guide improvements in design and team organization. In contrast to Pimmler and Eppinger, who propose their methodology to improve the design by rearranging units such that they reduce the number of off-diagonal interactions, the methodology suggested here uses the matrices for descriptive purposes only. This also allows using only the upper half of the matrix. The lower half is reserved to determine the interfaces' levels of reversibility (see next section).

elements, the interactions between these elements are documented and coded in a matrix. They suggest the consideration of four categories of interactions: (1) A spatial interaction identifies needs for adjacency or orientation between two elements, (2) an energy interaction identifies needs for energy transfer between two elements, (3) an information interaction identifies needs for information or signal exchange between two elements, and (4) a material interaction identifies needs for materials exchange between two elements. They further suggest specifying the importance and desirability, i.e., the intensity, of the interaction on a five-point scale from –2 to +2. The identification and scoring of each interaction for each of the four types results in a matrix that has in each interaction field four numbers between –2 and +2. The upper left corner has the number for the spatial-type interaction, the upper right the number for energy-type interactions, the lower left corner shows the number for information-type interactions, and the lower right corner displays the rating for material-type interactions.[48] Returning to the two example trailers, the upper triangles in Figure 3-12 and Figure 3-13 display the number, category and intensity of their interfaces. The empty cells can be interpreted as being filled with zeros.

Due to their mechanical characteristics, most of both trailers' interfaces are spatial in nature. Only exception is the energy and signal transmission from the towing vehicle to the rear lights. Note that despite their very different FCA schemes, both architectures exhibit a number of similarities with respect to the nature, i.e., category and intensity, of their interfaces.

---

[48] Other researchers have suggested developing coupling analysis tools. Johanneson, for example, uses Pimmler and Eppinger's interaction specification scheme to develop a method in which interaction and coupling information can be treated in the axiomatic design world. (Johannesson 1996, Johannesson 1997)

TRAILER 1

	1 Box	2 Hitch	3 Fairing	4 Bed	5 Spring	6 Wheels
1 Box		2 1 1 0		2 0 0 0		
2 Hitch	3 1		2 0 0 0			
3 Fairing		2 1				
4 Bed	2 1				2 0 0 0	
5 Spring				2 2		2 0 0 0
6 Wheels					2 2	

NATURE of Interfaces (upper triangle):
(adapted from Pimmler and Eppinger 1994)

Category:			
Spatial	S	E	Energy
Information	I	M	Materials

Intensity	
Required	2
Desired	1
Indifferent	0
Undesired	-1
Detrimental	-2

REVERSIBILITY of Interfaces (lower triangle)

Effort 1
Depth 1

Effort to reverse		*Depth in which interface is located*	
easy	1	shallow	1
medium	2	medium	2
difficult	3	deep	3

NUMBER of Interfaces

theoretical max	15
theoretical min	5
real	5

Figure 3-12: Interface matrix for trailer 1

TRAILER 2

	1 Upper Half	2 Lower Half	3 Nose Piece	4 Cargo Hang Straps	5 Spring Slot Covers	6 Wheels
1 Upper Half		2 0 0 0	2 0 0 0	2 0 0 0	2 0 0 0	
2 Lower Half	2 2		2 1 1 0			2 0 0 0
3 Nose Piece	2 1	2 1				
4 Cargo Hang Straps	1 1					
5 Spring Slot Covers	2 2					
6 Wheels		2 2				

NATURE of Interfaces (upper triangle):
(adapted from Pimmler and Eppinger 1994)

Category:			
Spatial	S	E	Energy
Information	I	M	Materials

Intensity	
Required	2
Desired	1
Indifferent	0
Undesired	-1
Detrimental	-2

REVERSIBILITY of Interfaces (lower triangle)

Effort 1
Depth 1

Effort to reverse		*Depth in which interface is located*	
easy	1	shallow	1
medium	2	medium	2
difficult	3	deep	3

NUMBER of Interfaces

theoretical max	15
theoretical min	5
real	6

Figure 3-13: Interface matrix for trailer 2

Finally, for comparative purposes the results of the first step of the interface analysis can be aggregated into a measure of interface intensity.[49] Spatial interfaces typically define stricter constraints to an interface than material exchange. Material exchange is likely to put tighter constraints on the interface than energy exchange, and energy exchange is probably more restrictive than signal exchange. Following this logic, rank ordering the four different categories of interfaces with decreasing 'comparative intensity' results in the following order: spatial>material>energy>signal. This information together with the specific numerical assessment of the interfaces, allows a qualitative comparative assessment of the interfaces. Corresponding to the FCA-maps, the data is aggregated for each function. For example, for both trailers, the majority of the interfaces between the major components is assessed as being spatial in nature and assigned a relatively high level of intensity (i.e., 2). This translates in relatively high levels of interface intensity for both designs.

3.4.2.2 Reversibility

The second interface attribute that needs to be analyzed is the interfaces' reversibility. The notion of various product changes such as upgrades, add-ons, adaptation, wear, consumption, or reuse - which are often assumed as being one of the major advantages of *modularity* - strongly depends on the reversibility of the interface.

The effort to reverse, or disconnect, the interface can serve as a proxy to determine the reversibility of an interface. This effort depends on two factors: first, the difficulty to physically disconnect the interface which is determined by the technology used to establish the interface, and second, the interface's position in the overall product architecture.

Theoretically, every interface can be disconnected. The idea, however, that modular product architectures have strong interactions within modules and weak between them, carries to some extent the meaning that the weakness of these relations can be translated into low effort to reverse or disconnect the interface. The repair of an outer panel of an automobile door illustrates that the answer to this second question lies on a continuum. Consider a standard car door as it is built today. The steel structure is welded together before it is painted jointly with the car's body-

---

[49] The purpose of this step is to transfer the most important results from the interface analysis regarding nature and intensity into the product architecture map. Since this aggregation necessarily eliminates some data, it is not meant to replace the foregoing analysis. For a deeper understanding of the interface characteristics the matrices developed above are still required.

in-white. Finally, all assembled parts are attached to the door. Theoretically, it is possible to remove all assembled parts, to cut out the damaged door outer panel, to weld in a new outer panel, to repaint the door, and to re-assemble its components. Contrast this with a door architecture, where the outer panel is not part of the load carrying structure and not welded to it, but attached with a reversible mechanism (e.g., screws). Here the door outer panel could be replaced without first removing many other parts. In addition, the reversible attachment mechanism reduces the level of skills and specialized equipment required to remove the old part and to attach the new one. Consequently, the overall effort to disconnect - or reverse - the interface between the outer panel and the rest of the door is lower for the alternative design than the standard steel design.

As this example demonstrates, the level of an interface's reversibility depends not only on its own characteristics (skill and equipment requirement) but also on its position within the overall product, i.e., how deep it is 'buried' in the product.[50]

For the purpose of comparing product architectures and their interfaces, I suggest assigning two values to each interface. First, the difficulty to disconnect the interface is assessed with a score from 1 (=easy) to 3 (=difficult). For example, a snap-fit connection can be considered easy, a bolt-nut connection medium, and a welded connection difficult. Second, the depth in which an interface is 'buried,' i.e., how much other units have to be disassembled or removed before a disconnection of the interface is possible, is assessed with a value between 1 (=shallow) to 3 (=deep). These values should be assigned with the comparative purpose in mind. The results for the two example trailers are shown in the lower triangles of Figure 3-12 and Figure 3-13.

Like the interface characteristics *intensity*, the characteristic *reversibility* can be aggregated for qualitative comparisons. With respect to the difficulty to reverse most of the interfaces, the two trailer concepts are very similar. For both architectures the difficulty can be considered to be relatively low, primarily because of the use of mechanical fasteners (only bed and hitch of trailer 1 are assumed to be welded together). Due to the relatively small number of components

---

[50] Kirchain, who has modeled the end-of-life recycling of automotive structures, uses the term *buoyancy* to describe the difference between the value of a part and the cost to reclaim it via disassembly (Kirchain 1999). The buoyancy is a function of how many other parts have to be removed *before* the part under consideration can be retrieved.

none of the designs exhibits significant 'depth' in which some of the interfaces could be 'buried.' Note, however, that a component like the *Upper Half* of trailer 2 has an above average number of interfaces. If this component needed to be replaced, it would require disconnecting all those interfaces.

Again, the aggregation of this information should follow the function or feature, since this is the driving force behind disconnecting an interface (except disassembly for material recycling).

3.4.2.3 Type (Standardization)

Finally, interfaces can be of different types with respect to substitutes and product families. These different interface types deserve particular consideration, because they determine most of the options one may gain through the use of interchangeable subsystems. Some researchers have used different types of interfaces to categorize different architectures. Ulrich and Tung, for example, have suggested five different categories: component swapping, component sharing, fabricate-to-fit, bus, and sectional (Ulrich and Tung 1991).[51] Ulrich later collapses the first three types into one and calls it slot architecture (Ulrich 1995). Likewise, Ulrich and Eppinger use the types slot, bus and sectional modular architectures (Ulrich and Eppinger 2000, p.184). Building on Ulrich and Tung's list, Pine renames the type fabricate-to-fit into cut-to-fit and adds the type mix modularity; the latter to extend the systematic to include processed products (Pine 1993, p.201). Other researchers have shifted their focus from describing the type of the interface to describing the type of process used to change the architecture. Baldwin and Clark propose 'operators' to describe the evolution from one design (product architecture I) to the next design (product architecture II). They suggest six 'modular operators:' *splitting* a design into modules, *substituting* one module design for another, *augmenting* – adding a new module to the system, *excluding* a module from the system, inverting to create new design rules, and *porting* a module to another system (Baldwin and Clark 2000).[52] The above definitions use different viewpoints

---

[51] See Ulrich and Tung 1991, p.10. Ulrich and Tung use the term 'modularity' to describe the different types. To consider that a product architecture can have simultaneously several different interfaces, they use the term 'hybrid architecture' (p. 8). However, since the ways modules interact which each other is in most cases a characteristic of the component (module) and the component it is connected to, I prefer the term *interface*.

[52] The modular operator *excluding* implicitly assumes that the system can function without the module that is removed by this action; in other words, the excluded module cannot be of fundamental necessity for the basic system function.

(the first focus on architectures while the operators describe activities) but they both implicitly describe the interface types. Figure 3-14 summarizes and compares the various definitions.

If interface is understood as a description of the interaction between sub-units of a product, it becomes apparent that the type of interface allows or limits the interaction between the participating units. A common idea of modularity is the relative ease with which an intermodular interface allows an exchange of sub-units, i.e., a change in the product configuration. This 'ease' has two components. The first is the interface's reversibility (see previous section). The second is the degree to which there are alternatives for an exchange. The latter one is subject of this section.

Author(s) **Interface Type**	**Ulrich & Tung 1991**	**Pine 1993**	**Ulrich 1995, Ulrich & Eppinger 2000**	**Baldwin & Clark 2000**
One-sided standardization	Component Swapping Modularity	Component Swapping Modularity	Slot-Modular Architecture	Substituting / Augmenting / Excluding
One-sided standardization	Component Sharing Modularity	Component Sharing Modularity		Porting
One-sided standardization	Fabricate-to-Fit Modularity	Cut-to-Fit Modularity		Substituting / Augmenting / Excluding
One-sided, but product wide standardization	Bus Modularity	Bus Modularity	Bus-Modular Architecture	
Two-sided and product wide standardization	Sectional Modularity	Sectional Modularity	Sectional-Modular Architecture	
n.a.	*n.a.*	*Mix Modularity*	*n.a.*	*n.a.*
n.a.				*Splitting*
n.a.				*Inverting*

Figure 3-14: Modularity, Interfaces, and Modular Operators[53]

The extent to which an interface allows this interchangeability of components is a matter of the perspective. For example, what has been called *component swapping modularity* and *component sharing modularity* can be the same thing, depending on the alternatives that exist on either side of the interface. If one module is defined as the one remaining in the system (often the larger one) and the interface allows the exchange of the other one, then the term component swapping is used. If the larger one is exchanged (i.e., the reference switches to the smaller one) the term component sharing is used. Note that the technical characteristics of the interface are identical in both cases. Thus, the use of these terms depends on which component is chosen as the reference system. Consider the interface between a lamp and a light bulb. If one changes to a light bulb with a different color and keeps the same lamp, it can be called component swapping.[54] If the same light bulb is used in a different lamp, however, it is called component sharing. And if we use a light bulb with a different watt number, we could call it fabricate-to-fit to express a type of scaleability.[55]

If the chosen reference system offers similar interfaces in various locations, the term bus modularity has been applied. This implies that a component considered to be the bus allows other units 'to hook on' at various places; just as a bus allows passengers to travel only sections of the whole bus tour according to the individuals' preferences. Examples are electrical systems, shelving systems with rails and the bus systems used in computers. If the standardization is taken one step further, it allows the connection of every unit with every other unit. This is what is meant by sectional modularity. An example are the LEGO blocks, others include piping fittings or sectional sofas.

These examples demonstrate two insights. First, they support the separate determination of function-component allocation and interface characteristics. For example, identical interfaces can be found in different function-component allocations, or a single function-component combination can exhibit different interface types. Therefore, the described features are characteristic of particular interfaces rather than the architecture as a whole. Second, the extent to which the interface can be considered standardized from each component's perspective is key to understand each interface's contribution to the product architecture and its role in a product

---

[53] Two of the modular operators that Baldwin and Clark suggest, *splitting* and *inversion,* are actually steps in changing the architecture fundamentally, and therefore do not implicitly deal with a certain type of interface.

[54] Here the lamp becomes the *product platform* and the colored light bulbs vary the overall product configuration. See also footnote 46 for the issue of product platforms.

[55] The terms *fabricate-to-fit* or *cut-to-fit* have been defined by their authors also to include the scaling of otherwise identical goods, for example, apparels. A shirt in different sizes is made out of similar components (same cloth, same design, same cut, but different size) and the interfaces are technically identical (seams), but located at different positions.

family.[56] Figure 3-15 suggests a mapping tool for different interface types. Each location on this map describes an interface from the perspectives of both elements (or components) involved. One extreme, the lower left corner indicates interfaces where there are very few alternatives to replace or substitute the components. The other extreme, the upper right corner, locates interfaces where multiple replacement or substitution options exist for both components. All other cases are combinations of these extremes.

The four circles with number refer to examples in the text above. Number 1 represents the lamp/light bulb combination. An example for number 2 can be imagined as a separate electrical system with a unique voltage (e.g., boat or remote house). All appliances can be used anywhere in this system, but only in this system. Number 3 stands for unique components on both sides of the interface (e.g., in a space station), and number 4 reflects the idea of the LEGO blocks, i.e., every component has multiple alternatives it could be replaced with, and it could replace.

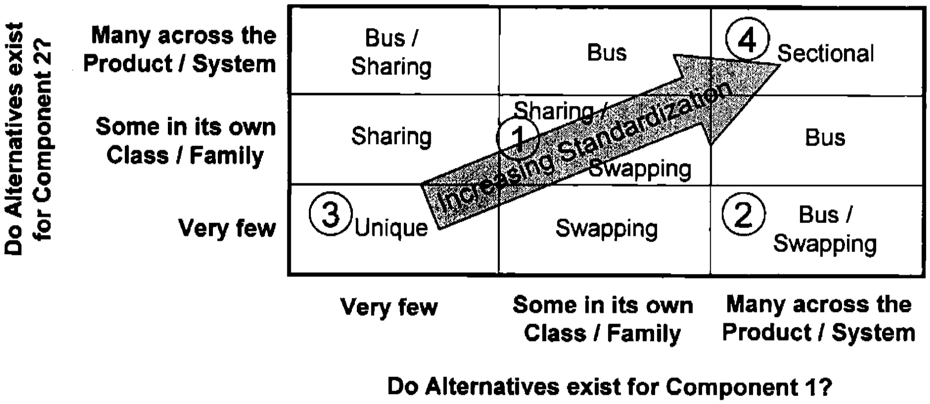


Figure 3-15: Interface types as a function of their level of standardization

Using this framework, the interface characteristic *type* can be investigated in more detail. Figure 3-16 shows the interfaces for the major functions of the two trailer examples. Each icon

---

[56] Obviously, the nature of the system and the definition of its boundaries also determine the population size of possible substitutes. A space station may be the only one ever produced and its entire class exhibits only one member. In contrast, mass produced products, by definition, have a considerable number of identical members in its class.

characterizes the alternatives on each side of an interface per function.[57] Not knowing the product families that could surround these example trailers, an educated guess is used here to illustrate the mapping method. The analysis demonstrates that regarding their degree of interchangeability, the functions of trailer 1 exhibit a higher level of standardization than those of trailer 2. Reason for this is that it is more likely for those components that carry the functions to find use in other designs for trailer 1 than it is for trailer 2, which show a higher level of idiosyncrasy. However, the level of standardization is not homogenous across either architecture. For example, while trailer 2's function *support cargo loads* is very unique, its function *minimize air drag* providing components can be pictured as being also usable for a similar but trailer trailer, and its function *transfer loads to road* is exactly as standardized as the one of trailer 1. In other words, the level of *modularity* can vary across a single product architecture.

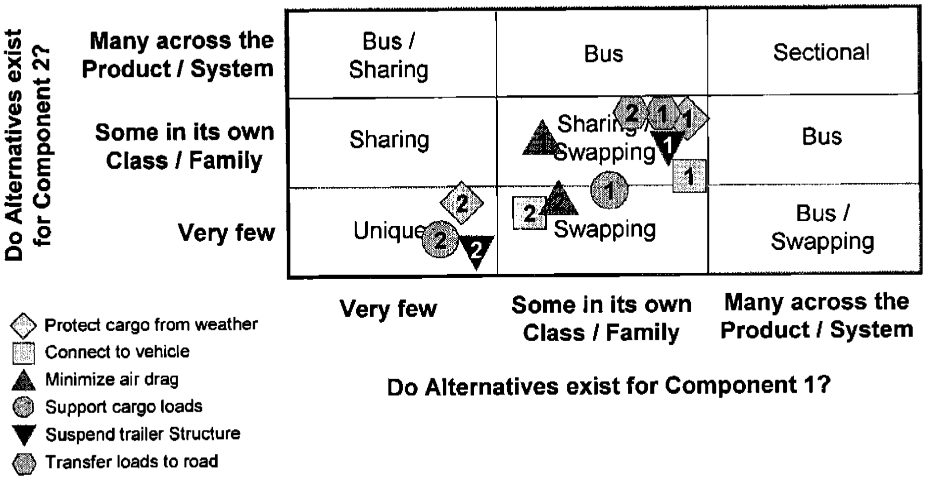


Figure 3-16: Interface types of two trailer examples

Similarly to the two previous interface characteristics, the characteristic *type* can be qualitatively displayed in an aggregated manner for each function in the product architecture

---

[57] Technically, this way of displaying constitutes an aggregation of all interfaces of those components that are involved in providing a particular function. I consider this aggregation reasonable because it is the interchangeability of a function or feature that is of interest rather than the component itself (in case of a perfect one-to-one relation between function and component they become identical).

map. If level of standardization is taken as the aggregate metric, the values for the functions of trailer 1 are medium while for trailer 2 most the standardization level for most functions is rather low, for some (e.g., transfer loads to road) on the same level as trailer 1's.

3.4.3 Pulling it all together: the product architecture map

Together with the function-component allocation data, the interface information completes the description of the product architecture. Consequently, adding the information for all three interface dimensions to the function-component allocation map results in the product architecture map. Figure 3-17 illustrates how the interface information in the trailer example can be added in the third dimension to the information about function-component allocation.

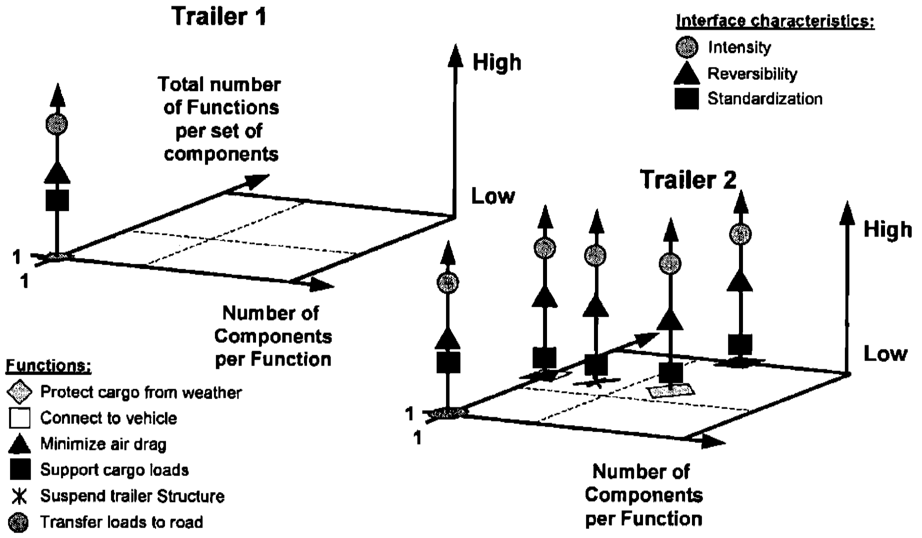


Figure 3-17: Product architecture maps for both example trailers

These product architecture maps serve as a graphic representation of the complete product architecture description. They allow quick visual references of similarities and differences of the analyzed product architectures. They do so on a level with much finer granularity than the concept *modularity* could do.

In the next two chapters I will build the second half of the foundation, i.e., the costing framework. After that follows an extended case study applying the product architecture description methodology to investigate the economic consequences of individual product architecture differences.

4. FRAMING OF COST ANALYSES

4.1 Chapter Introduction

The task to assess the cost implications of modularity, i.e., product architecture differences, faces two main difficulties. The first difficulty is to develop an understanding of what constitutes modularity and how to measure it. Chapters 2 and 3 serve this purpose. The second difficulty is to identify the costs that should be included in the analysis (and which to be excluded) and to be aware of how this framing of the analysis affects the results. There are multiple ways in which analysis boundaries are set, and each one of them has wide ranging implications for the results.

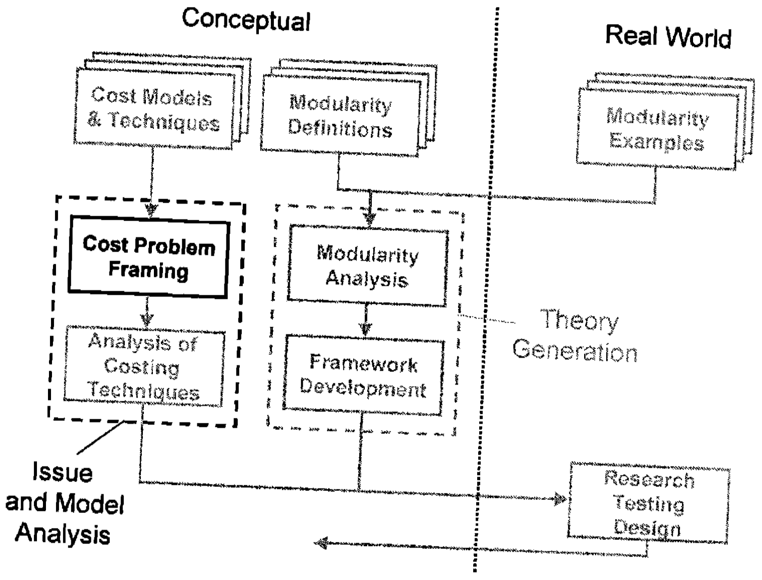


Figure 4-1: Content of this chapter in relation to the other chapters

This chapter presents an overview of how costs of physical products occur and how they can be interpreted. To this end, it also provides an overview of the relevant cost related literature. This chapter is structured into three parts. Each of the three parts covers a major dimension along which the boundary setting, i.e., the framing of the cost analysis, must be understood to be able to interpret the results. The three dimensions are concerned with boundary settings regarding *time*, *cost types*, and *uncertainty*.

First, boundary settings regarding time determine which phase of a product's life will be included in the cost analysis. This affects not only what costs will be included, but also when they occur and – at least indirectly – who has to bear them.

Second, boundaries have to be set around the unit of analysis. Similarly to determining a product's content by defining its boundaries, boundary settings regarding what cost types are included in the analysis or excluded from it can have a strong impact on the resulting costs. The growing importance of the questions about direct versus indirect costs for modern, highly automated industrial production processes indicates the significance of this type of boundary setting problem.

Third, any cost calculation makes assumptions on how closely its results will represent reality. Two general cases of how uncertainty is considered are discussed. One, is the cost analysis assuming a static nature or does it incorporate dynamic variations? Two, what kind of cost behavior is assumed? Is it viewed as deterministic or is the stochastic nature of their behavior considered?

In order to lay a foundation for the analysis of various cost monitoring and estimating techniques for the purpose of analyzing costs of different product architectures (see chapter 5), the three groups of boundary setting questions around time, cost types, and uncertainty are discussed in detail below. The discussion is focused on the impact product architectures difference have on costs.

4.2 Setting the Analysis Boundaries with respect to Time

If costs of different technologies are compared, the choices of the unit of analysis and the choice of the time frame are often not entirely independent from each other. This section's discussion of the boundary setting regarding time focuses on the product as the unit of analysis (section 4.3 will take a detailed look at the implications of the unit-of-analysis-choice), because this dissertation ultimately compares products, i.e., similar products with different architectures. For this reason, a product-based life cycle model serves as a guide for the following discussion.[58]

---

[58] If the unit of analysis is, for example, a process technology, as opposed to a product, the life cycle model is more difficult to apply.

4.2.1 Life cycle costs

Every product, every system, regardless of its size, runs through different phases in its 'life.' Generally, these phases can be grouped into design, production, use, and disposal (Blanchard and Fabrycky 1998).

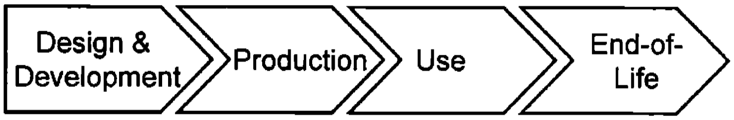


Figure 4-2: Phases of a product life cycle

In all life-cycle phases a number of different processes or activities are performed to 'do something with the product.' Each of these processes and activities creates a cost (Figure 4-3). These costs can occur at different times, at different locations, and can be borne by different constituents.

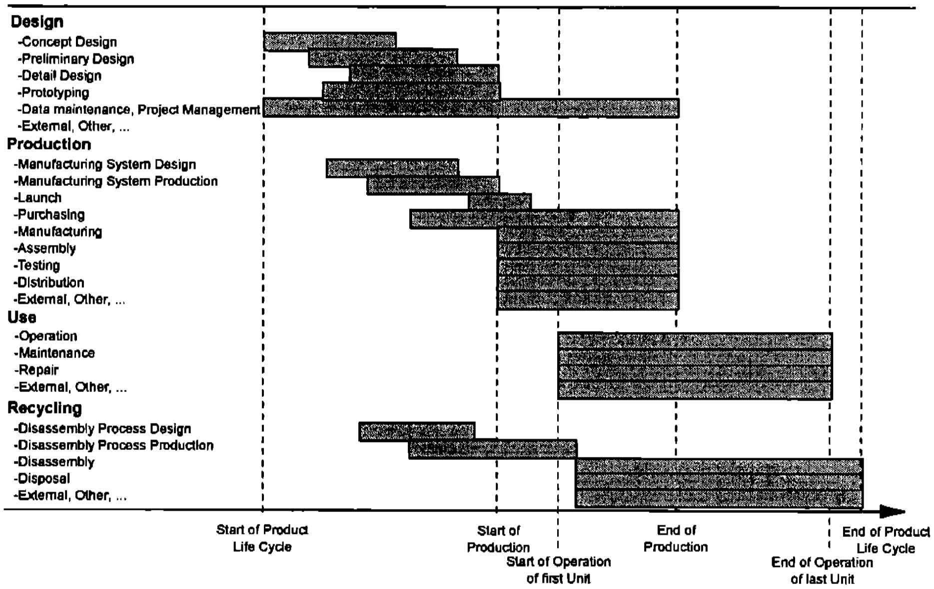


Figure 4-3: Various activities create costs throughout a product's life cycle[59]

---

[59] There are multiple ways to define product life cycle. This diagram describes the life cycle of *all units* produced during a model's life. In case of only one unit produced (e.g., expensive and special equipment), the diagram collapses to the individual's product life cycle. It needs to be emphasized that in the literature – and in the

Given that various costs occur at different phases during a product's life, the question then is to decide which costs are included and which are excluded in the cost analysis. To answer this question, two pieces of information are important: the life cycle cost profile and the ownership of the costs.

First, what is the cost profile over a product's life cycle? In other words, what are the absolute values and what is the relative distribution of the costs and durations of the individual phases? Obviously, a product's life cycle cost profile strongly depends on the length of the product's life as well as the product's value. It has been suggested to separate the universe of different products into three major categories: large-scale, medium-scale and small-scale systems (Asiedu and Gu 1998). This separates systems and products according to the absolute values of total lifetime and total life cycle cost. Large-scale systems can have total lifetimes of several decades and total life cycle costs of billions of dollars. Lifetime frames of medium-scale products are typically measured in years, with total life cycle costs ranging from thousands to millions of dollars. Small-scale products can have lifetimes as short as a few months and life cycle costs as low as tens of thousands of dollars (Figure 4-4).[60]

---

remaining discussion of this work - the term *product life cycle* is used to describe the life of a single product – similar to an individual of, say, an animal population. In other words, the individual product is the unit of analysis. This is to be distinguished from another use of the same term when describing a life cycle of a product as a concept, which occurs, for example, in the discussion about the concept of dominant designs. There the life cycle is described in phases through which the concept emerges, solidifies and stabilizes. Depending on the life span of an individual product (and the population size) this may or may not coincide with the life cycle of an individual product.

[60] The length of the product's life carries an additional issue for cost calculations. For systems or products with long life times (often large scale), the question of the time value of money alone becomes very important. More than for small and medium sized products, the assessments of large-scale products are usually very sensitive to both the shape of the cost distribution over time and the chosen interest rate. Section 4.2.3 will discuss this question further.

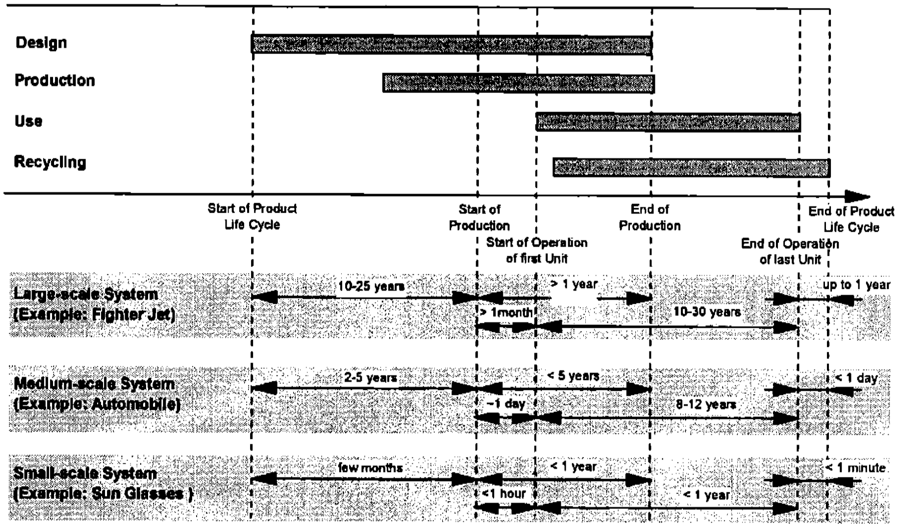


Figure 4-4: Different products/systems with different time scales

In addition to the absolute values, the relative distribution of time and cost over the different life phases and their sub-processes plays an important role, too. For example, small products, say a radio clock, will exhibit very small maintenance and support costs during its use, whereas for long living and large scale products as, for instance, a navy ship, these costs can significantly exceed the initial purchasing price over its lifetime (Sands et al. 1998).[61] Following the original goal of this study of understanding how architectural differences of physical, assembled products affect costs, the discussion through this chapter centers on medium-sized products and presents examples where product features influence costs.[62]

Second, the question of when certain costs occur does not necessarily determine who has to bear them. For example, warranty policies can transfer cost between producer and user, or most

---

[61] In their paper discussing Navy equipment standardization initiatives, Sands et al. state: "The initial development and procurement costs of an repairable (maintenance significant) end item typically comprises only about 36% of the total ownership cost (TOC) with the remaining 64% accrued during the operational and support phase of the item." (Sands et al. 1998, p. 117)

[62] There are, of course, numerous other costs that can occur. For instance, cost for sales and marketing can contribute significantly to the total product costs. So-called below-the-line marketing costs (discounts and incentives, trade sales etc.) in the car industry sometimes amount to similar numbers as the production costs (Brown 2001). For example, incentives alone were at $2,600 per car at Chrysler in January 2001 (Hyde 2001). However, these costs are not related directly to different architectures, but rather tactical marketing decisions. In contrast, the analysis presented here is structured around those costs that are related to architectural features. Nevertheless, the problems of allocating several types of indirect costs will be discussed in-depth in section 4.3.

of so-called external costs are often borne by the society at large while the user pays only a fraction of it directly. More generally, depending on a variety of additional factors such as market dynamics, level of competition, or institutional environment, a number of different cost allocation schemes are conceivable, enforced by different contractual agreements. For these reasons, and to focus on the cost effects caused by differences in product architecture, the following discussion will look at costs independently from the ultimate ownership.

4.2.2 Cost occurrences in individual product life cycle phases

This section discusses the various types of costs that occur during the individual phases of the product's life. It focuses on when these costs occur and the extent to which these costs can be influenced in the design stage, i.e., by a choice of certain product architecture features over others. Over the last two decades various cost optimization techniques have been proposed to allow design to influence the costs in different downstream activities (e.g., DFX). Each of these techniques focuses on a subset of the total costs. Therefore, the relevance of each technique hinges on the relevance of the specific cost block within the total cost picture and the particular perspective taken.[63] In contrast to these 'optimization' tools, the discussion below will follow a product through its entire life and will discuss different costs associated with each phase. With the multiple dimensions of modularity in mind, the discussion will address the extent to which each cost block is affected by (a) the extent to which function and components are aligned, and (b) various interface characteristics. While the focus of the discussion is on cost effects, other factors influenced by the architecture, like time and revenue, are discussed where appropriate.[64]

---

[63] In other words, if, for instance, the vast majority of the total costs incurs in the design phase, a design-for-reparability approach may not be the best use of the resources. Or, whether a design is 'optimized' for testability may only be relevant for a final customer if this has direct implications for her costs or the product quality.

[64] The factors *time* and *revenue* are not independent from each other. For example, increasingly shorter product life cycles make the timing of a product's market introduction ever more profit critical because the market entry determines the length of the remaining sales period and, therefore, the sales volume. In addition, it affects the profit margin per unit, which tends to decrease once competition sets in. For that reason, companies in many industries have introduced programs to reduce the development time. DaimlerChrysler's fastest product development project took recently 28 months, compared to 39 months in 1993 (Anonymous 2000b).

4.2.2.1 Design and development phase

The first phase of a product's life, here called design phase, includes all work required from the initial idea to the product launch. This encompasses conceptual and preliminary design, detail design and prototyping, testing, as well as supporting functions such as data maintenance and project management. Some of these costs actually occur also after the product launch (e.g., data maintenance), although it could be argued that other processes or activities further downstream are actually creating these costs. In general, the question is how are the costs for all activities and processes within the design phase affected by a product's architectural characteristics? For the processes further upstream, such as research and advanced development, this question becomes increasingly difficult to answer, mainly due to the problem of traceability of these costs. Many research activities can often not easily be assigned to a specific product (section 4.3 will discuss this allocation problem more in depth). Relative to costs for research activities, the costs due to resource consumption through design processes, from conceptual to detail design to prototyping, are somewhat easier to allocate to individual products.

Typically, product development performance is measured along three dimensions: product quality, development lead-time, and cost (Clark and Fujimoto 1991). Analogously, others have introduced the distinction between design efficacy and design efficiency.[65] Design efficacy is concerned with the output level of the design process, i.e., the product quality (or performance level). Since this dissertation's primary concern is the link between product architecture and cost, the focus of the discussion below will be on the question of design efficiency.[66] Although not entirely interchangeable, due to their close relationship cost and time will be jointly discussed.[67]

---

[65] Some authors point out the difference between design effectiveness and design efficiency. While the former measures what level of design quality was achieved, the latter measures the resource (and time) consumption required to arrive at a given level of design quality. Braha and Maimon, call this the difference between structural design complexity and functional design complexity (Braha and Maimon 1998).

[66] Obviously, one could ask the question of what kind of product architecture is capable of delivering the highest level of product quality. Although this is likely to affect indirectly the costs as well, it is a question fundamentally different from the one how product architecture differences affect costs.

[67] The exact nature of time-cost trade-offs is more complicated and covered by its own literature stream. For example, Roemer et al. develop a model to determine the effects of *crashing* activities versus *overlapping* activities – to trade-off lead-time reduction against a cost increase (Roemer 2000).

To address the fundamental question of how do architectural differences of a product affect the resource consumption during the design phase, some researchers have linked the task structure during the design process to the product architecture (e.g., Eppinger et al. 1994, Gulati and Eppinger 1996, von Hippel 1990). Assuming that the organizational structure mirrors (at least to some extent) the product architecture (Henderson and Clark 1990), the effort for design and development is determined by two parameters: the number and size of engineering design teams and the type and frequency of their interactions. The number of engineering teams and their size is then determined by the product's function-component allocation scheme. Communication in product development occurs in two dimensions: within teams and between teams. Research results indicate that a medium level of number and size of teams provide the opportunity for efficient design processes. This is because conducting design processes too much in a sequential manner (i.e., very small teams) might spare some iteration but will consume a long total time. On the other hand, conducting the design in one large, complex process, on the other hand, will cause numerous internal iterations.

Therefore, creating product architectures that balance the design complexity that incurs between the 'clusters' (read chunks, modules, components, etc.), i.e., the product as a whole (integration effort), on one hand, with the sum of the design complexity within the chunks on the other, appears to be a resource efficient approach. In fact, some methods developed to guide designers in decomposing product architectures pursue this goal (Michelena and Papalambros 1997). As for total development time, for a turbopump of a rocket engine, it has been shown that there is a number of blocks of the product architecture (modules, chunks, etc.) that translates into a certain number of teams, that minimizes the duration of the project development project (Ahmadi et al. 2001). Apparently, both costs and time functions exhibit a minimum if the product is decomposed into at certain number of subunits, and increases if less but larger subunits are chosen, and increases when more, but smaller subunits are selected (Figure 4-5).[68]

---

[68] In addition to the number and type of development process steps, the order of the processes and the emphasis placed on them can also impact time and resource consumption. Thomke and Fujimoto find that 'front-loading' the development effort, i.e., to identify and solve major problems early on, increases product development performance (Thomke and Fujimoto 2000).

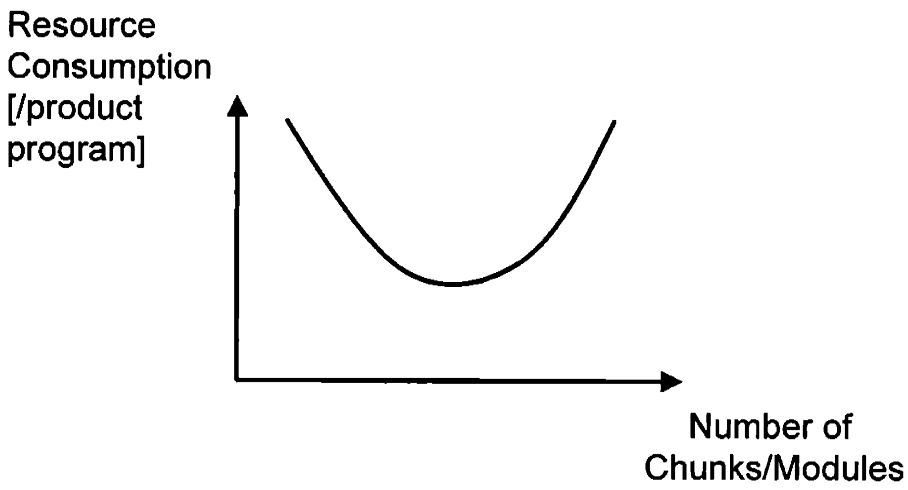


Figure 4-5: Resource consumption in product development

The precise value of *time* compared to *cost* can depend on a number of market parameters as well as the ratio between revenue and cost. For example, companies operating in fast market environments will especially value a product architecture's potential to reduce the time-to-market. Clustered architectures that allow conducting much of the design process at each cluster in parallel in order to arrive at the shortest possible total design time are of particular value for them. In a specific case about a Polaroid camera, for example, it has been found that the foregone sales in case of a longer development time far outweigh any achievable cost savings in manufacturing (Ulrich et al. 1993). In a case like this, to reduce development time is far more valuable than any cost savings in the production phase due to architectural changes.[69]

Also, strictly speaking, the design phase is only one component of time-to-market. If *market* is understood as sale (or begin of operation) of the first unit, then production preparations become part of the time-to-market, in particular tool design and production. Hu and Poli have compared assemblies made from stampings with injection molded parts regarding their effects on time-to-market (Hu and Poli 1997b). They find that parts consolidation can be disadvantageous

---

[69] On a related issue, Ittner and Larcker have tested the hypothesis that shorter development cycle time is associated with superior organizational, and thus, financial performance, and received mixed results (Ittner and Larcker 1997). In two industries (automobile and computers) they find stronger support for the hypothesis that certain product development practices (e.g., cross-functional teams, advanced design tools) improve performance.

with regards to time-to-market when the tool making for larger, more complex parts extends the total production time.

In addition to the particular product decomposition into 'chunks' (subsystems, modules, components, etc.), the characteristics of the interfaces between the 'chunks' are likely to affect design efficiency too. The weaker the interface connections are, i.e., the lower their intensity, the more the different design teams can be working independently on different subsets of the product. This may reduce the number of iterations between the teams, and thus increase overall design efficiency. In a case study of the development of an automotive climate control system, strong coupling has been identified as one reason for development cost increases (Terwiesch and Loch 1999). The weaker dependencies also may shorten the total development time (time-to-market) because it allows the design tasks to proceed in parallel. Similarly, analyzing the product development of integrated circuits (ICs), it has been found that higher levels of interface independence increase the design flexibility and reduce the risk of having to repeat experiments, i.e., prototypes (Thomke 1997). This independence of the design groups can also be affected by the interface nature. For example, a shift from mechanical connections (or interfaces), i.e., in Pimmler and Eppinger's framework *spacial* interfaces, to interfaces that are more electrical and electronic in nature, i.e., *power* and/or *signal* interfaces, will allow more freedom with regards to packaging components (Pimmler and Eppinger 1994).[70]

Finally, another dimension of how both characteristics of a product architecture, i.e., its function-component allocation and its interfaces, can affect development costs is a consequence of the nature of development work. Development costs are one-time costs in the product program's life, i.e., their contribution to the unit costs is highly sensitive to changes in the production volume. If only one product is ever produced, say a racing boat, this single unit has to bear all development costs. In contrast, for mass-produced products like vacuum cleaners, the

---

[70] On a related issue, Whitney develops an argument why mechanical design cannot be like VLSI design. Core of his argument is that basic physics often prevent mechanical designs from being separated into building blocks. For example, while VLSI process signals, mechanical designs often carry significant power. As a result, back load phenomena in mechanical designs are much harder to control and often virtually impossible to eliminate. "VLSI elements don't back load each other because they maintain a huge ratio of output impedance to input impedance, perhaps 6 or 7 orders of magnitude. If one tried to obtain such a ratio between say a turbine and a propeller, the turbine would be the size of a house and the propeller the size of a muffin fan. No one will build such a system. Instead, mechanical system designers must always match impedances and accept back-loading. This need to match is essentially a statement that the elements cannot be designed independently of each other." (Whitney 1996, p. 20).

design costs are shared by potentially millions of identical products. This issue is particularly important for the assessment of cost implications of architectural decisions, since some architectures allow better sharing of portions (chunks, modules, components) of the product with those of other products, and, therefore, allow the sharing of their development costs. The savings through the re-use of designs affects both development cost and time (Reinertsen 1997). A similar argument is made in the literature on multi-project management (Nobeoka and Cusumano 1993).

4.2.2.2 Production phase

In the case of many assembled and mass-produced products, the production costs are particularly important. For this reason, the effects that design decisions have on a product's 'producability' and, therefore, on its production costs, have always been of particular interest.[71] This is the reason why most design-for-X (DFX) techniques have been developed for sub-phases of the production phase.

With respect to the impact of product architecture on production costs, two sub-sets of production costs will be discussed here: (1) manufacturing and assembly, and (2) logistics. Each of the subsets is discussed in detail below.

4.2.2.2.1 Manufacturing and Assembly

Using the product architecture perspective, I begin with discussing the effects of size and number of components, i.e., the function-component allocation scheme, on manufacturing and assembly costs.[72] Based on historical experiences, cost reducing guidelines have been developed. The most widespread known representatives of this DFX family, design for manufacturing (DFM), and design for assembly (DFA), try to bring the focus on design features

---

[71] More than a decade ago, Nevins and Whitney have already pushed for more simultaneous consideration of product design and process design (Nevins and Whitney 1989). Their example of an automotive radiator demonstrates clearly the effect that a well thought-out product architecture can have on a production facility and, thus, its production costs (ibid., p.54). More recently, it has been argued to deploy similar techniques across firm boundaries. Because many of today's complex products are made of many components and subsystems and since many of these components and subsystems are no longer produced in-house, but produced by external suppliers, it has been suggested to expand simultaneous engineering practices to include purchasing departments (Bradley et al. 1997).

[72] All arguments are directed to unit costs.

that consume avoidable resources during manufacturing and assembly, respectively.[73] Design for manufacturing aims for simplifications of the manufacturing processes, which results in reduction of process variability and ultimately in higher yields. In contrast, design for assembly generally emphasizes part count reduction, the use of only one assembly direction and the preference of symmetrical parts (Boothroyd et al. 1994).[74] Evidence exists that supports both claims individually. In case of automobile rear lamp production, for instance, it has been found that complex products requiring complex manufacturing processes result in higher costs compared to simpler parts with simpler processes (Banker et al. 1990).[75] On the other hand, comparing the costs of electromechanical assemblies, it has been found that the assembly cost savings through part count reductions can be significant (Boer and Logendran 1999).[76] Part count reduction is generally seen as a cost reduction tool (Schonberger 1986, Galsworth 1994). These findings result in cost curves that increase in opposing directions and the minimum of the sum of the two depends on the specific shape of these two curves (Figure 4-6).

---

[73] For decades, most engineering handbooks have been giving general guidelines to engineers on how they affect costs by their choices of typical basic parameter, for instance tolerance or surface roughness (e.g., Michaels and Woods 1989).

[74] Boothroyd et al. have collected data from various case and industry studies and offer detailed cost functions for a large array of both manufacturing and assembly processes.

[75] Banker et al. identified five complexity factors: (1) number of moving parts of the mold (tool), (2) multicolor molding, (3) number of functions (of the product), (4) length of component (product), and (5) number of rejects. The most significant cost driver they found was number of moving parts in the injection-molding tool.

[76] In their study of a variety of potential cost drivers' impact on cost and time, Boer and Logendran find that number of parts in a product and number of assembly processes are the most important cost drivers. For this very reason, design strategies have been developed to support product redesign for part count reduction (Lefever and Wood 1996).

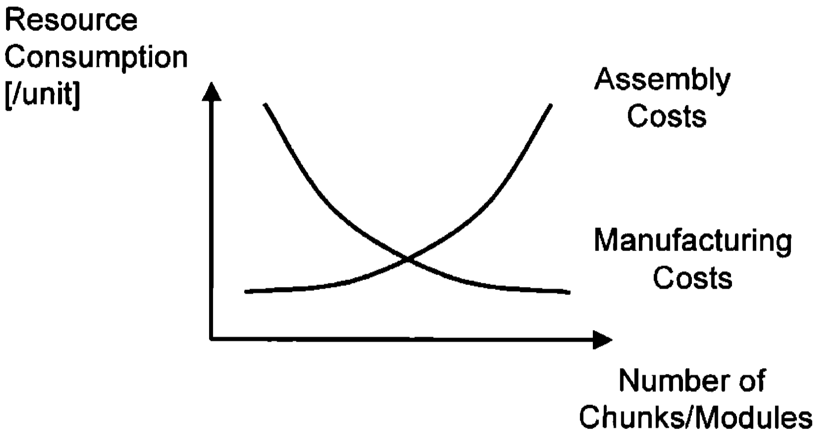


Figure 4-6: Production Costs

The argument for products requiring simpler manufacturing processes rests essentially on the concept that these processes perform more reliably than their more complex counterparts. Assuming that simpler products require simpler manufacturing processes, this means the product feature *complexity* affects the efficiency of the process (not only the efficacy of the product) which in turn directly affects the costs via process yields. In other words, a design that allows processes (a) to be robust, and (b) to balance variations among processes instead of accumulating them (and, therefore, contribute to an overall yield increase) is more likely to consume fewer resources.[77] With respect to product architecture, this observation caps the size of modules or chunks below a complexity level that becomes difficult to control. However, the relationship between prevention and appraisal costs is not necessarily monotonous or linear. For instance, for many processes the cost to produce 'good' units raises sharply when 0% defects are approached (Fargher and Morse 1998).[78]

The argument for products requiring fewer parts (and, likely, less manufacturing processes and assembly steps) to achieve lower costs is immediately obvious, as long as the process reduction is not paid for with lower yields or higher overheads. A shift from one manufacturing

---

[77] Particularly choosing certain joint designs can introduce adjustment opportunities that facilitate assembly, which in turn increase yield, which in turn result in lower unit costs (see Hu 1997 for an example how dimensional variations can accumulate in automotive body assembly and how design choices can mitigate this problem). Also, in addition to lowering direct unit cost, higher yields also tend to lower indirect cost, as rework inspection etc.

[78] Ideally, these costs should be linked to external failure costs (warranty costs) for a complete picture.

process to another to reduce part count can have a dramatic impact on assembly time and cost. For example, the instrument panel for the cockpit of the commercial aircraft Boeing 767-4ER used to be manufactured from 296 sheet metal parts and assembled with 600 rivets. A move to precision casting has reduced the part count to 11 and the assembly time from previously 180 hours to 20 hours (Vollrath 2001). In sum, the product's function-component allocation, i.e., its number and size of components, affects both manufacturing and assembly costs, typically in opposing directions.

From a unit cost perspective there is one other effect of product architecture on production costs: this is the use of common components *across* products. If the fixed cost portion of manufacturing and assembly can be distributed across more units, the unit production costs decrease. However, the magnitude of these savings also needs to be compared with the potential cost penalties for over-designing a sub-unit or module. For example, products whose costs are dominated by variable materials costs, such as automotive wire harnesses, may not gain through the use of commonality (Thonemann and Brandeau 2000).[79]

But not only the architecture with respect to number and complexity of parts impacts costs, but also the characteristics of the interfaces between them. Interfaces preferred from the low cost production perspective are such that they minimize complexity and uncertainty within the production process. This means, the better the process is known and the more likely it can be performed successfully, and the lower the total number of different processes in the production system is, the lower the expected production costs. The nature and intensity of the interfaces can also be relevant to the production. For example, electronic interfaces that consist only of a plug and a socket may be easier to assemble error-free than a complex mechanical rod connection. In addition, nature and intensity may affect the testability of pre-assembled units and modules, and in turn their costs.

4.2.2.2.2 Logistics

Continuing globalization with distributed production facilities lets logistics costs also come into play, especially for larger products. For the purpose of this work I define logistics costs as encompassing cost for storage, transportation and work-in-process (WIP) including inventory. While storage and transportation costs depend on number, weight, volume, and packaging

---

[79] The effect of common parts usage on inventory through risk pooling is discussed in the next section.

requirements of each subsystem that is directly affected, the size of WIP depends on the value of the product (or its subsystems), the speed of the supply chain, and whether parts can be pooled.

Storage and transportation need to be considered both inside and outside the plant, between suppliers and plant, and between plant and customers. For an automotive component plant, the logistics cost for internal transport and handling have been found to amount on average to 10% of the unit cost on the plant level (Datar et al. 1991). Logistics outside the plant obviously depends on mode of transportation, distance traveled and frequency of delivery. Product architectural features are most likely to affect these costs to the extent to which they determine packing space and protection requirements.

Architectural differences can also impact costs for inventory or work-in-process. The more an architecture allows late customization or postponement strategies, the more it can contribute to savings in storage and work-in-process costs.[80] A reduction of holding costs, (storage and WIP) caused by architectural differences is essentially based on the idea of risk pooling. Parts commonality has been identified as a way to reduce the safety stock level for a given service level (Collier 1982). Others have shown, however, that while the stock for a common part can be lower compared to the unique parts it replaces, the safety stock of the remaining unique components increases if a certain service level is to be maintained (Baker et al. 1986). Gerchak et al. confirm these findings for an arbitrary number of products and joint distribution as long as the costs for the product-specific components (that are replaced by a common one) are equal (Gerchak et al. 1988). For the two-product case, Eynan and Rosenblatt derive cost ratios that bound the advantage of the use of common components (Eynan and Rosenblatt 1996). In sum, the use of common parts can reduce inventory, but it needs to be investigated with (a) the specific demand pattern, and (b) the comparison with other cost (penalties) in mind.

Finally, the architectures effect on *time* can have additional impacts on costs via the detour of increasing demand volatility. Because demand volatility increases upstream ('bullwhip-effect'), architectures can reduce this effect if they allow for short lead times. Long lead times,

---

[80] From the automotive supplier industry in North America empirical evidence exists that processes with long lead times (e.g., stamping, heat treating, forging) are prone to exhibit larger inventories (Lieberman et al. 1999). No significant correlation was found between inventory levels (raw materials, work-in-process, and finished goods) and number of components the characteristic product consisted of.

together with high levels of demand uncertainty, can amplify the bullwhip-effect and create significant additional costs in the supply chain (Levy 1994).

4.2.2.2.3 Total supply chain

A complete assessment of the impact of architectural characteristics on production costs should incorporate manufacturing, assembly, and logistics costs. It is important to understand the order of magnitude of these different cost blocks. For example, Ernst and Kamrad, who investigate modularization and postponement as inbound and outbound flexibility, construct a model to evaluate different supply chain structures. Although implicitly, they find that cost saving effects on production and packaging processes is greater than the savings through inventory reduction (Ernst and Kamrad 2000). Overall, it seems that architectures that balance manufacturing process complexity with assembly process and logistics complexity are likely to gain total cost advantages.[81]

Finally, for the production phase there is also another criterion besides cost to consider: time. What is a time-to-market question in the design phase becomes a time-to-customer question in the production phase. The fraction of the time aspect of production (time-to-customer) that relates to the product architecture depends on the critical element on the time path. If it is assembly, a hierarchically structured architecture allows a higher fraction of the work to be conducted in parallel than more fragmented or integral architectures. If certain manufacturing processes (e.g., painting) are the critical work activities with respect to time, the architecture that is less dependent on the logic of these processes has an advantage to proceed faster through production. Finally, if the time critical element is rather the supply, then the use of common parts may not only reduce inventory (i.e., cost) but also lower the risks of lacking the right part at the right place at the right time (risk pooling).

---

[81] There are also possible indirect cost effects caused by change in the product architecture. For example, a change in the product architecture is likely to affect several players in the value chain, and may shift the power in supplier-buyer relationships (van Hoek and Weken 1998). These secondary cost effects are beyond the scope of this thesis.

4.2.2.3 Use and operation phase

In general, four types of costs occur during product use. These are the costs for operation, for preventive maintenance, for corrective maintenance, and all external costs incurred by the operation of the product.

4.2.2.3.1 Operation

Most products require some input to operate them. The costs for these inputs can be utility costs for energy, water, or pressurized air, or costs incurred by the product's characteristic, for instance labor requirements for a machine operation or taxes determined by the size of the engine of a car.[82] All these input costs are labeled here as operation costs. It is very difficult to make a general statement about the relationship between product architecture characteristics and operation costs, but some issues can be pointed out. For example, if a product's architecture affects the level of training for personnel, either directly or indirectly via other products in the product family, then the architecture affects the operating costs.[83] Or, if a product is frequently used in multiple modes, e.g., a tool machine, architectural features that allow to reconfigure the product quickly, i.e., function-component alignment and interface reversibility, improve the productivity of the product, or reduce its operating cost as measure by unit produced.

4.2.2.3.2 Maintenance

Most products that are designed for longer use (or multiple uses) need some maintenance from time to time. If the type of maintenance is intended to maintain the product's operability above a predetermined level, it is called preventive maintenance. Consumables like lubricants or oil, battery replacements, or the labor and equipment required to add or replace these materials fall in this category. In case the operability of the product drops below the predetermined level, corrective maintenance is required. This is induced by a total or partial failure of the product, which in turn can have various causes. They range from poor product quality (e.g., product does not perform as intended) to product misuse (e.g., using a product beyond its recommended age) to external events (e.g., a thunderstorm). Both types of maintenance may also incur costs by having the product unavailable for the duration of the maintenance work (e.g., machine

---

[82] In some countries (for example Germany) annual car taxes are calculated based on the displacement volume of the internal combustion engines.

downtime). This is particularly important for large-scale systems, i.e., capital-intensive equipment.

The topic maintenance, both preventive and corrective, has also been discussed under the headline of warranty cost. Warranty costs occur when a consumer claims the non-satisfactory performance of a product. Several theories describe the origin of warranty policies, emphasizing different economic effects and incentives. Blischke presents three theories to explain warranty as a tool to deal with information asymmetries. The exploitative theory according to which manufacturers would sell their customers low quality products under conditions that he determines; the signal theory, that explains warranty as a signal of product reliability by a high quality producer; and the investment theory, in which a warranty is understood as both an insurance policy and a repair contract. (Blischke and Murthy 1994, p.25). Whatever the precise theoretical underpinnings of today's warranty policies might be, they are themselves contractual agreements between the manufacturer of a good and the buyer of that good. In that sense they deal with the question of who will pay for some of the costs that occur during operation. From a neutral product life cycle perspective, there are two major questions to be concerned with. First, what is the likelihood that maintenance costs will occur during the product's use phase and, second, what will be the anticipated cost level for this maintenance? In general, both of these elements are unpredictable for each single produced unit. However, a broad stream of research literature has developed models and statistical analysis techniques to determine these costs under various conditions and warranty policies.[84] Using probabilistic models for product reliability and expected repair cost and time, it is possible to model the trade-off between up-front investments, i.e., for higher quality products, and the risk of higher operating cost. These costs considerations can be conducted from various perspectives. First, manufacturers are likely to be interested in profit maximizing. They tend to trade off higher product costs now against higher servicing costs later.[85] Second, buyers' utility may have also additional interest other than the pure cost

---

[83] Aircraft producers, for example, are trying to install similar, if not identical, cockpits into machines of different sizes, to reduce the need to retrain the crews.

[84] See Blischke and Murthy 1994 for a comprehensive discussion on warranty cost modeling approaches and techniques. Thomas discusses some decision problems in warranty planning for manufacturers (Thomas 1999).

[85] This thought concentrates on a pure cost argument, and excludes a company's considerations that are of rather legal or long-term strategic nature. In cases where legal consequences of warranty either potentially ruin the firm or threaten the management to be sent to jail, the cost trade-off decision might be extended to encompass

trade-off. Depending on its level of risk aversion, a high level of product reliability, i.e., low risk may represent a high value.[86] Third, the public policy perspective introduces additional goals. In most Western societies it aims to establish fair market conditions. Given that certain information asymmetries exist, especially for complex durable consumer products, the policy maker may require warranty policies that balance purchasing with selling power for the transactions. Furthermore, public policy attempts to maintain public health and safety standards. This also can have a strong effect on required warranty terms.

How do different product architectures affect the maintenance costs during the product's use phase? Again, it is almost impossible to make a general statement about the relation between product architecture and product reliability. Nevertheless, some major issues deserve consideration. Grouping parts with similar expected lifetimes together (function-component allocation scheme) is likely to reduce the repair and replacement costs. In addition, a product architecture that allows easy and fast access (interface reversibility) for either form of maintenance may cause lower costs for the maintenance.[87] Also, in case that a product has several identical parts (function component allocation scheme), less of these parts need to be stocked in inventory (compared to unique parts) for providing the same level of availability (Perera et al. 1999). Like the risk pooling *across* products, this should translate in lower spare part costs as part of the maintenance costs.

4.2.2.3.3 External cost

Finally, the operation of any product may also cause so-called external costs. These can be, for instance, damages to public health or the environment through emissions. A link between

---

additional parameters. The legal consequences of product liability, i.e., negligence, strict liability, or misrepresentation are especially harsh when issues like public health and safety are involved (see Akula 2000). Also, strategic consideration regarding firm reputation may lead to decisions other than the one with the lowest cost determined by the trade-off between production and service cost. For example, Ford's decision in August 2000 to actually close three assembly plants for two weeks to free up more tires for its replacement action after several deadly accidents caused by original tires displays considerations of both legal and strategic nature (Bradsher 2000b).

[86] Also, users might differ in their usage pattern of a particular product, i.e., intensity of use may vary. For a number of products, firms have introduced multi-attribute warranty plans. For instance, a car's warranty usually is limited by both vehicle age and mileage run. The real costs of a specific warranty plan then depend on the particular use patterns of the customers, unless the warranty plan uses an iso-cost warranty policy (see Chun and Tang 1999).

[87] To facilitate repair, Murthy and Blischke suggest, among other strategies, to us modular designs to reduce warranty costs (see Murthy and Blischke 2000, p. 48). They do not provide, however, a detailed specification of what they mean by 'modular design.'

product architecture features and external costs is very difficult – if at all - to establish, and goes beyond the scope of this work.

4.2.2.4 Retirement phase

In the last phase of a product's life cycle, costs are created by activities like disassembly or disposal. In addition to these direct costs, external costs like degradation of the environment or air quality can occur.

4.2.2.4.1 Disassembly and disposal

To estimate disassembly costs as a function of the product design or the product architecture is very difficult, particularly since it is often unclear which of several possible disassembly sequences is the most economical. The reverse of the assembly process may or may not be the most cost effective way to disassemble the product. Researchers have suggested a number of scoring processes to compare disassembly efforts for different designs. Some suggest comparing disassembly costs for different designs on a relatively high level of aggregation. Emblemsvag and Bras, for instance, propose to list all activities the disassembly of various products would require, compute the costs for each activity per time unit, determine the time each design requires each activity, and compare the results (Emblemsvag and Bras 1994). This type of analysis, however, does not tell us specifically what architectural features make one design more costly to disassemble than another. To answer this type of question, more detailed analyses are required. Das et al., for example, propose to compute a disassembly effort index based on seven factors, like time, tools, fixtures, access, instruct, hazard, and force requirements (Das et al. 2000). The fact that both the score for each of these factors as well as the weighting among them is based on qualitative assessments, demonstrates the problematic nature of the task to determine disassembly costs unambiguously. Others have extended this work to include bulk recycling in addition to disassembly activities (Sodhi and Knight 1998). However, while the product architecture affects disassembly (function-component allocation scheme and interface reversibility), its impact on bulk recycling is only relevant together with the specific values of the materials involved.

To determine the costs to landfill a product or parts of it are relatively straightforward. The results, however, are unlikely to depend on architectural features of the product (leaving material consideration aside).

4.2.2.4.2 External costs

Finally, like in all other product life phases, the recycling and disposal phase is likely to create external costs. These can be air and water emissions or degradation in landscape 'value' by contributing to landfills. Not only are these costs very difficult to assess (see the entire LCA literature), but also it is almost impossible to establish a cause and effect relationship regarding product architecture and these costs (again neglecting materials).

4.2.3 Role of time in cost calculations

The analysis of various activities along a product life cycle as potential cost sources and their link to the product architectures brings attention to an issue that is particularly important for cost assessments over long periods of time: the time value of money.

Every financial calculation that encompasses time periods longer than a year, uses interests rates to discount costs and revenues occurring in various time spots. Most often the values are discounted to arrive at comparable present values (de Neufville 1990).

Often the results of these calculations are very sensitive to the choice of the discount or interest rate. A high interest rate makes costs that occur many years after the product life cycle has begun may to appear as irrelevant and costs in the near future as very important. Vice versa, the choice of low interest rates increases the relative weight of future costs for the discounted value. Therefore, particularly cost analyses that are concerned with long product life cycles should conduct sensitivity analyses to ensure an understanding of the impact of the chosen discount rates.

4.3 Setting the Analysis Boundaries with respect to Cost Types

While the previous section was concerned with the analysis boundary settings along a product's life cycle, this section focuses on the analysis boundary settings regarding cost types considered for a particular cost assessment.

Two major issues fall into this category of potentially troublesome boundary definitions. They are closely related with each other. The first deals with the problem of the unit of analysis, the second is concerned with the question of direct versus indirect cost.

4.3.1 Unit of analysis

Most often, the unit costs are chosen for cost comparisons of assembled products. There are, however, other units of analysis that could be selected alternatively. For example, these could be product families, programs, departments, factories, companies or entire economies. The order of this list of potential levels of the unit of analysis indicates an increasing distance from the physical object itself. While a cost analysis focusing on a product makes it easy to assess costs that are directly related to the product (say material consumption), it makes the allocation or more 'distant' cost very difficult (say factory guards). On the other hand, for cost analyses on a company level, almost all cost are somewhat 'direct' (Figure 4-7).

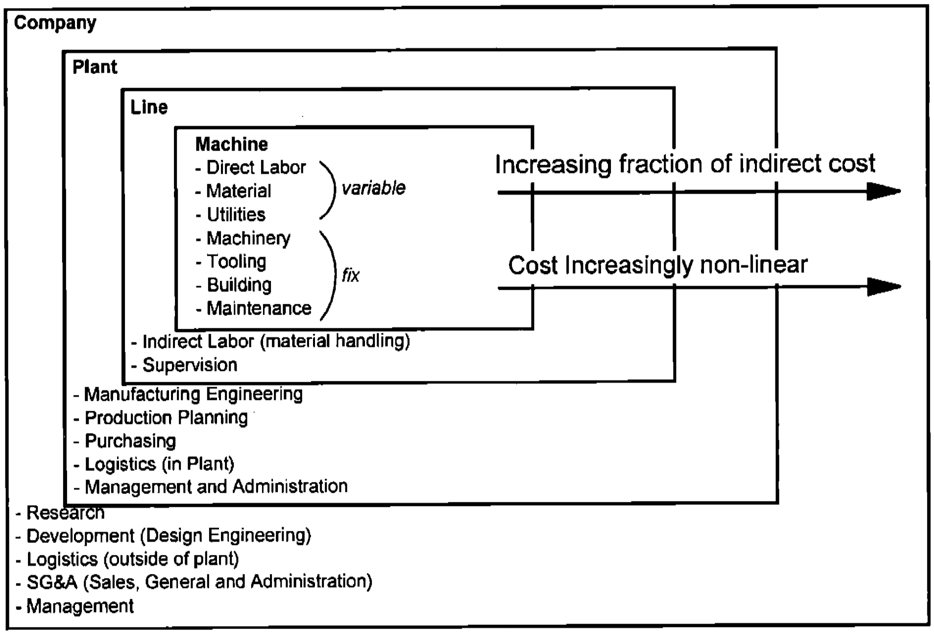


Figure 4-7: Different levels for cost analyses

Since this study is concerned with the cost effects certain product architecture choices trigger, it is logical to focus the cost analysis on a level where product architectures can be distinguished, i.e., on the product or a product family level. This in turn creates the above mentioned allocation problem large portions of the costs. In the past, the majority of the costs of a particular product were closely related to the design, production, and use of this specific product. This was mainly caused by two factors: (1) relatively low product variety and (2) high fractions of total costs were in direct labor for manufacturing the products (e.g., Cokins 2000).

In contrast, today's design, production, and often use processes, are at least as complex as the products they are intended to produce and use. This has resulted in increasing fractions of costs that are not closely related to a specific product. In everyday language the term 'overhead' is used for this cost category. 'Overhead,' usually encompasses costs with various levels of 'indirectness.'

4.3.2 Direct vs. indirect costs

The direct-indirect classification depends on the choice of the cost object. Horngren and Foster point out the following relationship:

> "A useful rule of thumb is that the broader the definition of the cost object, the higher the proportion of its total costs are its direct costs – and the more confidence management has in the accuracy of the resulting cost amounts. The narrower the definition of the cost object, the lower the proportion of its total costs are its direct costs – and the less confidence management has in the accuracy of resulting cost amounts." (Horngren and Foster 1991, p. 28)

Further, the accounting literature distinguishes between direct versus indirect costs and fix versus variable costs. While the first uses the criterion of traceability to separate direct from indirect costs, the second uses the dependency with regards to changes of the production volume as a measure to classify fixed from variable costs (Figure 4-8).

In the production arena, costs that are typically considered as variable are costs for direct labor, materials or utilities. In contrast, machinery, tooling, and building costs are usually considered fixed costs. These distinctions, however, are not clear-cut, but depend on the chosen time horizon, the chosen manufacturing technology, and the chosen accounting principles.

		Traceability of Cost to Cost Object	
		Direct	**Indirect**
Cost Behavior Patterns	**Variable**	Tires used in assembly of an automobile where the cost object is each individual automobile assembled	Power costs where power is metered only to the assembly department and the cost object is each individual automobile assembled
	Fixed	Marketing department's supervisor's salary where the cost object is the marketing department	Board of directors' fees where the cost object is the marketing department

Source: Horngren and Foster 1991, p.31

Figure 4-8: Fixed vs. variable and direct vs. indirect costs

A change in the chosen time frame can turn the same costs from fixed into variable costs. Labor costs, for example, viewed in short time frames become fixed costs, whereas in the long run they are typically variable in nature. The choice of a manufacturing technology may determine whether a specific or a generic tool is deployed. For instance, a shear as a cutting tool that can be used to produce other products as well exhibits variable cost behavior, whereas a specific cutting die doing the same job, but that can only be used for this specific product, becomes fixed costs. Finally, certain accounting principles can shift costs from the fixed cost column into the variable cost category and vice versa. The assumption, for example, that free machinery capacity can be employed for other jobs turns the allocated machine cost effectively into variable cost, whereas the assumption that the machinery is dedicated to a specific product results in fixed cost behavior.

In sum, what is typically called *overhead* is a very broad category. It is, however, a category that becomes increasingly important with the increasing product and process complexity of today's products. This has several causes. First, as mentioned earlier, the shrinking direct labor content contributes significantly for this development (e.g., Miller and Vollmann 1985, Galsworth 1994, p.63, Doran and Dowd 1999). Second, shorter time frames for product life cycles, and, third, increasingly heterogeneous markets, both demanding higher levels of customization, result in smaller production volumes per unique product.[88] Although difficult to specify, it is unlikely that the indirect cost shrink as fast as the direct cost. Table 4-1 gives an overview of the magnitude of plant-level overhead costs found in recent studies.

Author(s)	**Total Costs (=100%)**			**Activities considered**	**Industry**
	Direct Materials	Direct Labor	Manufacturing Overhead (MO)		
Banker, Potter and Schroeder (1995)	65.4%	8.9%	25.7%	Plant level study	Electronics, Machinery, Automobile components (mean values of 32 facilities)
Foster and Gupta (1990)	54.3%	6.6%	39.1%	Procurement, Production, Support	Electronics (mean values of 37 facilities)
Galsworth (1994) [p.85]	40%-65%		35%-60%	Total Costs: Function cost: 40% Variety cost:25% Control cost: 35%	Manufacturing
Hundal (1997)	45%-65%	8%-20%	22%-40%	Not specified	Aerospace, Computers, Electronics, General Equipm., Automobiles
Miller and Vollmann (1985)	20%-40%		60%-80%	Overhead Costs: G & A 20% Indirect Labor 12% Engineering 15% Equipment 20% Materials OH 33%	Electronics

Table 4-1: Estimates of 'overhead' costs

---

[88] The recent phenomenon of shrinking production volumes per model is quite significant in the automotive industry. Two examples: In the 1960s, GM's top selling models sold up to 1 million units per year, while today annual production volume of 300,000 units are considered 'upper range' (Anonymous 2001a). And in early 2001 Opel brand management executive director Alain Uyttenhofen stated: "For Opel to remain a mass manufacturer with annual sales of more than 2 million vehicles, it needs 20 models to compete, rather than 10 or 12 models years ago" (Ellis 2001).

Although *overhead* can have numerous causes, I will categorize its major forms for the purpose of analyzing product architectures' effects on costs along three lines of thought. First, there is what I call the general allocation problem, i.e., how can indirect costs be allocated to certain cost drivers. This includes a discussion of the choice of appropriate cost drivers. Second, whatever the selected cost drivers, there is an additional question regarding the shape of the cost curves. Various factors can introduce complexity in products or processes and result in non-linear cost curves. Thirdly, in complex production processes it is likely that the cost driver themselves are not independent from each other.

4.3.2.1 General overhead costs allocation problem

One characteristic feature of 'overhead costs' is their lack of direct volume dependency. In other words, activities that support in various ways the direct production processes do not necessarily vary in direct proportionality with the production volume. It has been argued that the costs for these activities vary with the intensity of frequency of these activities. For example, the time and manpower to write a purchasing order does not vary with the number of equal parts ordered, but each order incurs an average cost for the transaction 'write purchasing order.' Based on this idea, activity based costing has been developed (e.g., Kaplan 1991, Kaplan and Cooper 1998). It promotes a cost allocation process in proportions to the activities consumed by the products produced. Since activities are performed on various levels in the enterprise, the activity/product relationship can vary, and thus the product/cost relationship. Based on this observation, four levels of activity have been suggested to allocate costs: unit-level, batch-level, plant-level, and firm-level.

Empirical findings vary in their strength with which they support the argument that transactions are better proxies than production volume to allocate overhead costs. Foster and Gupta, for example, find strongest support for overhead driven by volume-based variables, and only limited support for complexity- and efficiency-based variables as overhead cost drivers (Foster and Gupta 1990). They caution their findings, however, by pointing out the difficulties of operationalizing complexity and efficiency notions.

In another empirical analysis of manufacturing overhead cost drivers, Banker et al. find support for the activity-based costing argument that overhead costs are driven also by number of transactions, in addition to volume (Banker et al. 1995). Using Miller and Vollmann's framework of four types of transactions: logistical, balancing, quality, and change (Miller and Vollmann 1985), Banker et al. approximate these transactions with area of floor space per part, purchasing and production planning personnel, quality personnel, and engineering change orders, respectively. In their cross-sectional analysis of 32 plants from the electronics, machinery, and automobile components industry, they find statistical significance of all of these measures with respect to manufacturing overhead costs.[89]

On each of the four suggested levels, the relationship between transaction and costs may itself depend on other variables. For example, the sensitivity of product costs to batch-size is strongly dependent on the used process technology. In three different case studies, Blocher and Berry find that design quality can change this relationship. They also find that the effect of product variety on product cost varies with the process technology and workers' education level. (Blocher and Berry 1998).

This discussion indicates that there are overhead costs related to activity frequency rather than production volume. The strength and particular shape of this relationship, however, can vary between industries and employed production technologies.

With respect to the question of how characteristics of the product architecture affect the cost allocation process, some observations can be made. An architecture that allows operations conducted closer on a per-unit basis allows more precise cost allocation. For example, a process that produces only one part at a time allows easy allocation of all non-direct costs (setup, purchasing, etc.). In contrast, architectures that cause complex logistical, balancing, or quality processes may make the cost allocation more difficult. To some extent, these are arguments for products with architectures consisting of fewer components (dimension function-component allocation) and with high levels of interface standardization.

4.3.2.2 Specific overhead costs allocation problems due to non-linearity

The fundamental idea of activity-based costing is to relate overhead costs to activity frequency. Typically, this is achieved by calculating the costs for an activity and dividing it by

---

[89] Banker et al.'s study demonstrates the relation between the cost drivers and overhead cost. They do not, however, explain whether the variations among the cost drivers are caused by differences in complexity issues of product structures, production efficiencies, or other reasons.

the number of products that collectively consume the activity. Implicit in this approach is a fundamental simplification: that the cost curve is linear. In reality, however, the relation between cost driver and cost is likely to be non-linear. Three reasons can create this non-linearity: (1) overhead cost activities are themselves volume dependent, (2) variety related overhead costs create non-linear costs on higher levels, and (3) the overhead cost drivers themselves are interdependent.

4.3.2.2.1 Returns-to-scale effects: if activities are volume dependent

Overhead costs are typically calculated as average costs. The sum of expenditures for all transactions of one kind over a certain time period, for example, a year, is divided by the number of transactions. This pre-condition of strict proportionality between overhead costs and activity implicitly assumes constant returns to scale. In the real world, however, many activities exhibit increasing returns to scale. In other words, the cost for an activity (or fraction thereof) depends itself on the volume level on which the activity is performed. These returns-to-scale effects can be quite significant, i.e., can change the assigned cost by up to a factor of two (e.g., Noreen and Soderstrom 1994).[90] Note that this volume dependency may or may not coincide with the production volume itself. Although it can be assumed that their curves both increase or decrease in the same direction, their particular shape may be very different.

4.3.2.2.2 Product variety effects: if costs occur that do not 'belong' to any product

Another factor that influences the position and shape of the cost curves is product variety. Product variety is a tricky problem from the accounting perspective since it creates costs, many of them indirect, at different locations throughout the firm. As Lingnau notes:

> "It can be concluded from our [..] observations that the normal situation, where a large number of variants are produced, leads to cost increases in practically all areas of a company. There are cost effects notably in connection with administration, preparation and development. Both planning and implementation are affected in the functional areas. As a result, shifts can be observed in the cost structure, leading to an increasingly large proportion of indirect costs on account of the additional work which is necessary for planning, control, monitoring and coordination as opposed to the actual production process (Hoitsch 1997, p. 55). This finding underlines the great

---

[90] In their cross-sectional study of hospitals, Noreen and Soderstrom test whether overhead costs are proportional to overhead activities. In their data they find that the average cost per unit of activity overstates marginal costs by about 40%, in some instances up to 100%.

> importance that needs to be attached to cost effects when decisions are taken regarding the production programme, even if the changes in question are (apparently) only minor." (Lingnau 1999, p. 153)

In addition to the propagation effect that variety causes, it also creates an additional allocation problem, because product variety can 'create' costs above the sum of the single production costs. Viewed from this perspective, product variety is not really a product attribute, but rather a production program and facilities attribute. Two aspects deserve particular consideration, when trying to assign variety costs to products. The first aspect is the level of variation (how 'much' does one variant differ from another) and the second is the effect of variation over time (what is the chance that the next product is a different variant).

To distinguish different levels on which variety can be observed, Ittner and MacDuffie define three levels of product variety in their study of overhead costs in automotive assembly plants: core or fundamental (model mix complexity), intermediate (parts complexity), and peripheral (option complexity). They find empirical support only for the latter two affecting overhead costs, ".. reflecting the considerable logistical, coordination, and supervisory challenges that accompany an increase number of parts and more complex manufacturing tasks." (Ittner and MacDuffie 1994, p. 29) As an explanation for not finding evidence for their hypothesis that variations in the core designs produced in a plant are the most costly form of product variety, they conclude, that it may be ".. due to the fact that model mix complexity primarily affects the body shop, a capital intensive operation. As a result, switching models among the variants that the body shop equipment can handle may have only a minor impact on direct or overhead labor requirements."[91] (ibid.)

Another approach to specify product variety has been followed by Anderson. She measures the impact of product mix heterogeneity on manufacturing overhead costs by identifying seven independent product attributes from engineering specifications. By measuring on the attribute-level, Anderson finds that increased overhead cost "is associated with increases in the number

---

[91] Ittner and MacDuffie use number of indirect and salaried labor as a proxy to measure overhead costs. To normalize these numbers they define a standardized set of activities: plant management and administration, direct supervision, manufacturing and facilities engineers, product repair and inspection, production control, material handling, and maintenance. They exclude people from fire brigades and security services as well as from corporate functions like product engineering. Finally, they adjust for vehicle size and workers' absenteeism (Ittner and MacDuffie 1994, p. 12).

and severity of setups and increased heterogeneity in process specifications (expected downtime) and quality standards (defect tolerance heterogeneity) of a plant's product mix." (Anderson 1995).[92]

How product variety is distributed over time affects the effort to balance and sequence production line. In their study of product variety, Fisher and Ittner find that "[o]ption variability has significantly greater negative impact on productivity than option content in automobile assembly" (Fisher and Ittner 1999, p. 785). They also find that variety's impact on indirect and overhead labor is much greater than it is on direct labor. They explain this with the built-in slack in automotive assembly lines that allows handling option variation in the first place. They point out, that because these costs are born through the variability complexity, it is difficult to allocate these excess costs to any specific product.[93]

While the idea to capture potential savings through the use of common parts could be depicted by moving the unit-of-analysis up to the product-family level, this simultaneously introduces also the problem that it is not necessarily clear whether the use of average costs is justified. For example, the decision to use an identical part in two different variants may results in scale economies for this part's production. But to what extent 'contribute' the two variants to the total savings? This question refers both to their share in the production volume and to their individual distance between the costs of a common part and those of an individually optimized part.

Despite the problem of precisely allocating the costs induced by product variety, it is common understanding that variety is more often a negative feature (from the cost perspective) than a positive one (e.g., Galsworth 1994, Suzue and Kohdate 1990). Viewed from the other side, simulations have demonstrated that parts commonality and process sequence flexibility

---

[92] Subject of Anderson's study are three weaving plants of a leading U.S. textile manufacturer. Through interviews with engineers and line workers 22 product specifications were identified. Factor analysis reduced them to 7 independent product attributes: raw material content, fabric weight, expected machine downtime, warp beam construction, fill thread, defect tolerance, and warp thread.

[93] In another study, investigating the effect of product variety on labor hours (i.e., *direct* and *overhead*) for all segments within automotive assembly plants, MacDuffie et al. find only limited support for their hypothesis that high product variety results in lower labor productivity (MacDuffie et al. 1996). In fact, they find that higher option variability (one of their measures for variety) is associated with fewer hours per car. As an explanation, they suggest "that plants with very high option variability are on a different, more flexible production frontier with respect to all kinds of product variety, and hence are less affected by this variability than more inflexible plants." (p.367) In contrast, they find option content as being statistically significant for predicting productivity.

reduce the variation in a production system and therefore improve performance (Nagarur and Azeem 1999).

Others have developed a methodology to assess designs, i.e., product architectures, with regards to the likelihood that the product experiences the future need for variety, for instance, through market forces. Ishii et al. developed a qualitative method to compare the need for product variety to the costs to produce this variety (Ishii et al. 1995). Their method rates the cost of manufacturing a particular variation on a scale from 0 to 1, where 0 means low cost and 1 represents very high cost. The measure is a compound of a qualitative variation assessment in three dimensions: number of variations, stage in production where variation occurs, and how 'painful' is it to make the change.

Martin and Ishii extend this idea and develop a number of indices with which they qualitatively determine the economic effects of variety (Martin and Ishii 1996). They further develop their method and propose two indices to measure product family variety (Martin 1999a, Martin and Ishii 2000). The first, the generational variety index, measures the estimated need for a subsystem of the product to be changed for the next generation. The second measure, the coupling index, identifies the extent to which subsystems interact with each other along various dimensions (the authors use Pimmler and Eppinger's framework), both as receiver and sender. For both indices, the authors propose a qualitative assessment using a (0,1,3,9) rating system. These compounded indices compare product family vs. product family, but they do not disclose the underlying relationship between cost and design decisions directly, let alone between costs and product architecture features.

Generally, the same conclusions can be drawn with respect to how architectural features affect costs as in the previous section that discussed the general allocation problem. However, the phenomenon ('variety related costs') discussed in this section make the allocation even more difficult. It also becomes clear, that variety related costs cannot be investigated on a product basis, but rather require to move up with the analysis at least to the level of product families. As a consequence, product architecture features that allow variety reduction across the product family are advantageous from the cost standpoint (dimension function-component allocation). In addition, architectural characteristics that allow re-organization of the production process (re-sequencing) to reduce volatilities can contribute to higher production performance and, in turn, lower unit cost (interface reversibility and standardization).

4.3.2.2.3 Interdependence effects: if cost drivers affect each other

When determining what affects overhead costs, typically cost drivers are identified that are treated (within limits) as independent from each other. In reality, however, some cost drivers are associated with other cost drivers. That is, the effect of changing one cost driver may depend on the level of the other cost drivers. For example, Datar et al. find in their study of an automotive lamp manufacturer that supervision, tool maintenance, quality control and scrap are not only driven by exogenous product and process variables, but also affect each other (Datar et al. 1993). In other words, the overhead cost determination must take into account that the non-linearity of the relationship between overhead activity and associated overhead costs, is caused by other, likely non-linear relationships.

For the analysis of the effect of product architectural features on cost, this problem of overhead allocation creates yet another level of complication. These three levels are increasing in 'remoteness' from the product. For this reason the conclusions form the lower level are always valid for the higher (more complicated) level. The additional allocation problems added on each layer, is increasingly harder to track and to interpret in a generalizable way.

As a consequence, an analysis of cost implications of architectural differences must cover the 'lower' level and should provide at least general insights at the higher levels.

4.4 Setting the Analysis Boundaries with respect to Uncertainty

The third dimension in which the boundaries of the analysis have to be specified is the extent to which the processes are assumed to be constant and deterministic. This discussion touches upon two issues. First, it needs to be specified whether the results of the cost analysis are considered as being constant or static. For example, ramping-up manufacturing processes to bring them to full speed or learning processes may require dropping this 'static' assumption. Second, it needs to be clarified whether the analysis considers the processes as deterministic or stochastic. While the former mostly allows easier computations, the latter often reflects reality better. The trade-off between the extra effort on one hand and the better reflection of reality on the other, needs to be determined for every case.

4.4.1 Constant vs. non-constant production unit costs

The extent to which a cost analysis can be considered 'static' includes two cases: (1) one-time change followed by a stable period, and (2) change over longer periods of time. In the first case, the ratio of 'ramp-up period' to 'normal production period' is the determining factor. If, for example, the whole production run will extend over several years and the ramp-up takes only a few days, then the focus can be put on the system costs considering it in its static stage. In contrast, if the production run is relatively short and the ramp-up takes up a significant portion of it, then the systems cost are not well represented by the production run alone. This effect is equivalent to the set-up time for a single machine. If the lot-sizes are large and the set-up time is short, then the set-up cost can often be neglected. In contrast, when set-up time is significant relative to production time, it is inappropriate to use unit cost derived from production time only. In the real world production environment, ramp-cost can represent a significant cost portion. For example, it can take up to six months to bring an automotive assembly plant up to full production load (Almgren 2000). Analogous considerations can be made for many of the overhead costs.

Manufacturing cost changes over longer periods of time can occur in two ways. Either the change itself is constant or it varies as the production unit costs do. The case where the change is (to some extent) constant is often caused by what has come to be known as the learning curve effect. The argument is that with accumulating production volume workers and engineers are getting better in what they are doing. They optimize the processes and their work environment in a manner that the same work goes either faster or requires less resources, or both. In either case, productivity increases continuously. The rate often found is around 80%, i.e., with each doubling of production volume the production unit cost fall by 20%. Empirical evidence has been presented that this effect indeed exists. For example, Anderson, who investigates the effect of product mix heterogeneity in the textile industry, finds that learning to cope with frequent set-ups reduced the overhead costs for these operations (Anderson 1995). Activity-based costing systems can help to detect these learning effects (Andrade et al. 1999).

The second case of changing unit costs is characterized by a dynamic change. That is, unit costs do not change by a constant rate, but follow dynamic patterns. One example of this phenomenon are non-constant unit costs as a result of different ways of sequencing different products through jointly used production processes. Flexible manufacturing systems (FMS), for example, can manufacture different products on the same machine. The set-up time, however, may depend on what product has been produced prior to the one under consideration. Will the same tool be used? If not, is the tool change time dependent on what tool was used for the prior product? Similar effects can be observed in painting operations, where a change from a light color to a darker one is relatively easier to conduct as one from a dark color to a lighter one.[94] This problem has been addressed through the use of activity-based costing systems in conjunction with production planning models (Koltai et al. 2000). The model proposed by Koltai et al. allows a periodic update of the overhead allocation bases and rates for cost monitoring purposes.

4.4.2 Deterministic vs. stochastic production unit costs

In addition to the fact that the length of set-up and production processes may be sequence-dependent, they can also be stochastically distributed. This makes it more difficult to balance the production jobs. As a result, set-up production times are on average higher than if they are deterministically calculated. The strength of this effect increases the closer a plant operates at its capacity maximum. Banker et al., who investigate the behavior of relevant costs attributable to stochastiscity in production environments, find that ".. the addition of a product (and, in general, a strategy of product diversity) increases overall congestion in the plant and consequently results not only in queuing delays for the new product but also in increased delays for all existing products" (Banker et al. 1988, p. 189).[95] For automotive assembly lines, it has been found that through pre-assembling subsystems off the main assembly line, the variation on the main line could be reduced (Kinutani 1997).

Although not completely, the problem of stochasticity of real production, set-up, and logistics processes is to some extent tied to the costs that are incurred by product variety as discussed in section 4.3.2.2.2.

With respect to the effects of product architecture choice on unit cost, the phenomenon described in this section cannot be determined with the product alone, but requires assumptions

---

[94] Darlington recommends using simulation tools to approach these kinds of scheduling problems (Darlington 1999). However, his example implicitly assumes that the relationship between the input factors and costs are known.

[95] Banker et al.'s analysis focus on how stochastic behavior of set-up and production processes results in manufacturing lead time extensions, and, thus, in higher work-in-process (WIP) carrying costs. (Banker et al. 1988)

(or data) on the production environment, including information on scheduling and the production program.

4.5 Chapter Summary

This chapter has discussed the major issues important for a cost analysis of the impact of modularity (product architecture characteristics). Three major areas have been identified as critical for conducting cost estimations and interpreting the results. First, costs that occur along a product's life vary in absolute and relative size, timely occurrence and distribution. These variations depend on product characteristics like size, lifetime, complexity, etc. The choice of a product architecture over another is likely to affect these lifetime costs. The size of these effects varies from phase to phase of the product's life cycle. Second, the choice of the analysis frame is implicitly defining what cost effects that are triggered by product architectural choices will be visible. The more the cost estimation is conducted close to the manufacturing process, the higher the accuracy of these direct costs tends to be, but the more limited is the visibility of indirect cost effects through the enterprise or the value chain. In fact, particularly variety-related cost impacts and interdependence-related cost impacts can only be made visible if the product's environment is taken into account (product family, production processes, etc.). Third, the choice of deterministic versus stochastic cost estimation should balance the additional information gain with the increased effort to collect data and to conduct the analysis.

This chapter has created the background for the analysis and selection of costing methods to measure cost effects caused by differences in product architecture. The costing methods are at the focus of the next chapter.

5. COSTING TECHNIQUES – OVERVIEW AND ASSESSMENT

> “The correct balance between satisfying functionality with four common parts versus one unique part cannot be determined a priori. Designers need a comprehensive cost model to balance the trade-off involved.” (Kaplan and Cooper 1998, p. 214)

5.1 Chapter Introduction

The discussion of the framing issues for cost analyses in chapter 4 has demonstrated the strong impact the boundary setting can have on the interpretation of the results. But even within defined boundaries it is not a simple task to construct models that can relate specific product features (such as product architecture characteristics) to costs. At the same time, however, early in the design is precisely when this information is most valuable. The problem, how to provide information when little data is available is indeed a tricky one. Disciplines as diverse as engineering, operations management, or accounting have attacked this question from different angles. Although the methods vary, they all attempt to provide information on cost implication early on in the product’s life.

Every product, every system accumulates costs over its lifetime through design, production, operation, and retirement. The precise shape of the life cycle cost occurrence curve varies with product life, product value, and modes of operation (see chapter 4 for a more in depth discussion of this phenomenon). For all products, however, it is clear that costs are committed *before* they occur. It is generally believed that a relatively large portion of the total life cycle cost is committed in the earliest life cycle stages, particularly during concept and system-level design (Figure 5-1).[96]

---

[96] Although accounts vary, as a rule of thumb it is often claimed that about 80% of the total product costs are committed during the conceptual stage of product development/design. Various authors present the idea that anywhere between 60% and 90% of the total life cycle cost are committed during design. Interestingly, although these numbers are used by a variety of authors from fields as diverse as accounting to engineering to management (e.g., Smith and Reinertsen 1991, p.100, Anderson and Sedatole 1998 p.231, Blanchard and Fabrycky 1998 p.561, Clancy 1998 p.25, Knight 1998 p.21, Sands et al. 1998 p. 118; Buede 2000 p.7, Weustink et al. 2000 p.1, Bhimani and Muelder 2001 p.28), none of them backs his or her claim with real data. (The only exception - Ulrich and

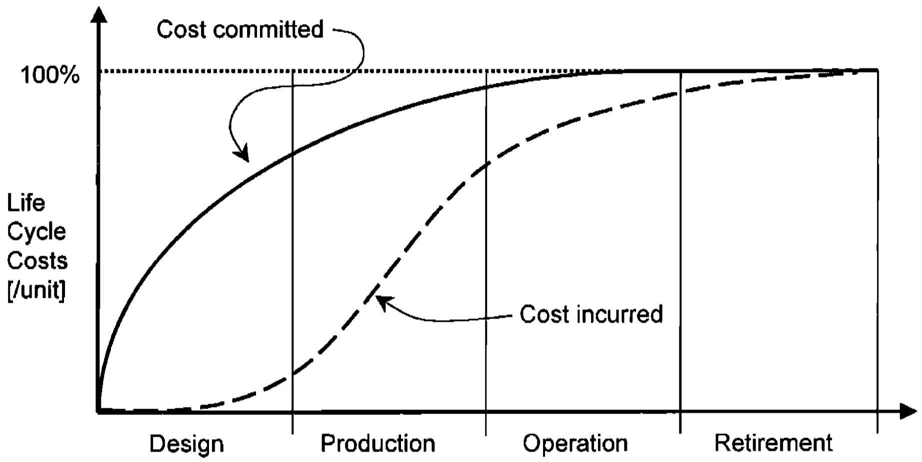


Figure 5-1: Cost committal and cost incurrence curves

If one understands the type of product architecture as a fundamental design choice early in the design and development process, then the knowledge about the cost implications throughout the product life of this choice is particularly valuable due to its leverage potential. At the same time, however, this knowledge is also very difficult to obtain at this early stage due to the lack of data.[97]

Despite this difficulty, there exists a considerable amount of modeling work to support designers with this type of information. Generally speaking, the goal is to predict the shape of the life cycle cost occurrence curve as a function of design decisions.[98] Due to the complexity of

---

Pearson 1998 - also does not specify portions of life cycle cost, but rather assess the cost influence potential of the designer versus the one of the production.) Others support the claim qualitatively that early decisions have strong effects on cost occurrences later in the process, e.g.,: "Product designers can greatly reduce the costs of products by explicitly considering the implications of their decisions for the manufacturing, marketing, distribution, and customer-service cost areas." (Horngren and Foster 1991, p. 411).

[97] The classic dilemma for cost assessment in early concept stages is characterized by the situation in which the engineer has very little information while (at least theoretically) having the biggest opportunity to influence cost. Obviously, in early concept or design stages of a product there is a lack of what is supposed to be the result of the development process: a detailed description of product and production process. This problem is the more difficult the less known the new technology or product is (Clancy 1998). For this reason cost analyses techniques that help understanding better the ways in which costs are influenced by early decisions have gained increasing attention.

[98] The question what determines the shape of the cost committal curve is an alternative research question. In cases of long time horizons and high risk situations (e.g., defense or space projects), the question about the shape of the cost committal curve and whether and how it can be altered (e.g., delaying decisions) can be of particular interest.

this problem various types of costing techniques and cost models have been developed from various disciplines.[99] The next three sections present the current status of costing techniques. Section 5.2 discusses cost allocation techniques and design guidelines as representatives for situations where data and cases are available. Section 4.3 follows to discuss and analyze advantages and disadvantages of different cost prediction models and methods developed by different disciplines. Section 5.4 concludes and summarizes the analyses of the applicability of the different methods to estimate cost effects of differences in product architecture. Figure 5-2 illustrates how this chapter forms together with the three preceding ones the background for the case-based experimental cost calculations.

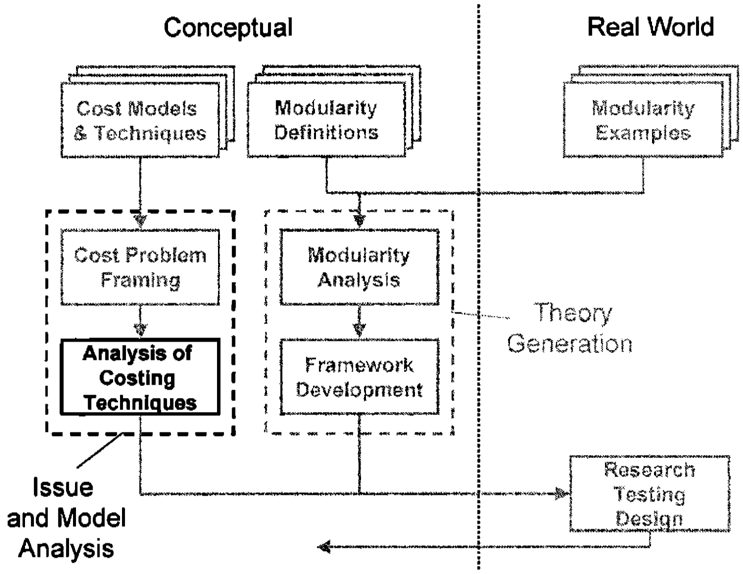


Figure 5-2: Content of costing technique chapter in relation to previous chapters

---

[99] Some argue that cost discussions need to be embedded in larger problems. For example, Bode and Fung have suggested introducing cost as a constraint in customer satisfaction maximization efforts. They normalize the relationship matrix of the house of quality and model the decision process as a linear program to optimize the resource allocation (Bode and Fung 1998). Others have suggested using cost as one constraint in a linear programming for the optimal assignment of modular product configurations to customers. Tönshoff et al., for example, develop a model that integrates four decision problems into a single model (Toenshoff et al. 1999). The four steps module design, module selection, bundle pricing and bundle pricing under demand certainty are formulated as a mixed-integer linear program. They find that an optimal bundling and pricing strategy based on a modular product structure can increase profitability. It remains unclear, however, how the cost functions respond to variations of the product architecture as a whole. Since the latter is precisely the focus of this dissertation, I restrict the analysis to costs to limit the complexity.

5.2 Cost Allocation Techniques and Design Guidelines

To establish the link between design and cost information, a multitude of different cost models and guidelines have been developed. They vary in their approach and their focus; they vary in the extent to which they require detailed data, and in their general applicability.

If the search to find the link mentioned above is understood as a sequence of steps over time (Figure 5-3), the beginning is the observation and collection of product and cost data after their occurrence. The next step is the search for relationships between product features, process characteristics, etc. on one hand and cost on the other. Once these relationships are understood they can be used to develop design rules and guidelines to steer designers to decisions that minimize costs. In this last step, the timely occurrence is reversed compared to the cost allocation in the beginning, i.e., the guidelines exist *before* they are used to create design data.

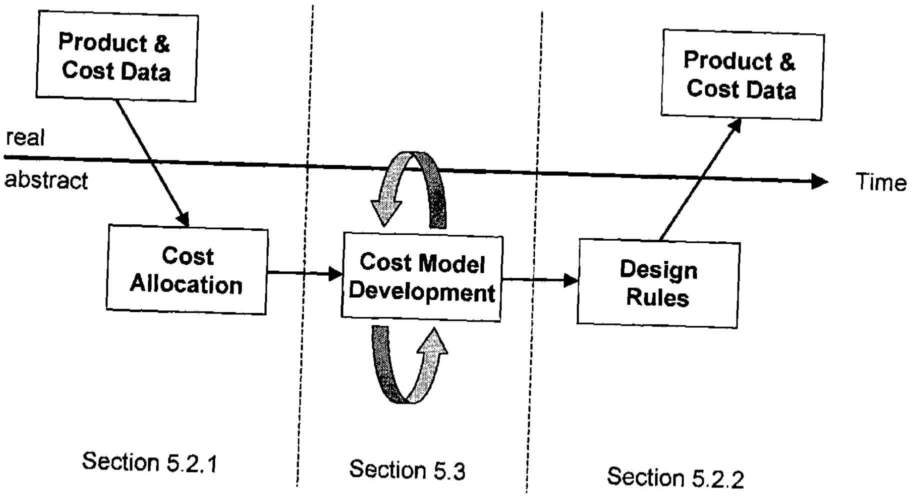


Figure 5-3: Cost allocation, cost model development, and design rules

In this section (5.2), I will describe the beginning and the end of this chain. First, I briefly describe and compare cost allocation methods (5.2.1), because their thinking is important for the understanding of the design rules and guidelines as well as many of the cost prediction and modeling techniques. Second, I will turn to what I call prescriptive methods. In their current form, and based on larger case bases, they have developed into a variety of design rules, product development management guidelines, and evaluation techniques. The middle part, the search for and modeling of the relationships between design and cost, is subject to the following section (4.3).

5.2.1 Brief review of cost allocation methods

Cost allocation methods are concerned with the way costs are attributed to entities of various kinds such as products, departments, facilities, or programs. While their purpose is less cost prediction than allocation after the costs occurred, the understanding of these methods is fundamental for thinking about prediction methods.

The category of cost allocation methods consists of two major cost monitoring, or cost accounting, methods. One emerged from the industrialization about hundred years ago, while the second has been developed over the last 15 years. The first is termed here 'conventional,' whereas the second has been named 'activity-based costing.'

5.2.1.1 Conventional method

Management accounting systems have been first developed in the period of the scientific management movement at the end of the 19th and the beginning of the 20th century (Kaplan 1991). Mass production became the prevalent form of production and industrial engineers attempted to standardize and simplify production processes to increase efficiency in the use of direct labor and materials. Labor-intensive processes and little product variety made this a sensible approach. As a consequence, costing systems developed that focused on measuring the direct costs, such as labor and materials, and used these measures also as a base to distribute resource consumption for supporting activities. Typically mark-ups on labor costs were defined to allocate indirect costs to all products produced. These systems calculated percentages that were added on direct labor and material costs for product costing purposes.

5.2.1.2 Activity based costing (ABC) method

With modern production technologies, product costs became less and less dependent on direct costs like labor and material, but more on indirect costs as various types of overheads. In addition, increasing buyer power resulted in higher levels of product variety. As a result, the activities performed by many resources were not demanded in proportion of the units produced. Thus, the unit costs produced by the conventional management accounting systems showed more and more distortions. To counter these drawbacks, activity-based costing was developed in the 1980s and 1990s (Cooper and Kaplan 1988, Kaplan 1991, Cooper and Kaplan 1992, Kaplan and Cooper 1998).

Fundamental idea of activity-based costing is to calculate the cost of activities that serve as cost drivers and 'charge' products with the time with which they consume an activity times the use rate per time unit. The cost drivers can be on various levels in the firm: "While some activity cost drivers are unit-related (such as machine and labor hours), as conventionally assumed, many activity cost drivers are batch-related, product-sustaining, and customer-sustaining" (Cooper and Kaplan 1992, p. 4).

Two decisions have to be made when developing an activity-based cost system. First, the number of cost driver needs to be determined, and second, which cost driver to use must be specified. These decisions are not independent from each other, because "the type of cost drivers selected change the number of drivers required to achieve a desired level of accuracy." (Cooper 1989a and b). It has been suggested to balance the accuracy level with the information collecting and processing costs, to provide an optimal selection of the cost drivers (Babad and Balachandran 1993).

When facing significant product variety, conventional costing systems tend to undercost complex products and overcost simple products. Various researchers have demonstrated that activity-based costing is advantageous to traditional costing approaches in determining product costs in multi-product environments (e.g., Banker et al. 1990, Hundal 1997, Clancy 1998). The differences in costs the different systems produce can be quite significant. For example, in their study of a company manufacturing automotive lamps, Datar et al. find that the plant's conventional cost accounting system allocated about 10% of total factory costs (i.e., on average $0.50 of $5 per product) for material handling (Datar et al. 1991). With help of activity-based-costing systems that use number of moves and distance per move as cost drivers, they show that material handling costs actually vary between $0.90 and $0.23 per product.

As briefly discussed in the previous chapter, the question of whether a cost can be considered fixed or variable is dependent on the choice of the time horizon. ABC, which attempts to make most if not all cost variable, is a tool directed to long-run decisions that are rather strategic in nature. This preposition is true for the product architecture choice for most products.[100]

Other potential problems can occur with activity-based costing systems just as well as with conventional ones. For example, the choice of interest and depreciation rates carries for both systems equally the potential for cost distortions.[101] Doran and Dowd point out that even ABC-costing systems potentially generate product cost distortions when depreciation and amortization costs from the company's financial system are used (Doran and Dowd 1999). They suggest understanding the capital cost as economic opportunity cost. Often, however, financial depreciation rates do not reflect "a reasonable proxy for opportunity cost." (p. 36) If fully depreciated assets are used, or undercharging for an internally developed process occurs, the resulting product cost are likely to be underestimated. On the other hand, if for tax purposes the shortest possible cost recovery period is chosen, the result may be too high product costs in the first periods of production.

5.2.2 Design rules and product develop management guidelines

This section's discussion encompasses techniques that are built on knowledge distilled from a large number of cases analyzed in the past. These techniques range from relatively focused design rules such as the DFX-family to more generic and higher-level product development management guidelines like target costing and value engineering.

---

[100] ABC has been criticized as guiding to poor short-run decisions. Another concept that has been developed to allocate costs for the short-run is the Theory-of-Constraints (TOC). First introduced in the popular book "The Goal" (Goldratt and Cox 1984), TOC originates from work close to operations like production scheduling. The Theory-of-constraints assumes all costs other than direct material as fixed. Then, to maximize profitability, the throughput must be maximized. The theory-of-constraints promotes to find the bottleneck in the existing system and adjust all other production to it to eliminate inventory. In the debate about whether ABC or TOC is the superior way of interpreting costs, various authors argue to understand both methods as the opposing end of a continuum with respect to planning time horizon: ABC for long-range planning, TOC for short-term decisions (Fritzsch 1997, Kee 1998). For the case of product-mix decisions a general model has been proposed that has ABC and TOC as its special cases (Kee and Schmidt 2000). Others suggest defining three categories of costs: resource costs, activity costs, and factor costs (Yu-Lee 2001b). Resource costs are here defined as having no association with production volume while factors costs vary directly with production volume. What Yu-Lee calls activity costs follows a narrower definition by separating the resources costs into an own category.

[101] See also the previous chapter (4) for a discussion of the time value of money for cost calculations.

5.2.2.1 Prescriptive methods (design-for-X rules)

If cost allocation methods (as the understanding of how to allocate costs that already occurred to a product or a department) are at one end of the continuum that links cost and design decisions, design guidelines are at the other. Design guidelines are prescriptive methods that use codified knowledge gained from experience in other cases for the application of new ones.

Over the last two decades, these design 'optimization' techniques have been proposed to support designers in achieving better designs for various foci.[102] Each of these techniques focuses on reducing a subset of the total costs (Figure 5-4).

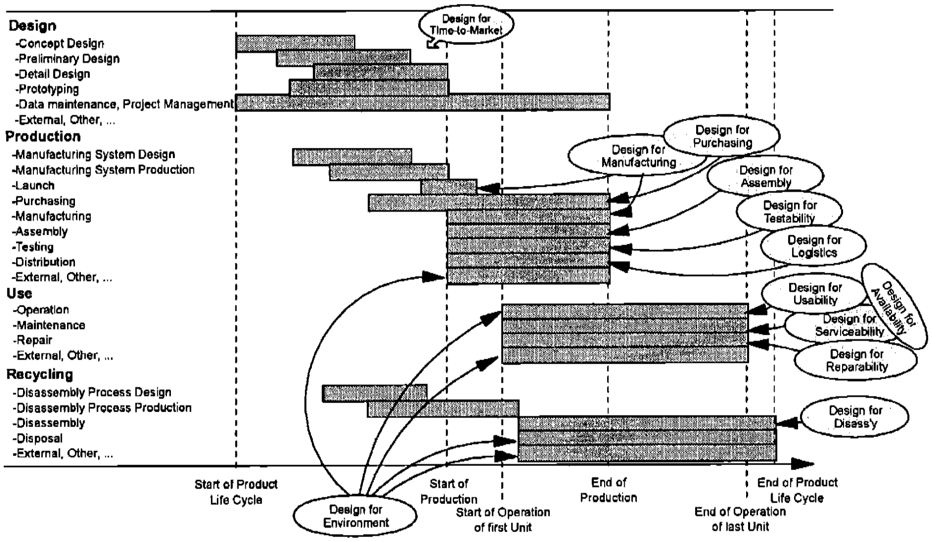


Figure 5-4: Various design-for-X techniques – each with its own focus[103]

Design-for-X guidelines have been developed to improve designs for a particular purpose, i.e., to reduce the cost of a particular downstream activity.[104] As such, they serve legitimate

---

[102] It is not always clear in what sense the term 'better design' is used. The two main representative, design for manufacturing (DFM) and design for assembly (DFA), in fact are guidelines to improve not the product with respect to its quality or performance, but to reduce manufacturing and assembly cost while maintaining product performance and quality. Ultimately, most design improvements can be translated into either cost reductions or performance improvements.

[103] Figure 5-4 shows only the most widely known DFX techniques. There are also other (albeit less often used) Design-for-X techniques; for example, Design for Security, Design for Safety, Design for the Physically Disadvantaged, etc. Like the others in the figure, they focus on a particular subset of product goals, too.

purposes for their specific application, but they are not well suited to answer the question of this research for two reasons.

First, their development was preceded by data collection to establish the link between design decisions and downstream effects of particular interest. Consequently, the DFX are already 'optimization' techniques whose constraints are aggregated in design rules. However, for the purpose of this research the knowledge about these links is more important than the design rules in their confounded form.

Second, each DFX technique aims at a particular cost block of the product life cycle cost. Therefore, the relevance of each technique hinges on the relevance of the specific cost block within the total cost picture and the particular perspective taken. In other words, if, for instance, the vast majority of the total costs incurs in the design phase, a design-for-reparability approach may not be the best use of the resources. Or, whether a design is 'optimized' for testability may only be relevant for a final customer if this has direct implications for her costs or the product quality.

Finally, not only the selection of which DFX-technique to use must be made, it is often unclear what the optimization according to one goal does to other interests. Examples are the sometimes competing objectives of design-for-manufacturing (e.g., simple part manufacturing) and design-for-assembly (e.g., few parts). To accommodate these multi-objective situations it has been suggested to combine the techniques to more comprehensive approaches. For example, an approach called *life cycle costing* has been suggested to take these various costs into account. Life cycle costing attempts to accommodate the calculation of those costs that occur over a product's lifetime (e.g., Sands et al. 1998, Bhimani and Muelder 2001). Ultimate goal is here too, to provide designers early on with information about the cost implications of their design choices for future product life phases (production, use, disposal). In its current form, however, life cycle costing is often rather cost monitoring, and thus, cost allocation, than estimation or prediction.

---

[104] The term *downstream* is here used for activities conducted with the product later in its life, i.e., production, use, or retirement.

5.2.2.2 Product development management guidelines and tools

In contrast to the design rules that have been developed to support the designer in making better decisions, this section briefly presents two concepts that are understood to guide the entire product development process: target costing and value engineering.

5.2.2.2.1 Target costing

Over the last couple of decades market dynamics have shifted a portion of the negotiation power from suppliers to buyers. In other words, increasing competition in most markets has increased the cost pressure for most firms. As a result, costing approaches have been developed to reflect these changes. Whereas classic costing and pricing schemes typically promoted an additive approach, i.e., starting from the basic manufacturing costs other costs for various types of overheads were added to this base (usually as a percentage mark-up), the most recent costing approaches demand to reverse the order of this process. Let market conditions determine the maximal price of the product the market 'allows,' and then work backwards to find ways to produce this product for not more than the 'allowable costs.' This process, which introduces market pressures into the design departments, has become known as Target Costing (e.g., Cooper and Chew 1996, Cooper and Slagmulder 1997, Mudge 1971).[105]

Essentially, target costing consists of three phases: market-driven costing, product-level target costing, and component-level target costing. Market-driven costing establishes an upper price bound the market under consideration is willing to pay for the product. The value perceived by the customers, the competitors' offerings, and the firm's own strategic objectives influence this first target costing phase. Its outcome is the target selling price. Subtracting the target profit margin from the target selling prices produces the allowable cost.

The second phase, product-level target costing, uses the allowable cost as its input together with current costs for an existing product. The difference between these two costs determines the target cost reduction objective.[106] The authors stress the importance of enforcing the cost target ('the cardinal rule'), which includes killing a project if it fails to achieve the targets. Many

---

[105] Other authors prefer the term *Design to Cost* to describe their approach, which is similarly directed to introduce costs concerns early in the design process (e.g., Burman 1998). However, since target costing is the more comprehensive technique, i.e., it is concerned with costs along the whole value chain, I have chosen it as the representative of the class of 'market-driven' design support techniques.

[106] Sometimes an additional strategic cost-reduction challenge is added which increases the difference between current cost and allowable cost.

of the decisions to determine these costs and to balance conflicting goals are expected to be made by the chief engineer.[107] As recommended tools to actually achieve the target cost, value engineering, quality function deployment, and design for manufacture and assembly (DFMA) are listed.

Finally, in the third target costing phase, component-level target costing, the cost goal is decomposed and assigned to subassemblies and components. With help of an intermediate step, function-level target costing, target costs on product-level are transformed into target costs on the component level. Function costing is based on historical cost-reduction rates or market analysis.[108] Component-level target costing is a process based on component cost history. There are two variants: functional analysis and productivity analysis. Functional analysis uses functional tables that relate historical values of functionality to the size of primary determinants (i.e., typically physical characteristics of the major components that make up the assembly). In a second step, the size of the primary determinant is then related to its costs. This second step appears disadvantageous if either the number of historical cases is small or the number of primary determinants is large, or both. Productivity analysis suggests decomposing the entire production process into process steps and to provide cost per process step via cost tables.[109] This in fact relates closely to detailed cost estimations, with all its advantages and disadvantages. The literature does not provide insight on how to do this beyond the direct manufacturing processes. The calculated component-level target costs finally receive a reality-check in negotiations with the suppliers. In sum, target costing can be understood as a comprehensive management method to maintain cost discipline throughout the product development process chain.

---

[107] "At Toyota, the chief engineer is expected to make his own decisions about where cost reduction is to occur. One of the objectives of the target costing system is to focus the attention of the design engineers in the design divisions in the right place" (Cooper and Slagmulder 1997, p.146). These foci of attention may have to be readjusted if market conditions or available technologies change.

[108] "For example, Isuzu uses monetary values or ratios to help set the target costs of major functions and asks customers to estimate how much they are willing to pay for a given function. These market-based estimates, tempered by other factors such as technical, safety, and legal considerations, often lead to adjustments to the prorated target costs. For example, if the prorated target cost for a component is too low to allow a safe version to be produced, the component's cost is increased, and the target cost of other components is decreased to compensate." (Cooper and Slagmulder 1997, p.145)

[109] Toyota makes heavy use of this kind of cost tables (Tanaka 1993).

5.2.2.2.2 Value engineering / Value analysis

Although value engineering is not completely distinct from target costing, is has a slightly different philosophy. In fact, Cooper and Slagmulder suggest using the tools of value engineering within their target cost framework during product-level target costing (see previous section).

Value engineering is a term that functions as an umbrella under which numerous cost reduction and value improvement techniques are subsumed. If the value a product delivers is understood as its functionality divided by its costs, then to improve the value either the functionality can be improved at identical costs, or the costs are reduced while maintaining competitive functionality. The latter is what the Society of American Value Engineers defines as Value Engineering:

> "Value Engineering is the systematic application of recognized techniques which identify the function of a product or service, establish a monetary value for that function, and provide the necessary function reliably at the lowest cost." (Mudge 1971, p.5)

As a consequence, value engineering promotes the combination of systematic analysis and creative methods to (a) evaluate the entirety of improvement possibilities and (b) try to find solutions previously overlooked. Accordingly, the tools used within the value engineering framework range from creativity techniques to quality circles to detailed cost analysis (Zentrum Wertanalyse 1995, Johnson 1997). Sometimes the application of Design for Assembly (DFA) is understood as part of the value engineering approach (e.g., Ramdas and Sawhney 2001).

Both cost allocation techniques as well as design rules and product development guidelines are important concepts and tools for learning about the link between design decisions and costs and applying this knowledge to design tasks. The middle part of this chain, however, is where the relationships are found and constructed. This part is the subject of the next section.

5.3 Cost Models and Modeling Techniques

This section provides the missing link between cost allocation techniques and design rules: the cost models and modeling techniques themselves. Common for all of these models is that they aim at finding and specifying the relationship between some product characteristics and costs in order to allow the prediction of costs for a product not yet built. I group the population of cost models loosely into two categories: top-down models and bottom-up models. The first group is characterized by techniques that require relatively large data sets and demand that the new product is relatively similar to those in the existing data set. Techniques in the second group require much smaller data sets for prediction. This second group includes two, quite different subsets: mathematical models and process-based modeling techniques. Since the boundaries of the two larger groups are not perfectly impermeable I will also briefly discuss some hybrid versions. The review of all models is conducted with respect to their applicability for the task to predict the cost implications of product architecture differences.

5.3.1 Top-down costing methods

Common element of all top-down approaches is the attempt to estimate or predict future costs without modeling in detail the product or the production process. Typically higher-level aggregated data is used to predict future costs, but the way in which the relationship between product or process characteristic is estimated varies. In general, methods in this group require relatively similar products and large data sets or both.

5.3.1.1 Parametric cost estimation

As the term indicates, parametric methods provide one or few parameters with which cost estimates can be inter- or extrapolated from known product/cost relationships to estimate the costs of the 'unknown' product. These methods have been developed with parameters on various levels of abstraction, from simple ones, like size, to complex ones, like features.

In its most simple appearance, a parametric method to predict cost is a 'rule of thump.' Examples are simple rules to calculate costs based on materials costs, adjusted by a fixed multiplier or other scaling factor ('mark-up'). Due to their simplicity, parametric methods are used in many different industries, especially for very early estimates. Various sources publish scaling factors for items such as equipment and installation for different industries, ranging from chemical processing industry (e.g., Uppal 1996) to construction (e.g., Young 1997) to ship building (e.g., Jasaitis Ennis et al. 1998) to the power equipment industry (e.g., Bielefeld and Rucklos 1992).

Typically, similar equipment of different capacity with known costs is used as a starting point. Next, a scaling factor, in its simplest way based on experience or 'intuition,' is applied to scale the cost from the known piece of equipment to the one with formerly unknown costs. The determination of the scaling factors can include more sophisticated methods such as regression analyses of various kinds.

The simplicity of this method is obviously its advantage. It makes it easy and quick to use. The downsides are its crude level of accuracy and the requirement of having cost information of some sort for similar equipment or products. In other words, only for items similar in kind (but different in capacity, for example) costs can be meaningfully estimated with this simple method.

5.3.1.2 Large scale empirical methods / Regression analysis

The most widespread used method to find the parameters that actually best approximate the impact on cost predictions are regression analyses. For prediction purposes, regression analyses can yield good estimates under two conditions: First, historical data has to be available, and second, the new design is not too different from those in the regression data set. The technique of regression analysis can also be used to arrive at good first-order estimates of certain cost components, for instance tool cost, within a larger effort to estimate the cost to produce a certain design (Clark et al. 1997).

Regression analysis can be conducted on various levels of sophistication, from simple linear regressions to multi-step, multi-variate regressions. To capture interaction effects, for example, Datar et al. suggest to simultaneously estimate the cost drivers (Datar et al. 1993). Using data from an automotive lamp manufacturer, they treat supervision, tool maintenance, quality control, and scrap as endogenous variables and product and process features as exogenous variables. Applying a two-stage least square procedure, they find that the endogenous variables are not independent from each other.

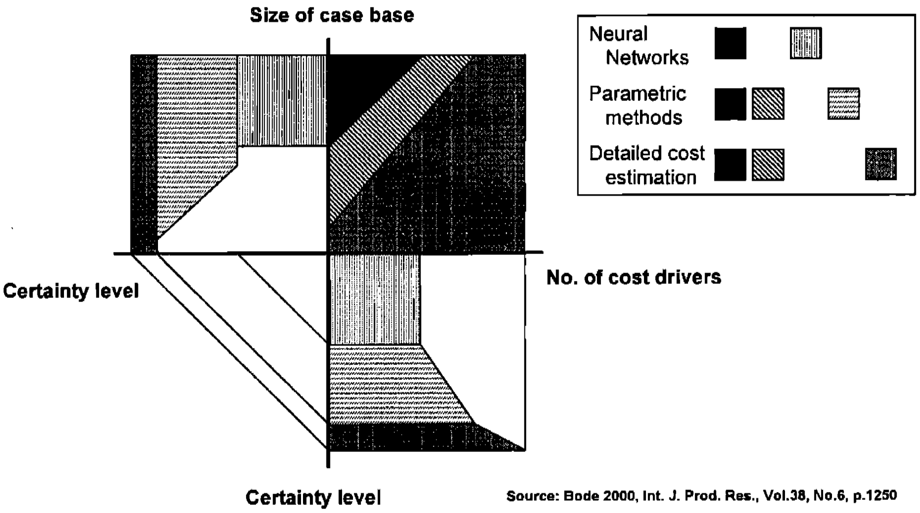
A sufficient number of data points are the prerequisite to arrive at statistically significant results. The level of variation should be large enough to allow the data explain something, but smaller than the population, otherwise the explanatory power decreases. For the same reason, the new design whose costs are to predict, must reasonably fit in the population.

5.3.1.3 Neural Networks

Over the last decade, neural networks have often been suggested as a better solution than standard regression analysis to find hidden relationships between independent variables and dependent variables, including costs. Shtub and Versano, for example, have developed a neural network to estimate costs for the manufacturing process of steel pipe bending (Shtub and Versano 1999). Based on five input parameters (number of bends, axes in space, inner and outer diameter, and distance form bending to end of pipe) they form four families with their existing data set. Using the method of leaving one out they make the network estimate the cost of the omitted pipe. By comparing these results with a regression analysis on the same data set, they find that the neural network outperforms the regression analysis with regards to the estimation accuracy.

In environments where the new product or systems is sufficiently similar to the case base, neural networks have been use to predict downstream consequences, such as costs for highway projects (Al-Tabtabai et al. 1999), or performance and costs for a shaft connecting a motor and two belts (Szykman 1996).

Neural-networks as non-parametric approximators have been identified to work well when the case base of similar cases is reasonably large, certainty exists which attributes have a cost effect, cost drivers are few, but it is unknown how the drivers influence cost (Bodė 2000). Neural networks share some of the drawbacks of regression analyses. They work very well as long as there is enough data to teach the system (Figure 5-5). Entirely new technology, however, or very different architectures for that matter, is characterized by a lack of historical data. This problem becomes even more complicated when the system is highly complex, that is the number of required input data is large.



Source: Bode 2000, Int. J. Prod. Res., Vol.38, No.6, p.1250

Figure 5-5: Application domains for neural networks compared to parametric methods[110]

5.3.1.4 Complexity theory based cost estimation

Another idea to arrive at factors describing relationships between product features and cost includes the use of complexity theory. Hoult and Meador, for example, suggest measuring the complexity of a product by using its dimensions and the associated tolerances (Hoult and Meador 1997). They argue that their complexity measure[111] multiplied by a constant, which is specific to the manufacturing process used, estimates the time required to manufacture the part. Together with the cost rate per time for using particular manufacturing equipment, this allows to calculate the manufacturing costs.

Using the complexity index as a scaling factor is intriguing, but it requires complete and precise data (e.g., specifying dimensions and tolerances), which makes the method difficult to use for prediction purposes. Also, the method requires relatively similar parameter types (e.g., dimension and tolerance) which limits its applicability for radically new cases.

---

[110] Extreme ends of axes mean 'large' and the center point means 'small'

[111] Hoult and Meador define a product's complexity as the sum of the logs of the ratios of each dimension to the tolerance associated to the dimension (Hoult and Meador 1997, p.720).

5.3.1.5 Expert Systems

Another type of top-down cost estimation methods can be described as pattern matching. Instead of finding parameters with which to explain (and predict) future costs, these methods use existing data for comparative purposes, i.e., they compare data known (or estimated) for the new products with existing data sets. The most widespread name for these types of approaches is expert system.

The use of historical case information, stored in databases, is the underlying idea of expert systems. With help of defining matching criteria it is possible to retrieve the case that best matches the new one for which the cost estimate is required. Rehman and Guenov suggest to structure the database hierarchically in order to allow designers to find best matching cases of every level, from product to subassembly to part (Rehman and Guenov 1998). It is likely, however, that this approach works better with existing designs and slight variations thereof than with entirely new designs. In addition, for the cost estimation of the changed design, the authors suggest an incremental and opportunistic strategy; they call it the 'blackboard approach.' Here the control resource detects the design changes and triggers actions in recalculating cost. Again, this approach seems less fitting the more the new design deviates from existing ones.

A comparable approach is suggested by Liebers and Kals (1997). They propose a cost decision support for product design that consists of several layers of aggregation. If possible, the highest level is used when applying templates, i.e., cost data from existing products that are sufficiently similar and adjusted for the differences. In cases where these are not available, the product is disaggregated into assemblies. On this lower level the step above is repeated, that is, if available, cost data for comparable assemblies are retrieved and adjusted for variations. Only if these templates are not available, the disaggregation continues down to the part level. Here the authors suggest to estimate part cost based on material and production cost.[112]

Following a similar line of thinking, Ten Brinke et al. develop a model that defines a product structure in terms of elements and relations. They link the elements to cost information via cost drivers and use this information to compare products (Ten Brinke et al. 2000). Another example for expert systems is the one developed by Chan and Lewis. For simple products, and a limited set of processes, their computer-based model uses stored data together with user input to

---

[112] In their paper the authors consider these detailed cost estimations as beyond the scope of their study.

estimate part costs (Chan and Lewis 2000). For more complex cases, Chen and al. have designed a set of expert system modules (sic!) that are connected. The module concerned with costs has stored information about individual manufacturing processes such as drilling, turning, etc. and employs DFM rules for comparative purposes (Chen et al. 2001).

Although labeled 'functional cost analysis,' Pugh's tool to evaluate designs also resembles an expert system. He proposes a matrix listing all functions and all parts and the individual costs in each cell. Adding across parts then establishes function costs (Pugh 1990). Recognizing that this method requires detailed design data, he limits its use to a check on the designs evolving from design activities.

Common for all expert systems is to store data about products and/or processes that are sufficiently similar. Although the structuring into hierarchies to compare products or similar items on different levels (products, subassembly, parts) makes the method more flexible, it still is dependent on availability of historical data and provides no solution for the problem when radically new or different designs need to be investigated.

5.3.1.6 Feature based cost estimations

To use parameters on a more abstract level, it has been suggested to design feature-based costing methods. Hu and Poli have developed a cost estimating scheme to compare components manufactured by stamping or injection molding (Hu and Poli 1997a). They define three classes of features: design features, manufacturing features, and assembly features. Design features are generally functionally oriented and include primitive, add-on, and macro features. Manufacturing features in general are process oriented. They encompass basic, subsidiary, and side-action features. Finally, assembly features are generally both function and process oriented. Handling and insertion are assembly features. Using this structure they compare parts with equal functionality (equal design features) to determine manufacturing and assembly cost. In a case study, they compare a product made from five stamped parts with one made as one injection molded part. With strong emphasis on tooling cost, Hu and Poli develop guidelines that help to find a cost optimal decision, given certain ratios of tooling and assembly cost.

Although less explicit, Leibl, Hundal and Hoehne have developed a similar idea (Leibl et al. 1999). Their approach is also feature-based, i.e., they calculate the cost of a drilled hole or a joint. Databases with these basic data are linked to CAD systems to support designers in finding low-cost solutions. The authors describe that their program calculates the cost of the feature, compares the cost of different physical variations for the feature, and forecasts the cost in a sense that it uses rough geometry data as a base reference. As a test case, they use a very simple part.

Brimson proposes yet another variation of feature costing. He suggests to identify the product features, then to determine the activities required to produce these features and to determine the cost of each activity (Brimson 1998). These data are collected in the same way as an activity-based costing system does. Brimson recommends investigating whether the product characteristics will cause the process to vary and if so, to what extent. Final step is the association of product features and costs. The approach essentially declares product features as cost drivers and attempts to assess the sensitivity of the results when their parameters are changed.

In a similar approach, Greenwood has proposed mapping product features to manufacturing elements, combing these elements into operations, transforming these operations into manufacturing objects, and linking the objects to enterprise processes (Greenwood 1997). Like Brimson, he suggests using an activity-based costing procedure to determine the manufacturing object costs.

While feature-based costing uses more abstract product characteristics, the link to costs is often accomplished with expert systems, i.e., data stored for similar features. As such, it exhibits the same limitations.

5.3.2 Bottom-up costing methods

In contrast to the top-down approaches, the bottom-up methods are not restricted by the existence of larger data sets. This category of methods encompasses two quite different sub-groups, each of which is discussed below. The first uses abstract modeling to build cost models, and the second uses first principles physics to model – primarily – manufacturing processes to form the links between product characteristics and costs.

5.3.2.1 Abstract modeling methods

The sub-group of what I call abstract modeling methods is prevalent in the operations research and operations management world. The models are mathematical representations of the processes under consideration. Due to their origin, most of these models have been developed for the life-cycle phases with strong emphasis on operations, namely production and product development.

To my knowledge, most of these mathematical models have been applied to cases where the architectural differences are either rather simple or bounded such that they do not exceed a certain complexity level. Take, for example, the modeling of the production cost of different automotive wire harnesses to find the optimal level of commonality across a product family (Thonemann and Brandeau 2000). By assuming a non-differentiating function of the wire harnesses, the task becomes to find the trade-off between decreased fixed-costs due to larger lot sizes and increases variable unit costs, since some of the wire harnesses will now necessarily be 'overdesigned.' The level of architectural difference, and therefore modularity, is in the range that has been labeled *simple* in chapter 2. The rest of the product is unaffected by whatever variant is assembled to it (bounded by the minimal functionality).

Another example for these 'abstract cost models' focuses on logistics and supply chain costs, primarily inventory costs (Lee and Tang 1998). Here the product architecture is determined as having several features and the customer can choose variants of these features and combinations thereof. That implies a product architecture descriptions are kept on a combinatorial level, i.e., a *medium* level. The cost model then finds the optimal process sequence depending on the demand structure, or identifies the demand conditions under which a particular production and supply chain sequence is optimal. Implicit is the assumption, however, that the processes can be re-sequenced at will, i.e., a simplified notion of interface is applied.

In the realm of product development, products also have been modeled as not exceeding the *medium* level of architectural differentiation (see chapter 2). This notion allows to model the search for an 'optimal' process to design and develop a product as a succession of design chunks. The key focus then is on the sequence and clustering of the tasks as a function of their probability for rework (Eppinger et al. 1994), or how they can be related to each other, e.g., through overlapping (Roemer 2000).

Common characteristics for these types of cost models are their abstract views on product and process as well as on cost. For example, product differences along a feature like *quality* are modeled as 'high' vs. 'low,' processes are assumed to be interchangeable, and costs are assumed to be linear, concave or convex.

These models aim at allowing general insights in relationships, rather than detailed numbers. As such, these models are often very elegant, but are not very well suited to cope with the kind of multi-dimensional product architecture descriptions as developed in chapter 3.

5.3.2.2 Process-based cost models

Fundamental idea of process-based cost modeling is to capture the engineering information of manufacturing processes to understand their impact on cost. Manipulating design specifications or process operating conditions results in repercussions on costs. Process-based cost modeling works backwards through interrelated steps of transformation to arrive at the cost effects of engineering decisions:

> "A process-based cost model, like any other engineering process model, serves as a mathematical transformation, mapping a description of a process and its operating conditions to measures of process performance, in this case cost. Unfortunately, the measures of performance which are of most interest are rarely determined directly by those operating conditions that are most convenient to measure to manipulate. Therefore this transformation must be built up in a stepwise fashion repeating the following two tasks: (1) Isolating those factors which directly determined the metric being estimated and then (2) understanding how the magnitude of those factors are set by the process in question" (Kirchain 2001).

Therefore, the main task when building process-based cost models is to identify the relevant cost elements, to establish the contributing factors and to correlate process operations to cost of factor use. Process-based cost models have been developed for a number of manufacturing processes and on various levels of detail and sophistication.

Locascio has suggested a relatively simple version of process-based cost modeling. For the assembly of printed circuit boards she determines machinery and cycle time at each station required for a particular design (Locascio 1999). Machine time consumed translates into fix costs and labor time results in variable cost per board. Once she has determined the production equipment she proposes bottleneck analysis of the system to identify the design features that are most promising for cost reduction efforts. Using an activity-based costing approach, her method identifies resources consumed by a particular design. Since the designs are, however, very similar to each other, the method needs to compare only slight variations in the product configuration.

A more detailed version of process-based modeling has been developed at the Materials Systems Laboratory at MIT as Technical Cost Modeling (TCM). It has been used for assessing the competitive positions of various materials (e.g., Busch 1987, Han 1994, Clark et al. 1997, Kang 1998). Technical cost models are representations of production processes. They incorporate first principle engineering knowledge on how product and process choices impact process requirements, speed, and yield, and, therefore, the costs. As an example, consider the case of an automotive body part designed to be manufactured using a stamping process. Several parameters of the part (material, size, weight, complexity, etc.) determine the minimum size of the press required to manufacture this part. Similarly, the product features determine the type (and thus, costs) of the tool required. Together with additional process descriptions such as machine run rates and yield, the TCM forms a representation of the manufacturing process in question.

In other words, TCMs contain a model of the manufacturing process, that is, they have built-in the relationship between characteristic product features and manufacturing costs – for the manufacturing process they are intended to model. For this reason, they allow to assess future costs in two major directions. First, it is possible to assess the impact of design changes (e.g., part gauge) on manufacturing costs. Second, using what-if analyses it is possible to assess the impact of changed input factors on the overall part costs. For example, the impact of using lower wage labor can be easily accessed.

As TCMs are process-based models, they focus on the machine determinants, i.e., rather the direct costs. Based on engineering principles, TCMs are very good tools to understand direct costs closely related to the manufacturing process and variations of it affect these costs. They, however, have drawbacks, too. While they do not require a large case base, they do require some data or estimates about the product beyond labels as 'high quality' or 'low quality.' Like all bottom-up approaches, however, the data requirements tends to make it difficult to model higher levels of indirect costs.

5.3.3 Hybrid cost estimation methods

Some researchers have suggested combining different cost estimating approaches from the two groups. Proponents of the experts systems, for example, suggest to design the experts system in a hierarchically fashion. If on the product level no matching product can be found in the database, then the program starts searching on the assembly level. If this fails too, it does the same on the component level. If this is still unsuccessful, they propose to model the missing data using a bottom-up approach (Liebers and Kals 1997). Goal of this procedure is to restrict the time consuming modeling process to cases where absolutely necessary.

Ben-Arieh uses a database for his variant estimation approach to determine machine parameters, i.e., a set-up plan, and uses explicit cost computations to transform these time-plans into costs (Ben-Arieh 2000). The bulk of his program rests on the database that has stored a variety of parameter values of historical cases. His analysis of rotational parts focuses on direct manufacturing costs only.

Hybrid approaches seem to offer valuable advantages but need also to be carefully designed for their intended purpose.

5.4 Chapter Summary

This chapter has reviewed existing cost allocation techniques and design guidelines, as well as multiple cost modeling techniques. The cost allocation techniques form the underlying basis for the remainder of the discussion. Design rules (DFX, etc.) and product development management guidelines (target costing, value engineering, etc.) are discussed as the possible outcome of cost modeling work.

The cost models and modeling techniques can be separated into top-down and bottom-up approaches. Top-down approaches attempt estimating the cost of a new product through either statistical analyses or various pattern matching techniques based on past experiences. They typically require that the case base is rather large, the deviation among the cases not too big, and the number of cost drivers limited. Bottom-up approaches use either abstract modeling or cost analyses built on the process physics of manufacturing processes. Biggest drawback of the abstract modeling is that they mostly do not allow the architectures to differ too far from another. In contrast, process-based cost models allow larger differences along this criterion, but require comparatively more detailed data. Figure 5-6 summarizes the findings.

Costing Techniques	Application Criteria	Data Set Requirement (minimum case base)	Acceptable Number of Cost Drivers	Acceptable Difference in Architecture Decomposition	Required Certainty of Data Input
Top-down costing methods	Parametric	Large	Low	Small	Medium
	Regression Analysis	Large	Low	Small	Medium
	Neural Networks	Large	Low	Small	Medium
	Complexity-theory based	Medium	Low	Small	High
	Expert systems	Large	Medium	Medium	High
	Feature-based	Medium	Low	Small	High
Bottom-up costing methods	Abstract modeling	Small	Small	Small	None
	Process-based cost models	Small	Medium	Large	Medium

Figure 5-6: Application domains for various costing techniques

Based on the modularity and product architecture analyses (chapters 2 and 3) as well as the analyses of costing techniques and costing technique capabilities (chapters 4 and 5), the following chapter will design the research study, defines its boundaries, and select its subjects.

6. RESEARCH DESIGN FOR TESTING THE MODULARITY FRAMEWORK

6.1 Chapter Introduction

The previous chapters have provided this research project with the two necessary ingredients. First, the modularity analysis has resulted in the development of a framework that allows describing the multiple aspects of modularity. Second, the analysis of costs that occur in various life cycle stages together with the analysis of existing costing techniques have laid the groundwork what costs to look for and how to measure them. The third and final step, to establish the link between product architecture and costs is core of this and the next chapter. This sixth chapter lays out how the actual study, which is presented in chapter 7, has been designed and how the research subjects were selected. Figure 6-1 presents a schematic overview of this chapter.

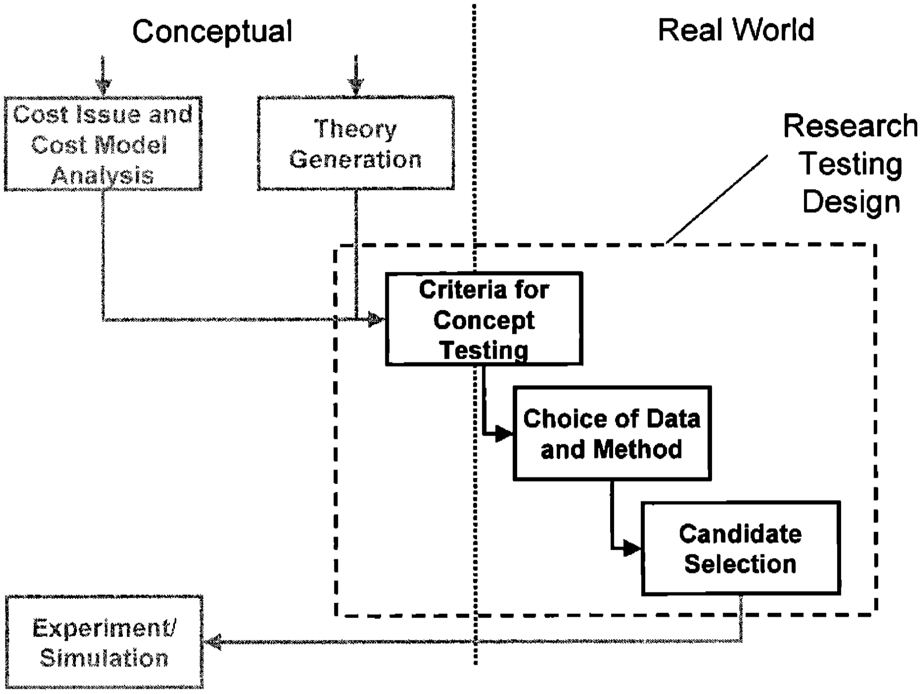


Figure 6-1: Content of research design chapter

6.2 Important Criteria for Concept (Framework) Testing

The main goal of this thesis is to establish a better understanding of the link between modularity and costs. As the framework developed in chapter 3 has demonstrated, modularity has in fact multiple faces and can be much better represented by the multi-dimensional construct 'product architecture.' As a consequence, the method to investigate the link to cost then must accommodate this characteristic of the construct. In particular, it must capture the variations in the individual dimensions of the construct, ensure comparability of the outputs, and simultaneously allow controlling for (almost) all other factors.

6.2.1 Independent variables: Multidimensionality of 'product architecture'

The multidimensional nature of *product architecture* presents the researcher with a relatively high number of cost drivers. In addition, the fact that costs arise throughout a product's life cycle and that costing techniques are complicated through multiple non-linear relationships makes the investigation of the link between product architecture differences and costs even more difficult. In other words, in order to test the framework developed in chapter 3, the research design must enable to investigate the relationship between architectural differences and cost in great detail. As a consequence, the type of investigation is somewhat exploratory in character.

6.2.2 Dependent variable: Comparability of cost outputs

A second factor to be observed in order to establish a link between product architecture and cost is the comparability of cost outputs. Just as the framework developed in chapter 3 allows measurements along the individual dimensions that are comparable across products, the method to test the framework must produce output data, i.e., cost data that are comparable. This could include considering rates for currency exchanges or inflation, when comparing cost outputs of different geographic regions or over time.

6.2.3 Control factors: Isolate 'product architecture'

While the research design should allow investigating individual dimensions of *product architecture,* simultaneously it must be possible to control for (almost) every other factor that may cause cost differences. Two major issues deserve consideration: the comparability of products and non-product related factors, i.e., those external in nature.

To isolate the link between the dimensions of product architecture and costs, one needs to be able to control for functional differences across the products. For if the allocation of function to components is part of the product architecture description, differences in functionality complicate or even dilute the comparability. Likewise, the research design for testing the product architecture framework must allow to control for all input factors that affect the output costs and but have no direct ties to architectural differences. Examples are differences in wage rates or operational performance across factories.

In sum, the method to test the framework must enable the separation of the 'independent variables' from each other, must ensure the comparability of the 'dependent variable,' and must enable to control for all other possible influences.

6.3 Research Design

6.3.1 Statistical analyses vs. case studies

Comparisons of the merits and disadvantages of statistical studies compared to case studies have been conducted and discussed in various fields in academia. Superior statistic precision, reliability, and testability of statistical analyses are typically juxtaposed with the advantages in relevance, understanding, and exploratory depth of case studies (Meredith 1998).

It has been argued, that these two prototypes of research methods are rather complementary than competing approaches. If research is in its early stage with respect to a phenomenon, theory or construct, it is necessarily exploratory in character. This early stage is rather a phase of hypothesis generation than one of hypothesis testing (McCutcheon and Meredith 1993, Malhotra and Grover 1998). As a consequence of the large number of possible effectors relative to the level of depth of understanding of the linkages between the effectors and the output, most researchers typically apply case-based or qualitative methods rather than large-scale quantitative methods like statistical analysis. The set of methods suited better for early stage research encompasses methods case studies, exploratory surveys, or experiments.

In contrast, in later stages, when the role of effectors is better understood and the number of 'comparable' cases has increased into a region where statistical instruments can produce significant results, large-scale empirical studies with statistical analyses become the more predominant form of research. Data collection for this research approach is typically conducted in survey form.

Survey-based empirical studies investigating the implications of modularity are rare, and those that exist use rather simple constructs for modularity. For example, Duray et. al test empirically the use of different manufacturing strategies and tactics of companies using different levels of customer involvement and different levels of modularity (Duray et al. 2000).[113] Their modularity measure consists of two factors, one representing 'modularity through fabrication,' and the other 'modularity through standardization.' The first factor measures with five questions whether a customer can actually influence the product design, e.g., "Each customer order requires a unique design; customer can specify new product features." The second factors determines whether customers orders can be assembled from pre-defined (and pre-designed and pre-fabricated) designs, e.g., "Each customer order is assembled from components in stock; customers can select features from listings." Although Duray et al. receive statistically reliable results, it is unclear how this rather broad notion of modularity can be translated into design advice. Categorizing each company into one of four archetypes and using perception-based measures for financial performance, Duray et al. find some relationship between their modularity measure and performance, but not which sub-dimension of modularity affects which performance result.[114]

To avoid having to measure performance in monetary terms, other survey-based empirical studies have selected labor hours as performance measure, either directly for product development stages (Clark and Fujimoto 1991), or adjusted for invested capital for production stages (Womack et al. 1990).

6.3.2 Research design: Case studies as controlled experiments

In addition to the criteria discussed in section 6.2 of this chapter, the research design to test the product architecture framework builds on the insight on costing techniques gained in chapter 5. There, two groups, top-down and bottom-up, had been analyzed and compared with respect to

---

[113] Duray et al. employ multiple methods in their research on modularity, encompassing both case studies and survey-based empirical work.

[114] Duray et al. state: "Although both high and low performers are found among all mass customization types, we do discern better business performance among the types that use standard modules and employ modularity in the later sates of the production cycle." (Duray et al. 2000, p.623)

their requirements for data set size, acceptable number of cost drivers, coverage of product architecture differences and certainty of data input.

Overlaying these requirements with those for independent variables, dependent variables, and controls, shows that survey-based empirical studies clearly neither capture the richness of the multidimensional construct of product architecture, nor permit to establish the architecture-cost link in an unambiguous way.

As a consequence, I choose to construct a portfolio of cases to investigate the impact of architectural differences along the individual dimensions. A bottom-up research approach through case studies allows modeling a far greater number of variables more accurately, but for much fewer cases. Each of the cases will be both described in detail with help of the product architecture framework from chapter 3, and will be modeled in detail with help of process-based cost modeling methods. The cases will be selected such that they provide comparability along the products' functions and sufficient variety across cases with respect to product architecture.

Note that I use the term *case study* to describe cases in an experiment style; as opposed to the way the term is used in most social science disciplines where it describes a study that follows an individual event that unfolds in its natural environment. In contrast, in this research, each case represents a design, i.e., a design specific solution for a product. The use of process-based cost models then allows to understand each case like a laboratory experiment in which all external environment factors, like wage rates or taxes, can be completely controlled.

Obviously, the increase in internal validity, i.e., to find and to understand causality of relationships between architectural features and cost effects, is paid for by a decrease in external validity, i.e., to generalize the results to larger populations. However, the purpose of case studies is not to rely on statistical generalization (like survey research), but on analytical generalization (Yin 1994).[115] Consequently, I will select individual cases that differ along the dimensions of 'product architecture,' model the costs of each case with help of process-based cost models, and test the robustness of the results conducting sensitivity analyses. To do so, I select a candidate,

---

[115] "The external validity problem has been a major barrier in doing case studies. Critics typically state that single cases offer a poor basis for generalizing. However, such critics are implicitly contrasting the situation to survey research, in which a "sample" (if selected correctly) readily generalizes to a larger universe. *This analogy to samples and universe is incorrect when dealing with case studies.* This is because survey research relies on *statistical* generalization, whereas case studies (as with experiments) rely on *analytical* generalization. In analytical generalization, the investigator is striving to generalize a particular set of results to some broader theory." (Yin 1994, p.36, *italics his*)

i.e., a product, which allows me to define multiple instances, i.e., cases, of this product with the appropriate characteristics.

6.4 Candidate Selection

The selection of a suitable product as a candidate for this study is guided by the idea to mimic laboratory experiments. A candidate must be represented by multiple versions of the product, i.e., each version of the product forms an individual case. The candidate selection process is framed by three criteria discussed below.

6.4.1 Selection criteria

The appropriate selection criteria are concerned with the comparability and the variation across the cases as well as with a practical aspect of this research: the data accessibility.

6.4.1.1 Comparability across cases

The first criterion for selecting a candidate as a research subject is how well it allows controlling for other factors during the experiment. Good controls are necessary to isolate the cost effects induced by differences in product architecture. While some factors are relatively easy to control, e.g., exogenous factors such as wage rates or taxes, others are relatively difficult to be kept entirely constant across the instances of a candidate, i.e., across the individual cases. The framework of chapter 3 analyzes the scheme by which the product's functionality is allocated to the individual components. For the purpose of the analysis of the cost effects, one step to control for comparability across cases is to analyze case instances with the same functionality. This equals to control for 'product functionality' across the experiments.

Total functionality, understood as the sum of all functions that a product or device offers, can represent functions ranging from lifting heavy loads for a fork lift to offer an aesthetically pleasing appearance for a coat. To insure apples-to-apples comparison across cases, the different cases should provide similar levels of functionality to the customer.

The level of functionality must consider three dimensions. First, and most simply, products representing the individual cases should provide the same functions. For example, if the candidate were an office chair, all cases had to provide the customer/user with functions like provide seating cushion, provide back support, or provide back adjustment. Second, since many products are themselves part of nested hierarchies (Christensen 1992a), it is possible that the product under consideration provides a function only jointly with other products, i.e., it provides only a fraction of the function. In these situations it must be determined if the cases under comparison deliver a comparable fraction of the particular function.

Third, and in some instances most difficult to assess, are functions that are provided on different performance levels across cases. If performance is understood as describing the level at which a function is provided, different product variants (cases) are likely to display different levels of performance. In some cases, the performance level can be exactly determined: the distance a car requires to decelerate from, say 60 mph to 0, is measurable and its value directly comparable across products. In other cases, the relationship is so complex that the level of performance is often described in qualitative terms. The driving feel of a car is a good example for this phenomenon. Finally, there are cases where personal taste is overwhelmingly responsible for the performance evaluation (e.g., color, shape).

Although the true performance functions are likely to complicate the problem by their non-linearity (5 meter difference in breaking distance may be less important between 60m and 55m than between 30m and 25m) and by the subjectivity of the assessors (one person may consider a particular ride feeling as hard, another one as medium), in many cases it is sufficient to establish a threshold that a function must pass to achieve the assessment being provided. Differences in performance beyond the threshold are then neglected. Every case, however, has to be carefully assessed for every major function whether this assumption can be made.

6.4.1.2 Sufficient variation of the independent variable across cases

In order to establish a meaningful relationship between an independent variable and a dependent variable, the selected cases should demonstrate sufficient variation of the independent variable, i.e., the individual dimensions of *product architecture*. Since a large number of variables limits the number of cases that can be studied, "it makes sense to choose cases such as extreme situations and polar types in which the process of interest is 'transparently observable.' Thus, the goal of theoretical sampling is to choose cases which are likely to replicate or extend the emergent theory." (Eisenhardt 1989, p.537) Consequently, the candidate (i.e., a product or subsystem) chosen for this research must exist in multiple cases, i.e., instances that vary sufficiently along the dimensions of *product architecture*.

6.4.1.3 Data accessibility

In academic reality, a factor always to be considered is the extent to which data are available to the researcher. This study requires two different types of data sets about the same products. First, to understand the different product architectures the products exhibit, technical information is required. This includes materials and geometric specifications, manufacturing process descriptions, and assembly sequence information. Second, to establish comparable cost estimating scenarios, data on production processes are required. Some of these data, for example exogenous data as wage rates and interest rates are modeled and calculated in a simulative fashion. Other data, for example variables that determine the speed of a machine, and in turn its throughput, are taken from real life data, for example survey-based industry averages.

A particular problem with regards to data accessibility is the cost data for purchased parts. Companies that sell complex assembled products, today often make only a small fraction of its parts and components. This means, that the cost of the purchased parts amounts to a substantial fraction of the total cost. Therefore, even without attempting to model every individual rivet, in order to arrive at a comparable cost number for the whole product, the cost data for the purchased parts must be accessible.

6.4.2 Case candidates

Six subsystems (or chunks) of the automobile have been pre-selected as candidates for case studies. This pre-selection is based on choosing a complexity level between as high as necessary to offer interesting variations and as low as possible to keep the analysis manageable. Each of these subsystems will be introduced below and the potential for serving as a candidate for case studies is assessed.

6.4.2.1 Chassis

The subsystem *chassis* offers two types that could be considered for this analysis: (a) a complete chassis or (b) so-called corner modules. Each type is discussed below.

In the early years of the automobile, i.e., at the beginning of the 20th century, the design of the vehicles followed their predecessors' design: automobiles were constructed like carriages. This meant having a chassis separate from the passenger cabin. While the design of most passenger cars departed from this in the 1950s with the introduction of the unibody concept (chassis and cabin are now formed by large panels), the design of trucks has kept the idea of separate body and chassis.[116] This is also the reason why for trucks a separate chassis can be supplied at once by a supplier (Kimberley 1999).

Determining the boundary of the subsystem *complete chassis* is relatively straightforward for the truck-style versions. Comparing them with car chassis, however, becomes very difficult because car chassis are part of the unibody, and therefore, hard to separate functionally. In addition, the performance levels of chassis for cars are not easy to compare. These problems are even more complicated for the corner modules. Most of the functionality that they provide they do so in concert with each other and the rest of the car. Thus, it is difficult to control for functionality and performance comparability.

With respect to variations of the product architecture across cases chassis do not exhibit large variations. In fact, the variations are rather small, given that most truck chassis follow a very similar concept of having two longitudinal rails connected by several crossbeams. The variation among the subsystem of a chassis, the corner modules, is on a medium level, exhibiting different cross-arm configurations and geometries.

Finally, data accessibility for both types, i.e., the whole chassis or the corner modules, was found to be limited.

6.4.2.2 Cylinder Heads

Every automobile propelled by an internal combustion engine has a cylinder head. Although the engines have always been the home ground for the OEMs, more recently the separate manufacturing and assembly of the cylinder heads has come under scrutiny (Brooke 2000a).

Cylinder heads itself are typically casting components, made from either aluminum or steel. These parts are then machined to produce the smooth surfaces for the gasket, that seal engine head and engine block. The criteria of comparability of functionality and variation in product architecture are tightly intertwined in case of cylinder heads. Perfect comparability would prohibit including additional adjacent components like air intake manifolds or valve-timing systems. At the same time, the valve train technology (rocker, camshaft position, fixed or

---

[116] Today, often the term body-on-frame is used in the auto industry.

variable timing, etc) is likely to be a major source of architectural variation, but it also threatens the comparability of the products in performance terms. In sum, the comparability is assessed as being medium, while the architectural variation of cylinder heads currently in production is rather small.

Finally, accessibility of cylinder head data for case studies is relatively poor.

6.4.2.3 Doors

The doors of modern automobiles include a number of different subsystems, i.e., a door is a relatively complex product; and itself a subsystem of a car. The boundary defining the functionality of an automobile's door is for most existing designs very similar.[117] This facilitates the comparability of different door designs with regard to their functionality and performance dimensions.

In the past, the doors of most automobiles currently in production have been built up using the same type of architecture, i.e., the variation has been rather small. The 'natural' characteristic of doors having only relatively weak but clear defined interfaces to the rest of the car, however, has made them prime candidates for new concepts. It is for this reason why more recently a number of door designs with different architectures have reached the development and small-scale production stages (Kochan 2001). A second reason for more recent development work on new door concepts rests on the fact that parts of the door represent interior, and the interior has been found as the area in which OEMs first move towards forming modules (McAlinden et al. 1999, Sako and Warburton 1999).

Data accessibility to some of these new door designs is evaluated as being on a medium level.

6.4.2.4 Front-Ends

When the car body design technology moved towards the unibody concept about 50 years ago, the front of most vehicles became a part of this design concept. More recently, however, it has been suggested to separate parts like bumper, radiator, headlights, and crossbeam and group them in so-called front-end modules (see Tajima and Yasugahira 1999, Anonymous 2001b).

---

[117] Some safety aspects may represent a dimension where the performance level of different doors may differ. This is because some car designers include the doors in the load-carrying path in case of a frontal accident while others do not.

Main driver for this separation often is to increase the ease of all assembly operations in the engine compartment (Whitney 1988).

The boundary definition for functionality and performance of front-end (modules) presents some difficulties, because functionality content changes if the boundary of the subsystem *front-end* is moved. For example, some discuss to divide the hood into two smaller covers and include the smaller one that would give access to oil and water reservoirs into the front-end (Moulin et al. 1999). Performance comparison problems occur particularly in the realm of safety. Not only is crash protection to be delivered by the entire body (plus chassis in case of trucks) but also performance levels are likely to be different for different designs. On the other hand, this analysis could make the assumption, that all versions of front-end designs in vehicles available for end customers pass a minimum safety threshold. Nevertheless, the safety aspect obviously plays a pivotal role for a front-end subsystem.

Given that a variety of technologies has been proposed for how to design and manufacture the cross-beam, front-ends may offer a sufficient level of architectural variation among them, although the principal architecture is likely to be structured around a beam-type carrier. On the whole, comparability and variation issues are evaluated as being on a medium level.

Data availability for different front-ends is assessed to be on a medium level.

6.4.2.5 Instrument Panels / Cockpits

Besides the subsystem door, the instrument panel has been the area where the OEMs have made the biggest steps towards re-defining product architectures by re-designing components and grouping them into so-called cockpit modules. Commonly, the instrument panel as a group of components includes the instrument cluster, HVAC[118] components, wiring harness, a top surface cover, a structural carrier, and sometimes the steering column.[119] More recently, some suppliers push towards getting the business for all of these components (e.g., Anonymous 1999e, Martin 1999b, Murphy 2000).

---

[118] HVAC is an acronym common in the auto industry and stands for Heating, Ventilation and Air Condition

[119] The use of the terms *instrument panel* and *cockpit* lacks somewhat of a clear definition in the industry. Most often the *cockpit* includes the instrument cluster plus controls, while the *instrument panel* encompasses the cockpit plus the injection molded panel itself, sometimes even the IP beam (a cross car beam that supports the instrument panel).

The functionality and performance boundaries for instrument panels can be established relatively clear with regards to most functions. Exceptions are most electronic functions, and the extent to which the carrier plays a role in vehicle structure beyond carrying the cluster, HVAC parts and top cover. In sum, the level of difficulty to separate the product from its upper-level system for a comparative analysis is assessed to be medium.

The relatively large number of components typically included in a cockpit module lets one assume that there exists considerable variation among the instrument panel population with regards to *product architectures.* However, this is not really the case because the tasks of the components differ significantly from each other. That makes it difficult to change, for instance, the part-function allocation. In some aspects, however, this is possible. For instance, it has been suggested to integrate HVAC ducts into the carrier (cross car beam) to reduce part count and thereby assembly cost (Dewhurst 2000).

The accessibility for data concerning instrument panels is assessed as being medium.

6.4.2.6 Roofs

Historically, the automobile's body design as a unibody has made the roof an integral part of the body-in-white. As a result, neither was the variation among different roofs large nor could they easily be separated in terms of function/part allocation. From the interior perspective, however, OEMs and supplier have been attempting to group headliner, visors and additional controls into so-called headliner-modules. More recently, a company has been reported of having developed stamping technology that allows stamping pre-painted steel panels without degrading surface quality (Buchholz 2000). Together with appropriate joining technique, this would allow to pre-assemble the entire roof separately and facilitate the analysis of a car roof as a separate component.[120] It would not, however, necessarily increase the variation in internal architectures. To summarize, the comparability of most existing designs is rather poor, while the variation is small.

Since the reported case is not in production yet, the data availability for roofs as a candidate for case studies is relatively poor.

---

[120] For a body design different from the classic unibody, a separate roof is already in production. The Smart, a two-seater produced by the European company Micro Compact Car (MCC) has a prefabricated and pre-assembled roof construction (Anonymous 1999d).

6.4.3 Candidate selection

Figure 6-2 summarizes the assessment of the case study candidates. *Doors* recommend themselves a candidate for case studies through their good comparability with regards to functionality and performance and the large level of variation of architectures. For these reasons, the door is selected as the candidate for experimental case studies to assess cost effects of different product architectures. The case studies are the subject of the next chapter.

Selection Criteria Candidates	Comparability of Functionality and Performance	Variation of Independent Variable	Data Accessibility
Chassis	Poor	Small/ Medium	Poor
Cylinder Heads	Medium	Small	Poor
Doors	Good	Large	Medium
Front-Ends	Medium	Medium	Medium
Instrument Panels	Medium	Medium	Medium
Roofs	Poor	Small	Poor

Figure 6-2: Overview candidate assessment

7. CASE STUDIES – LINKING PRODUCT ARCHITECTURE FEATURES TO COST

7.1 Chapter Introduction

The previous chapter resulted in the selection of automobile doors as the candidate best suited for testing the product architecture framework developed earlier. In this chapter, the actual experiments are conducted (Figure 7-1). To do so, multiple instances of the selected candidate are determined as 'experiments' and appropriate adjustments to guarantee comparability are made. Product architecture assessment and cost modeling form the core of this analysis. Sensitivity analyses link these two pieces together to gain insight in the architectural effects on costs. Finally, a brief discussion of the experiments concludes the chapter.

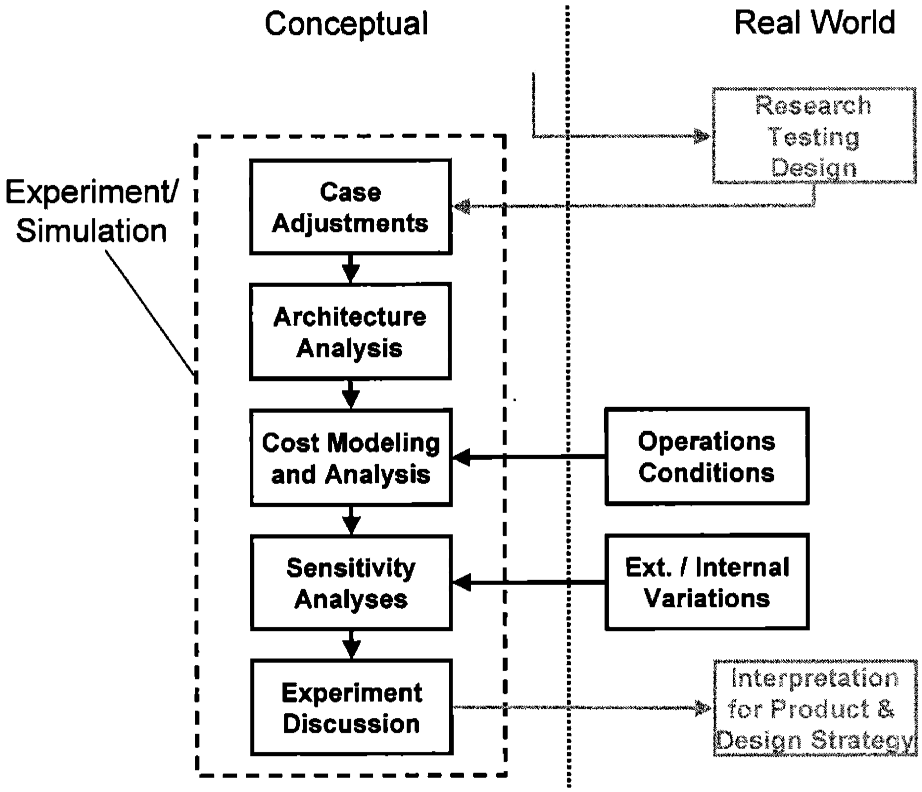


Figure 7-1: Content of case studies chapter

7.2 Overview: Automotive Door Designs

Most recently, the automotive industry has been very active in developing new door concepts. Because it is relatively easy to separate, a door forms a somewhat contained subsystem that can be used as a test bed for new designs to introduce, for example, lightweight materials or reduced assembly effort.

7.2.1 Recent door studies – door modules

Historically, the car body has been the responsibility of the OEM.[121] This included the stamping of the steel panels and their subsequent assembly. The structure of automobile doors followed an idea similar to the one for the body. Stamped steel panels were joined using welding and flange hemming processes. Suppliers typically have contributed components like window regulators, wiring harnesses, or trim panels for the final assembly process. The supplier involvement in these parts is one of the reasons why the first pre-assemblies that suppliers offered and called 'modules,' were developed around either the window regulators ('hardware modules') or the trim panel ('software modules'). Several suppliers began proposing these types of modules in the middle of the 1990s to the OEMs (e.g., Jost 1995, Buchholz 1998a, Mapleston 1999).

An earlier door module analysis from an OEMs perspective defined four architecturally different modules: (1) a simple plug-in-frame that consolidates window channels, (2) a rib structure that acts as carrier of window regulator and speaker, (3) a carrier plate made from steel or plastic that carries window regulator, wiring harness and speaker and seals the dry from the wet side, and (4) a module that combines trim and hardware components (Birkholz and Stark 1997).

More recently, a study on the European auto market found seven different types of door 'modules' in various stages from development to production (Sako and Warburton 1999). They vary in functionality and performance, but four represent different types of hardware modules (a carrier made of steel or plastic/composites with window regulator and wiring harness, some also include the window, some seal the dry side from the wet side of the door), the fifth is a trim

---

[121] This is true concerning at least the last 50 years.

('software') module with added hardware functionality, the sixth consists of window frame, window regulator and glass only, and the seventh is the conceptual idea of an entire door. The four hardware modules include cases 1 to 3 from Birkholz and Stark's analysis, the trim module equals number 4.

Finally, others have associated successive generations of door modules with the various types. The Korean automotive component manufacturer Samlip, for example, describes four different generations, and one additional intermediate generation (Samlip 2002). According to its description, a first generation door module is a simple steel stamping that enables to pre-assemble latch and lock onto it. Their 'second generation' module is represented by the design that exhibits a stamped steel panel as a carrier for window regulator, speaker and latch. Samlip's third generation of door modules is a design with a carrier made from non-sheet shaped composite (essentially Delphi's superplug design). A slight variation of this design they name generation 3.5. Finally, their fourth generation is a version that includes structural components, window frame, glass run channels, and the window regulator.

Table 7-1 summarizes and compares the different types of module definitions and descriptions. It also assigns a number to each type of module. Figure 7-2 to Figure 7-8 show examples for the individual types.

Type	Authors / Study source Module Description	Birkholz & Stark 1997	Sako & Warburton 1999	Samlip 2002
1	Module includes Latch / Lock combination on a simple stamped steel panel	N/A	N/A	"1st generation: latch/lock module"
2	Module includes window, glass run channels and window regulator	N/A	"niche module with integral window frame"	N/A
3	Module includes window regulator, speaker, and latch; carrier is made from stamped steel;	"plug-in frame"	"steel carrier"	"2nd generation: steel module"
4	Module includes window regulator, speaker, and latch; it also provides structural support to the door; carrier is made from stamped steel	"rib structure"	"steel carrier with door structure support"	"2nd generation: steel module"
5	Module includes window regulator, speaker, and latch; it also provides structural support to the door; carrier is made from plastic / composite	"carrier plate plastic/hybrid"	"plastic/ composite carrier (i.e., Delphi superplug)"	"3rd generation: Superplug"
6	Module includes window regulator and window; structural components are made from extrusions / stampings	N/A	"steel carrier with glass and integral door structure support"	"4th generation: cassette module"
7	Module includes window regulator, speaker, latch, and trim components; structural carrier is made of plastic/composites	"Integrated carrier/trim module"	"plastic interior trim based integrated module"	N/A
8	Module encompasses complete door	N/A	"complete door including exterior steel"	N/A

Table 7-1: Current door modules

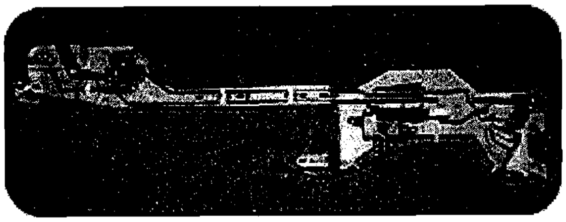


Figure 7-2: Type 1 door module[122]

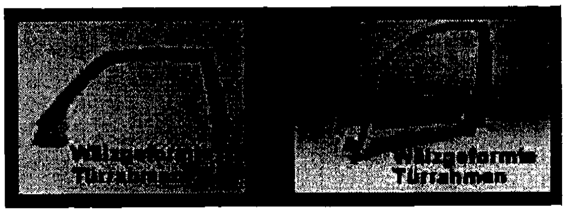


Figure 7-3: Type 2 door modules[123]

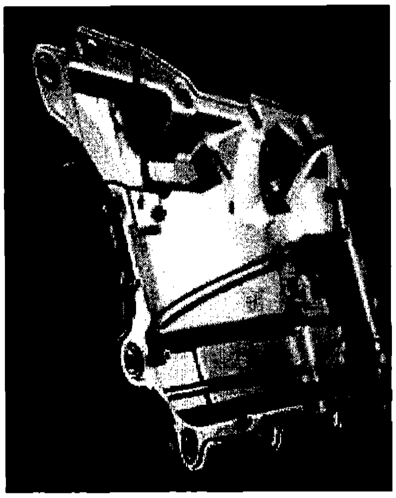


Figure 7-4: Type 3 door module[124]

---

[122] Picture source: http://samlip.co.kr/eng/products/chassis-1.html

[123] Source: http://www.duraauto.com/products/door.asp#Door%20Hardware%20Modules

[124] Source: http://www.borealisgroup.com/public/customer/automotive/interior/door_module_carrier/MainPage.jsp

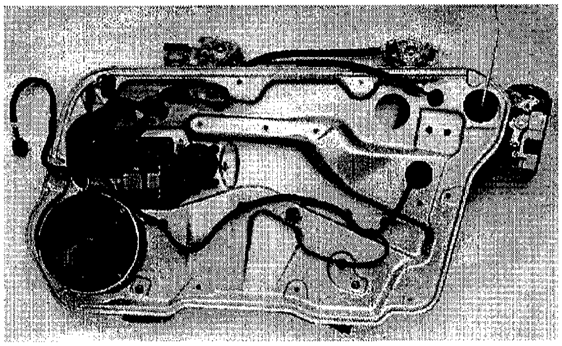


Figure 7-5: Type 4 door module[125]

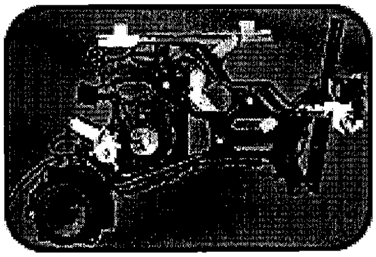


Figure 7-6: Type 5 door module[126]

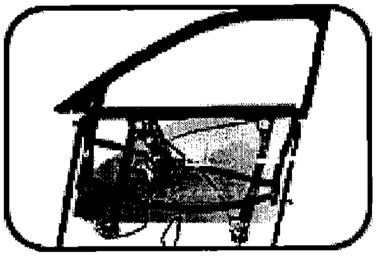


Figure 7-7: Type 6 door module[127]

---

[125] Source: http://www.arvinmeritor.com/products/car/doorsystemsphotos.asp

[126] Source: http://samlip.co.kr/eng/products/chassis-1.html

[127] Source: http://samlip.co.kr/eng/products/chassis-1.html

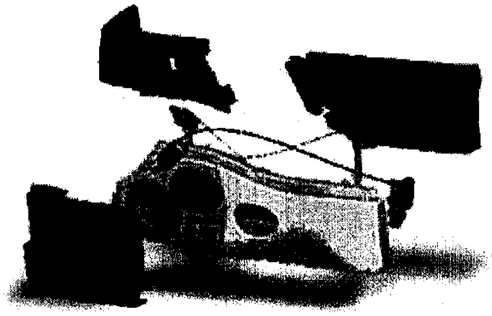


Figure 7-8: Type 7 door module[128]

While these cases have early on occupied the term 'module' in the automotive industry, they are very different concerning the functional content, used materials, employed manufacturing processes, etc. This variation in content and functionality makes it difficult to compare these 'modules' and to isolate the cost effects of modularity alone.

7.2.2 Today's mass produced automobile doors – door structures

A recent worldwide study of assembly plants conducted by IMVP collected data on product architectures of automobile door structures. The survey includes 65 vehicle assembly plants in Europe, North America, South Africa, Japan, Korea, and Taiwan. Participants were asked to identify material and manufacturing processes of the five major components of a driver-side front door of the vehicle model that represented the highest volume in the given plant.

The results demonstrate that there is amazingly little variation with respect to product architecture across almost all models. Figure 7-9 shows a total of 311 counts out of 325 (65 plants times 5 components) possible. The remaining difference represents missing data. Figure 7-9 illustrates that 95% of all counts fall in the material category 'steel.' Only one model had panels fabricated from composites and two other vehicles employ other materials for the anti-intrusion beam, one aluminum and the other a non-specified material. In other words, models produced in high production volumes are not very experimental with respect to materials.

---

[128] Source: http://www.visteon.com/news/features/121500.html

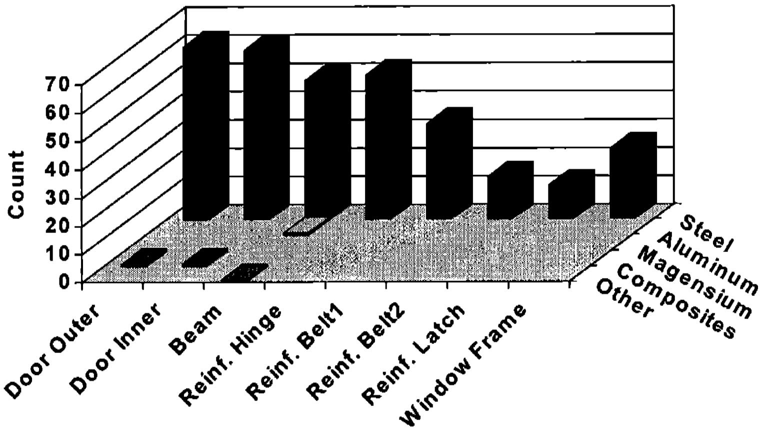


Figure 7-9: Materials used for door construction (total n_{count} = 311)

Looking at these 95% (i.e., steel components only) for components and manufacturing processes demonstrates that while there is some variation in the manufacturing processes employed (particularly for the anti-intrusion beam and the window frame)[129], almost all of these door designs use a fundamentally similar product architecture concept. This door architecture concept is the equivalent to the body-in-white concept for the car body as a whole. In the case of the door, two large steel panels (door inner and door outer) form a shell shaped structure, which is reinforced by a number of smaller reinforcements and strengthened by an intrusion beam. Figure 7-10 illustrates the similarity of the door designs. For example, more than 90% of the designs exhibits stamped steel outer and inner panels.[130]

---

[129] The window frame is the only component with which some models deviate from the conventional door architecture. While conventional doors provide a window frame essentially with the two large panels, some designs have panels that not exceed what is called the belt line, i.e., the lower bound of the window opening, and provide the frame as an individual component. This observation does not change the fundamental similarity of the architectures, however.

[130] The lower numbers for most of the reinforcements are the result of the attempt to keep the survey flexible and to allow to capture – if existing – new designs. The questionnaire asked to list the name, material, and manufacturing process for the five *major* components of the door of the model with the highest annual production in the plant under consideration. While there seems to be agreement that the two large panels belong to the five major parts, an agreement on the order of importance of the other components seems to be lacking. However, the fact that the components listed in Figure 7-10 lists components that are all common for this conventional design allows to conclude that the graphic is spread out not due design differences but due to differences in ordering of the components by the survey participants.

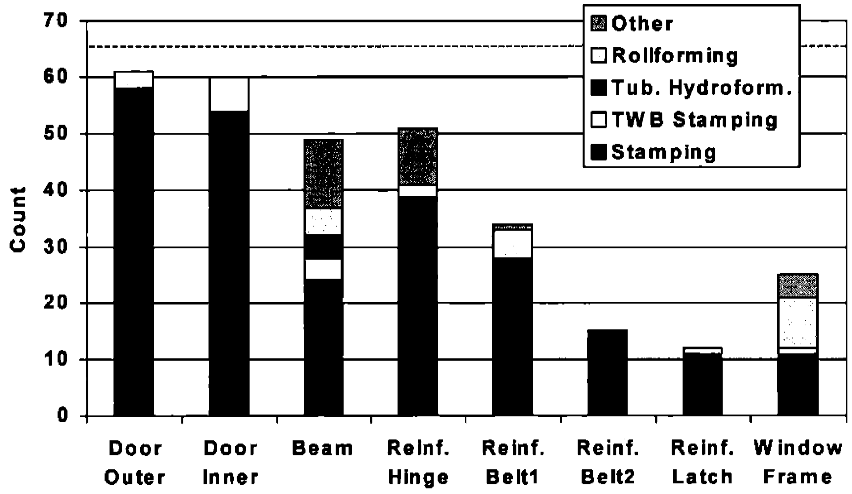


Figure 7-10: Manufacturing processes employed for major door components (total $n_{count} = 307$ - steel only)

7.2.3 Conclusions for experiment design (case selection)

On one hand, the modules in the door module studies make comparisons very difficult because of the functional differences across the modules. A comparative analysis requires similar boundaries, however. On the other hand, the vast majority of door structures (whose comparability can be controlled much better) that are in mass production today exhibit very little variation. Most of the designs employ very similar product architectures.

As a consequence, this analysis chooses several cases of door structures that have progressed into the development stage, but are not yet in production. While small adjustments were made to ensure comparability in terms of size, each of the cases is based on a design that has been built at least as a prototype. The individual designs are described and analyzed in the following section.

Due to the fact that there is no production data on these door designs, the costs have to be predicted for these products. Process-based cost models provide the tools to accomplish this task. The cost analysis follows in the section after the next.

7.3 Experiment 'Environment' (Control Factors)

To ensure comparability of the different cases requires careful thought about the boundary definition for the products. The boundary conditions with respect to the products are relatively easy to make comparable due to the somewhat contained structure of automobile doors. In contrast, the boundary setting for the products' functionality requires more careful analysis. Below, the two major dimensions of this boundary setting problem are discussed: (a) the comparability concerning functional content and (b) the comparability concerning performance or quality level. The discussion is presented separately for functions wholly provided and for functions only partially provided by the product. Finally, a function list is determined that is to be used for the experiments.

7.3.1 Wholly provided functions

In the ideal case for comparative purposes, the products under investigation would provide identical functionality to the customer. In terms of functional content, that means that the *number* and *type* of functions is the same across the different products. In terms of performance or quality, again ideally the functions would be provided on an identical level.

It is important to understand the extent to which these ideals are approached by the individual products. While *content* is relatively simple to identify, *performance* is more difficult to distinguish. Chapter 3 grouped a product's 'function' into 3 categories: technical functions, features, and product attributes. For the technical functions as well as attributes the comparability with respect to performance and quality are straightforward. In contrast, for features (e.g., aesthetic appearance) the performance or quality level is much more difficult to establish for comparative purposes. However, as also discussed in chapter 3, for many cases the performance or quality level surpasses (or has surpassed) a threshold value beyond which it looses its differentiating factor. In these cases, the features can be treated like simple technical functions that are either provided or they are not. In other words, if the function under consideration is viewed as a binary variable (it works or it does not), then there is only one (acceptable) performance level for this function. If, on the other hand, performance is perceived on different levels, the question becomes how large is the performance difference? What are the criteria used to measure this difference? Particularly difficult are performance dimensions that are somewhat subjective by their nature (e.g., sound when closing a door, extent to which paint of several painted surfaces match/do not match).

An additional quality dimension is the time over which the product will perform its function as expected. This question is closely related to the question of warranty issues. For instance, if an automobile door is slammed all carrying parts inside are subjected to inertia forces. That means that the carrier, independent of material or design, must be able to withstand these forces, ideally for the life of the product. Investigating this question with a simulated door-slam, one study found that a module carrier can be made from different materials (here steel and glass-filled thermoplastic) when designed properly without creating a part with inferior quality (Hoff et al. 1998).

Two attributes are discussed individually below. They are concerned with 'size' and 'weight' of the example products.

7.3.1.1 Size

For the purpose of this analysis, the example case studies control for product size. This is achieved by (a) selecting the same door for all cases, i.e., the front-door driver side door of a mid-size car as a base case and (b) by scaling the dimensions of all cases according to the base case. This idea assumes that the different cases could be mounted to the same vehicle, i.e., they are geometrically equivalent.

7.3.1.2 Weight

Lightweight as a performance dimension is generally delivered by the whole vehicle.[131] Weight seen as a global product performance parameter translates into power requirements for normalized vehicle performance such as top speed or acceleration. This in turn affects the fuel consumption of the vehicle. All else equal, lower weight can be considered advantageous. For this reason, more recently various OEM have begun introducing vehicles that use higher portions of lightweight materials as aluminum or magnesium (Crosse 1999).

For the door cases a similar approach is selected. *Weight* is assumed as a global product attribute and approximated with having a linear performance curve, i.e., lighter is better. It is

---

[131] Technically, the weight distribution throughout the vehicle (front/back, left/right, height) is another dimension that influences driving behavior. For the purpose of this analysis, I will neglect these higher order effects.

recognized that the different designs have different weights, but this performance dimension is kept separately. Differences in weight distribution within an individual door are neglected, only total weights are considered.

7.3.2 Partially provided functions

The question of comparability becomes even harder if the product delivers its functionality only partially. In order to establish fair conditions for comparability along the content and the performance dimensions requires some extra thought, because in this case the two dimensions are intertwined. If a product provides a function only partially, i.e., it provides the function only jointly together with other products, the question of what fraction the product contributes is interrelated to the quality of this contribution. Take, for example, the paint quality of an automobile door. While the paint has its own measurable level of quality, it also contributes to the overall appearance of the vehicle. The way in which it does so, however, is a function of both the door's paint and the paint of the rest of the body, because a major quality criterion for this function is the 'evenness' the paint achieves across all parts.

Two product functions that are only partially provided by the doors are discussed in detail below. They cover the functions *connection to body and sealing* and *safety.*

7.3.2.1 Connection to body and sealing

All doors in this analysis connect in almost identical ways to the rest of the car. All of them rest on two hinges in front and have their latch at the rear end of the door connecting with a striker to a striker plate when the door is closed. The striker plate is fixed to the body structure. All of the doors open in driving direction.[132] For these reasons, the following comparative analysis does not consider hinges or check links. Any combination of hinges and check links that provides the functionality 'to keep the door open below a certain closing force' could be installed in each of the different designs.

---

[132] This configuration has become somewhat of a standard for today's car doors. There are, however, alternatives. For example, in the early years of the automobile, many doors opened backward, i.e., they had the hinges in the back and the latch at the front. Other alternatives are doors that open upwards, e.g., at the Mercedes 300SL in the 1950s, or downwards, e.g., Joalto has designed and built a prototype of a door that slides under the car (Joalto Design 2000). In addition, the hinge mechanism could be used to distinguish doors that swing open from those that slide to open. All doors of this analysis are swing doors.

Also, all doors in this sample of experiments need to seal the gap between the door and the rest of the car against rainwater, wind, and noise. Although the performance level of this function depends mainly on the pressure applied by the closed door onto the seal, and in turn, on the accuracy of the geometric position and/or adjustability of hinges, latch and striker, all variants require a rubber seal, which is typically attached to the door cutout at the body. It is for this reason that this seal will not be considered in this analysis. It is assumed that all door designs enable the establishment of a watertight seal between door and body.

7.3.2.2 Safety

The second function a car door necessarily provides only partially is *safety*. There are typically two separate cases in which the doors of a car play an important role with regards to safety of the occupants of an automobile: frontal crash and side collision. The role a door plays in the first case differs from design to design of the whole body-in-white. In general, the more a door contributes to the stability of the passenger cage (without blocking access to the passengers after the crash), the better its safety performance.

However, if the assumption can be established that the safety performance of the individual designs surpasses at least a minimum threshold level, the cases could be compared. For the purpose of this analysis, it is assumed that the structures of all door concepts used as cases are contributing to the overall case safety at least on that minimum level, which is set by the base case. A similar assumption is made with respect to the second type of crash, i.e., side impact. All designs deliver at least a minimum level of safety, although in reality different materials, different hinge types and different levels of overlap with the body-in-white are likely to produce different results in crash tests.[133]

This assumption is based on the fact that all cases used for this analysis are at least in the prototyping stage and have fulfilled crash tests according to regulation FMVSS 214 (Federal Motor Vehicle Safety Standard 214). In the U.S., these standards are administered by the Office of Vehicle Safety Compliance of the National Highway Traffic Safety Administration. FMVSS

---

[133] In addition to the door structure, the components at the inside of the door towards the driver can have different impacts on the driver in case of a side collision. These differences are partially grounded in material specifications (e.g., Anonymous 1997a), partially a result of design differences (e.g Uduma 2000). Again, this study assumes that in all cases materials are being used for trim panels that deliver at least a minimum safety level. This assumption is particularly reasonable since the analysis focuses on the door structures.

214 – Side Impact Protection – "specifies performance requirements for protection of occupants in side impact crashes. The purpose of this standard is to reduce the risk of serious and fatal injury to occupants of passenger cars, multipurpose passenger vehicles, trucks, and buses." (NHTSA 2002) The standard prescribes static and crash test requirements. "Static Requirement: Vehicle doors must provide resistance to load applied via a rigid steel cylinder. Crash Test Requirements: Dummies in vehicle must meet requirements when stationary vehicle is impacted by moving deformable barrier at 54 km/h (33.5 mph), similar to intersection crash." In Europe, the ECE-R 95 test procedure prescribes similar test requirements.[134]

7.3.3 Function list for the experiments

Having discussed thoroughly the issues of performance for wholly and partially provided functions, the function list for the experimental case studies is chosen such that it does not violate the considerations mentioned above. In particular, the functions are considered equal in a sense that each product architecture delivers the function above a minimum threshold level. This makes the performances comparable.

The attribute *size* is controlled for by designing the experiments. Since this treatment does not work for the attribute *weight,* it is treated as an additional parameter outside of the architecture-cost analysis. Likewise, safety is considered as provided on or above a sufficient level and the sealing function is not considered.

Fundamentally a door is required to provide support of the body structure in its closed position for driving and frontal crash. Thus, *provide structure for driving and crash* is defined as the first function the products need to deliver. In addition, the safety function *provide side impact protection* is listed due to its importance in the automotive context (see previous section). Thirdly, automobiles are bought to a large extent due to the images they convey. This image is partially characterized by the outer shape of the vehicle and the doors are an important part of this appearance. Therefore, the third function that has been defined is to *provide aesthetically pleasing appearance*. Finally, the door structures serve also as the base for all other components the customer expects such as windows, loud speaker, and inner trim panel. Consequently, the

---

[134] The regulations differ somewhat in the measures they request to be taken. While the U.S. safety standard measures acceleration of rib and pelvis of a dummy caused by a side impact of the prescribed nature, the European guideline also requires measuring a viscous criterion (product of rib deformation velocity and rib deformation), rib deformation, pelvis load, and abdomen load (Anonymous 1998b).

fourth function of the function list is *provide structure to carry other parts.* Figure 7-14 shows the resulting function list for the product 'door structure.'

1. Provide structure for driving (and frontal crash)
2. Provide side impact protection
3. Provide aesthetically pleasing appearance
4. Provide structure to carry other parts
   (window, regulator, speaker, etc.)

Figure 7-11: Function list for automobile door structures

7.4 Experiments (Case Studies)

This section describes the experiments, i.e., the case studies. It provides descriptions of the four different designs including the manufacturing processes required for their production. Further, it presents comparative product architecture analyses using the framework developed in chapter 3. Finally, cost models are constructed and used to calculate production and logistic costs for the four experiments.

7.4.1 Case descriptions

The case descriptions provided two types of information. In addition to a description of the artifact itself, type and order of the manufacturing processes used to produce the doors are presented. The entire production process can be subdivided into four major processes. While individual designs may exhibit minor differences, the general path through production is exhibited in Figure 7-12.

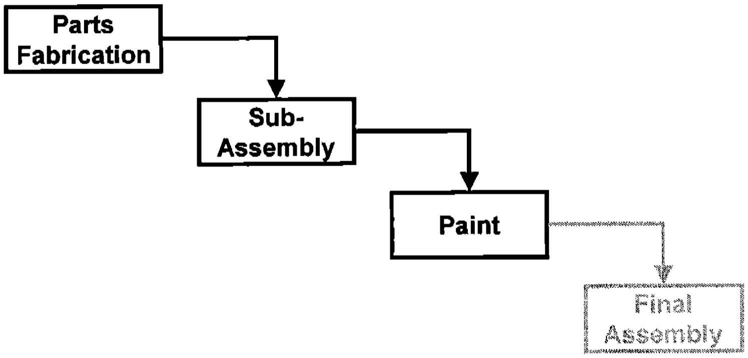


Figure 7-12: Generic manufacturing process steps

First, the major parts need to be manufactured. This process step will be called *parts fabrication*. Second, they need to be assembled to form the structure of the door. This step will be called *subassembly*. Third, the doors need to be painted, and fourth, the door is equipped with all those parts and components that would get damaged when painted. These steps are called *paint* and *final (trim) assembly*, respectively. While for all experiments (doors) the process steps *parts manufacturing*, *subassembly*, and *painting* are considered, the process step *final assembly* is included in the case descriptions to present a picture resembling the complete process.[135]

7.4.1.1 Case 1: The conventional door

The first door design represents the vast majority of car doors that are built today. Henceforth, it is referred to as the *Conventional Door*. It can be considered the base case for this study. The structure of the conventional door design consists of a shell shaped construction that is formed by two large stamped steel panels, the door inner panel and the door outer panel. In addition, several smaller steel stampings are used as reinforcements, in particular for the hinge area, the latch area, and the belt area. An anti-intrusion beam made from high strength steel provides side impact protection (Figure 7-13).

---

[135] The cost of the fourth step, final assembly, is predominantly driven by the costs for the purchased parts. Therefore, unless there are major design differences (with the exception of the inner trim panel there currently only few) the costs barely reflect product architecture differences. See also boundary setting in section 7.4.3.

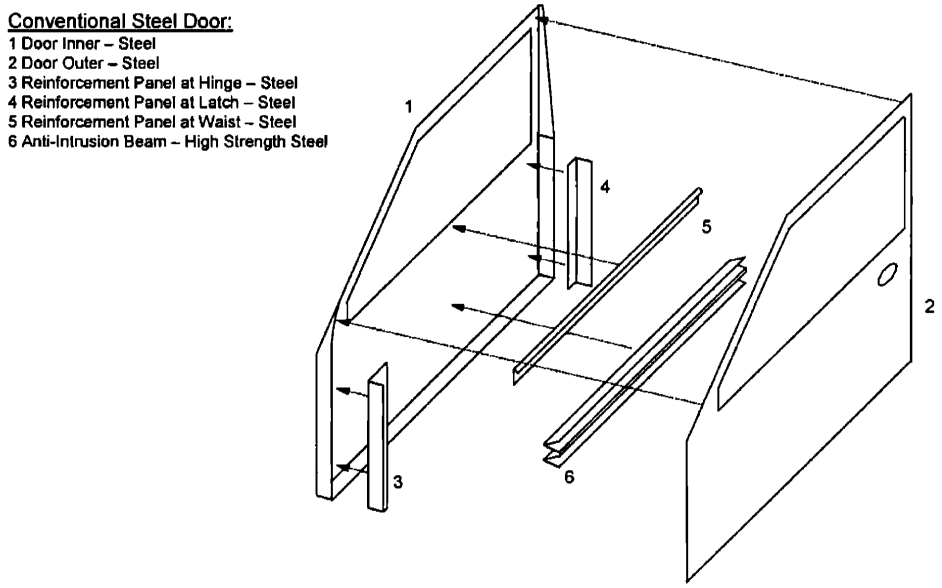


Figure 7-13: Schematic representation of the structure of the conventional door (case 1)

The exact production processes employed to manufacture this architecture are displayed in Figure 7-14 and described below.

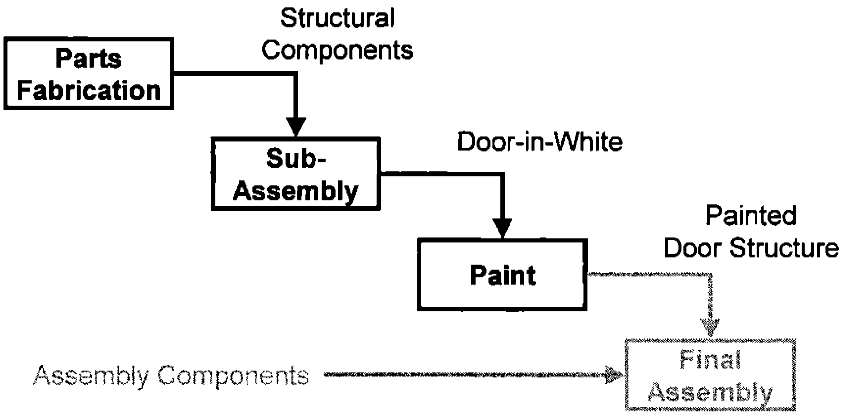


Figure 7-14: Production path for conventional door

With the exception of the anti-intrusion beam, all other parts are manufactured using blanking and stamping operations. First, from coils with steel several hundred feet long, blanks are cut-off in order to fit them into stamping presses. Equipped with dies resembling the part geometry, the stamping presses form the part in a combination of deep drawing and stretch-bending processes. The anti intrusion is formed using a high-temperature forming process (hot-rolling, hot-stamping).

During subassembly, the smaller stampings and the anti-intrusion beam are welded to the door inner panel before the inner and the outer panels are joined using a flange hemming process, in addition to some spot welds to fix inner and outer panel in their relative positions.

This welded door structure, the door-in-white, is then attached to the body-in-white, which next travels to the paint shop. The car body and closures are painted jointly. After painting, the doors are removed from the body at the beginning of the final assembly (trim) line.[136] The doors are mounted in pairs (of groups of four in case of four-door cars) on racks that typically hang on an overhead conveyor. This conveyor forms essentially the door trim line. While traveling through it, all remaining components get attached to the door, i.e., window regulator, glass, wiring harness, latch, locks, speaker, trim panel, mirror, and other, small components.

7.4.1.2 Case 2: The cruciform door

The door design for the second case study is characterized by its central load carrying part: Cross-shaped, it connects the hinges at the door front end with one (or potentially two) latch(es) at the rear end. For this reason this design is named *cruciform door.*[137]

---

[136] This process represents a so-called door-off final assembly strategy. 'Door-off' characterizes a separate door trim line, whereas with a 'door-on' strategy the final assembly of the door trim and hardware components is conducted on the main trim line with the doors mounted to the body-in-white.

[137] The cruciform design has been developed by Joalto, Inc. (Joalto Design 2000). Prototypes have been built to test manufacturing feasibility and safety performance (Townsend et al. 2001). Thus, functional equivalence with the base case can be assumed.

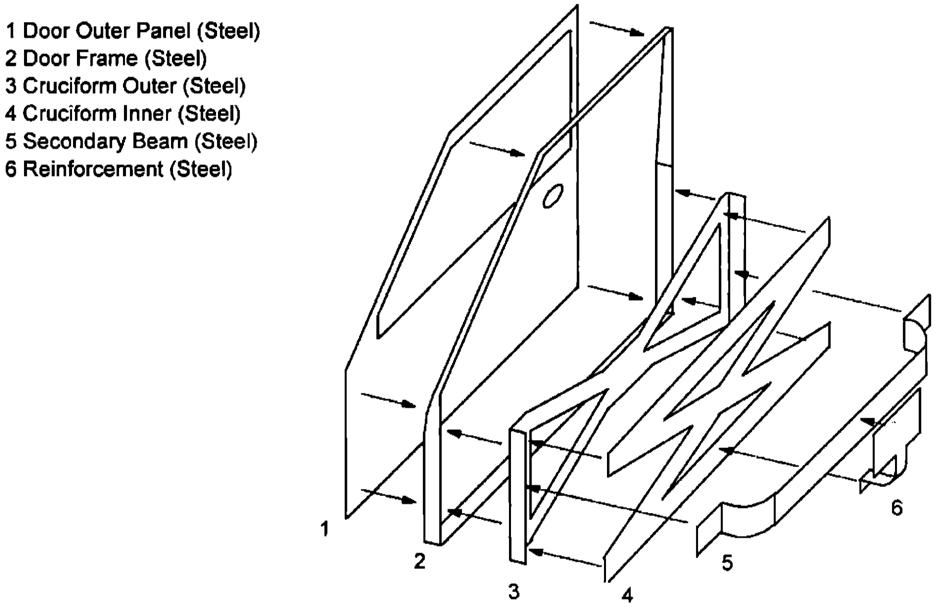


Figure 7-15: Schematic representation of the structure of the cruciform door (case 2)

Although this design uses the same material as the base case, mostly steel, its architecture differs significantly. The process description follows the standard four steps *parts fabrication, subassembly, paint* and *final (trim) assembly* (Figure 7-16).

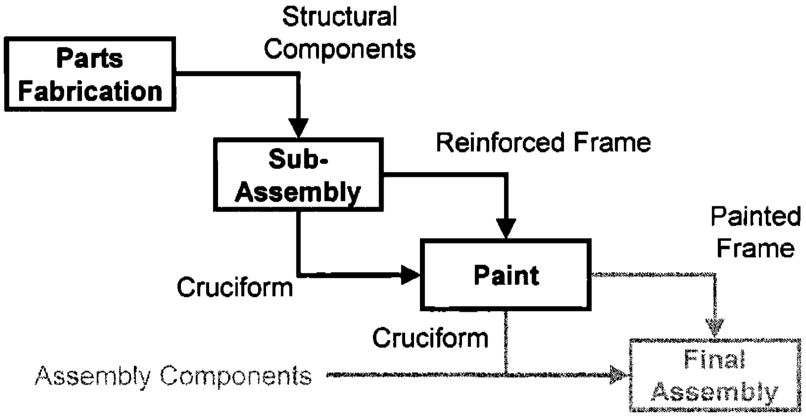


Figure 7-16: Production path for cruciform door

All structural parts are manufactured using blanking and stamping operations. Cruciform outer (3), cruciform inner (4), the secondary beam (5) and a reinforcement (6) are welded together to form the load carrying unit, the cruciform. The outer panel (1) is identical to the outer panel of the base case. Finally, the door frame (2) is assumed to be a stamping as well. It serves to carry the outer panel (1) and to connect the outer panel to the cruciform (Figure 7-15).

The subassembly process results in two structural parts, the cruciform made from four stampings and the outer panel reinforced by the frame. The reinforced outer panel is then equipped with hinges and subsequently attached to the car body to get painted together with the body. The cruciform can be painted separately. It does not have to match the paint quality (or even the color) since it will not be visible once it is installed.

After painting the reinforced outer panels are removed from the body and enter the door trim line. Although similar to the base case, this trim line is shorter because major components (window regulator, glass, wiring harness) are pre-assembled onto the cruciform. Next, this subassembly is attached to the door. Finally, the door assembly process is completed by attaching smaller components and the trim panel to the door.

7.4.1.3 Case 3: The cast frame door

The third case employs a door with an architecture that is very different from the conventional design. Central element of this door architecture is a frame that is cast in one piece and made from magnesium (Figure 7-17). For this reason this design will be referred to as the *Cast Frame Door*. The cast frame carries the load of the door as well as all other components. The frame is ring-shaped and covers the circumference of the whole door. An aluminum outer panel covers the outside of the frame and an anti-intrusion beam made of composites provides side impact protection. Designs similar to this one are in the research and development stage or have been prototyped at a number of OEMs and first tier suppliers.[138] In size and functionality, this design matches the base case, i.e., the conventional door.

---

[138] Prototype studies using different designs have demonstrated the feasibility of structural door components made from cast magnesium. Designs range from replacing the structural door inner panel of a conventional door with magnesium casting (Anonymous 1993b) to manufacturing a door frame with integrated anti-intrusion beam from magnesium (Tikal and Vollmer 1997). More recently, Mercedes-Benz has introduced a door consisting of a magnesium die-cast frame and aluminum outer skin (The Economist Intelligence Unit 2000c). Likewise, anti-intrusion beams made out of fiber-reinforced plastics (FRP) have been successfully tested for their ability to withstand side impacts (Anonymous 1998b).

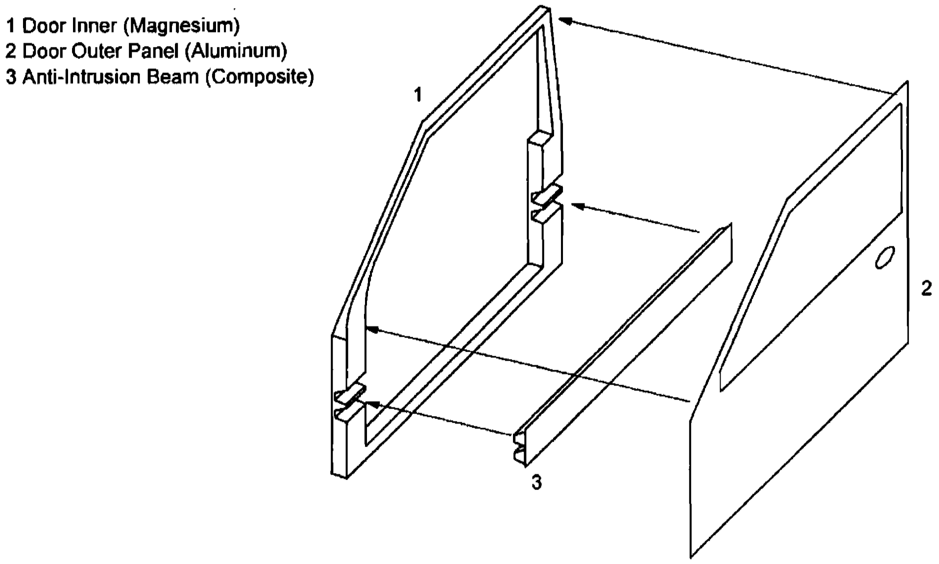


Figure 7-17: Schematic representation of the structure of the cast frame door (case 3)

The differences in design are accompanied by differences in production processes, relative to the base case. However, while major differences are found in the areas of part fabrication, subassembly and final assembly (painting is considered identical to the base case architecture), the production path is almost identical (Figure 7-18).

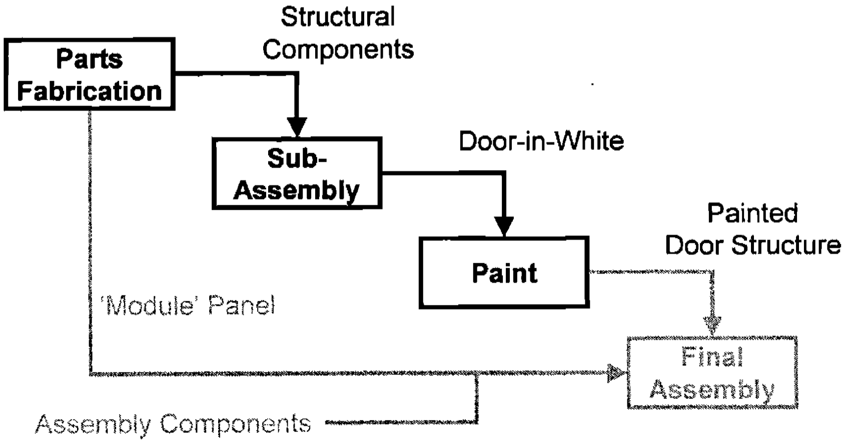


Figure 7-18: Production path for cast frame door

The parts fabrication step includes the manufacturing of the three major parts. A cold-chamber die casting process is used to manufacture the cast frame. Its surface will be subsequently powder coated to protect it against corrosion. The second component, the outer panel, will be manufactured with the same processes as the door outer panel of the base case design, except that aluminum is considered as the material. Although the process steps are identical (i.e., blanking and stamping) there are differences in the process conditions caused by the material change. Mainly, the stamping presses run slower to accommodate the larger spring back that accompanies aluminum relative to steel. The third structural component, the anti-intrusion beam, is made of a glass fiber polyester mix and manufactured using a pulltrusion process.

During structure subassembly, the anti-intrusion beam is attached to the door inner frame. Bolts are used to join the two components. The second assembly operation is the process of joining door outer and cast frame. Similarly to the conventional door a flange hemming process will be used, supported by adhesive bonding.

The completed door-in-white is attached to the body-in-white and travels with it through the paint shop identical to the case of the conventional door.

While the paint process is assumed to be identical with the conventional designs, the assembly process for the cast frame door again differs from the one employed for the conventional design. An additional part, the 'module' panel, is manufactured. This compression-molded component is made from a 30% glass/polyester mix and serves as a carrier for the window regulator, the wiring harness and the latch subassembly. These components are mounted to the panel in the separate module assembly process. The completed 'modules' are then brought to the final assembly line, where they are attached to the doors in a shortened door trim line. The door line is reduced in length by the amount of work that has been transferred to the module assembly. After the 'module' has been attached to the door, the door is completed with the remaining parts.

7.4.1.4 Case 4: The extrusion frame door

Like the previous design, the fourth door architecture is also characterized by its main structural part: a frame welded together using aluminum extrusions. For this reason this design is henceforth referred to as *Extrusion Frame Door.*[139] Based on real case data, the design has been modified with respect to the window frame and scaled in size to be comparable with the base case (Figure 7-19).

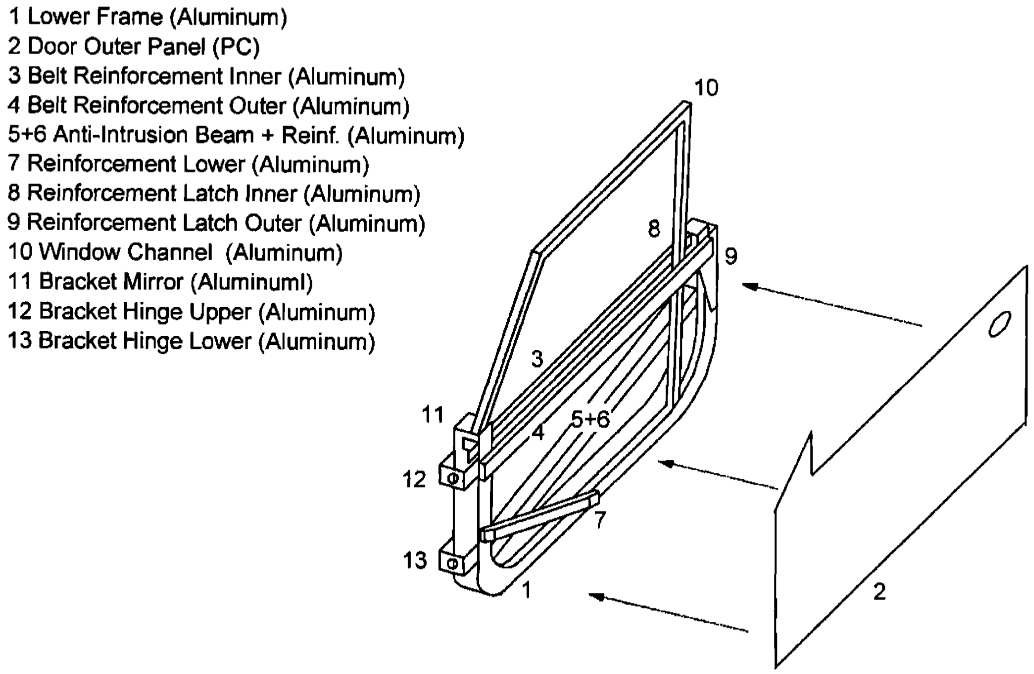


Figure 7-19: Schematic representation of the structure of the extrusion frame door

Although this design makes to some extent the reordering of some processes possible, the four process manufacturing steps parts fabrication, subassembly, paint and final assembly are followed in conventional order here for comparative descriptive purposes (Figure 7-20).

---

[139] The extrusion frame door resembles to some extent a design type for the vehicle body called space frame design.

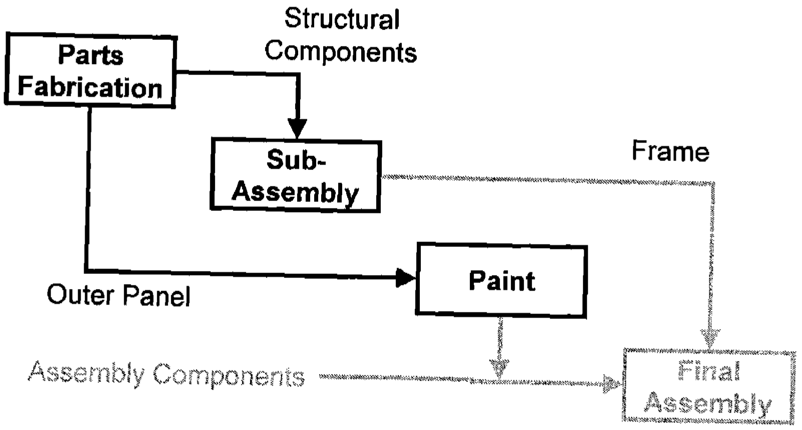


Figure 7-20: Production path of the extrusion frame door

The structural component of the extrusion frame door is a frame that is formed by several aluminum extrusions. A U-shaped extrusion forms the lower frame (1). Two straight extrusions serve as belt reinforcements inner (3) and outer (4). A diagonal reinforcement solves for statically determinacy and provides the function of an anti-intrusion beam (5). Additional reinforcements increase the stability of the latch area (8+9). Extra brackets provide mounting surfaces for the mirror (11) and the hinges (12+13).

Most of the frame's parts are manufactured as extrusions; some reinforcements are stamped (e.g., the reinforcements in the latch area). Several of the extrusions require subsequent bending. Regarding their geometry, the stampings are product specific (in particular the bracket for the hinges).

The subassembly process step includes the assembly of all but two extrusions using welding processes (see Figure 7-21). The two pieces of the window channel (10 a and 10b) are assumed to be joined by welding in a separate operation. The window channel is later mechanically fastened to the rest of the frame during subassembly.

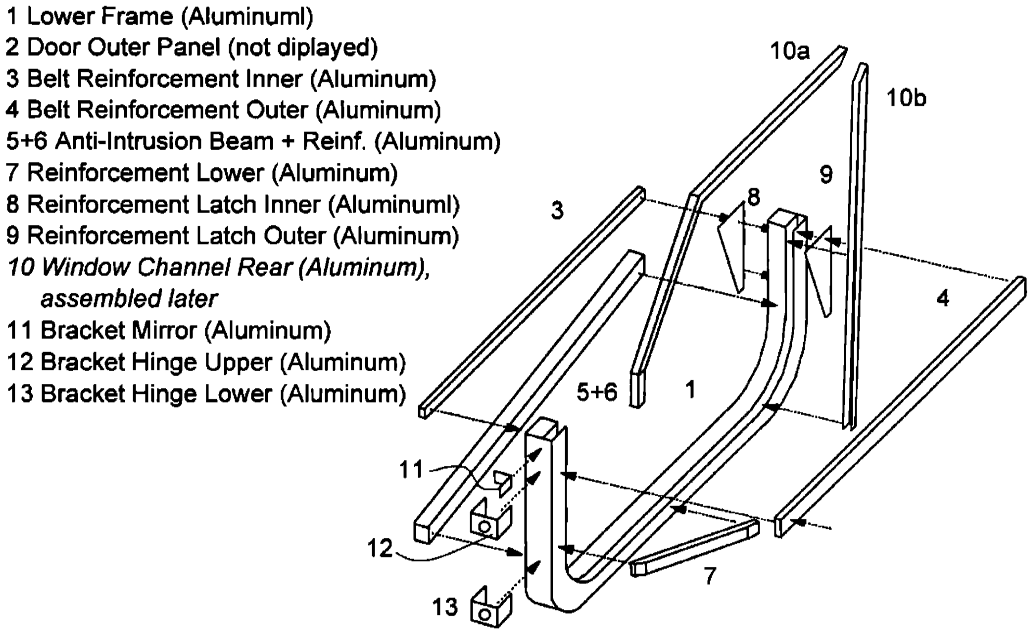


Figure 7-21: Assembly of the extrusion frame

The outer panel is assumed to be manufactured from a thermoplastic material and injection molded (compare Pothoven 1999).[140] With this door design, the door outer panel is the only door component that needs to be painted.[141] After painting it is delivered directly to the final assembly line. The window regulator is attached with three rivets to the unpainted aluminum frame before they are jointly shipped to the final assembly line. Starting with the frame, the doors are built up around the frames (only the inner panel is equipped with wiring harness, switch and small parts in a separate pre-assembly process). The final assembly process begins with the attachment of the window frame. Next, the door frames are mounted on overhead conveyor racks and travel through the final assembly line where they are completed with wiring harness, inner panel, glass, mirror, other small components, and, in a last step, the outer panel.

---

[140] Alternatively, similar outer panels have also been manufactured from thermoset materials (Anonymous 2001b).

[141] Technically, the outside of the window frame is also visible. However, since the fame is attached with mechanical fasteners, a potential painting of the frame would not constrain the process sequence.

7.4.2 Product Architecture Analysis

The following analyses of product architecture differences employ the framework developed in chapter 3. To limit the complexity of the experiments, only the door structures are considered and, consequently, the extent of the supply chain is limited to encompassing only *parts fabrication, assembly,* and *paint.*

For each case a function-component matrix is constructed to allocate the functions determined in section 7.3.3. Appendix B (chapter 0) lists the function-component matrices for all cases (Figure 10-1 to Figure 10-4). The matrices enable to calculate for each function individually the two indices *component count* and *total function.* These two indices are then employed to map each case's function-component allocation scheme in function-component allocation maps. These maps measure how a given product architecture deviates from the 'ideal' of perfect modularity. They show each function's position individually in the plane of possible allocation schemes.

The second part of the product architecture description, the interface assessment, includes the three interface dimensions *nature, reversibility,* and *standardization.* Interface matrices are used to assess the first two dimensions and standardization maps to evaluate the third. Appendix C: Interface Assessments presents the data in all detail (Figure 11-1 to Figure 11-8). Finally, all data are aggregated for visual representation in the product architecture maps.

7.4.2.1 Case 1: The conventional door

Mapping all functions with help of their two indices reveals that different functions are located in different regions of the map (Figure 7-22). The function *structure* is found in the integral-complex region. The reason for this is that almost all components of this design contribute to this function. In contrast, the function *aesthetic appearance* sits in the modular-like region due to the circumstances that it is solely provided by one component, the door outer panel. It sits not at the corner of the map, i.e., the spot of perfectly modular functions, because the outer panel also contributes significantly to the function *structure.* The two remaining functions, *side impact protection* and *aesthetic appearance,* are located in the integral-consolidated region because they are predominantly provided by a few (one or two) components that simultaneously also provide other functions.

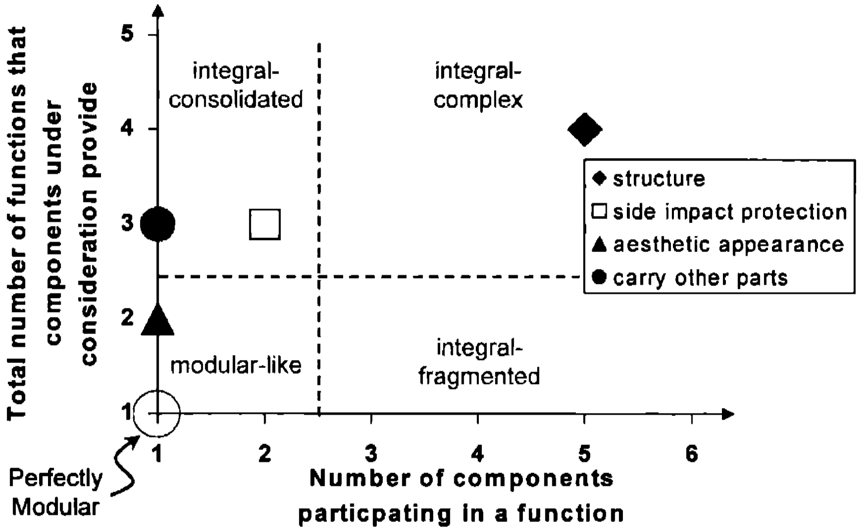


Figure 7-22: Function-component allocation scheme for conventional door

Analyzing the interfaces of the conventional door demonstrates the central role of the door inner panel. It is the only component that connects with every other component. In fact, every component connects only with the door inner panel. The nature of the interfaces is exclusively spatial (this is true for all cases of this set of experiments, since only door structures are considered here). The intensity level is high for almost all interfaces because most components collectively form the structure of the door. In turn, the structure requires not only precise geometric position, but also transmission of mechanical forces. The reversibility level is very low for all interfaces. All interfaces require high efforts to disconnect them and some also exhibit relative large depths. The high effort necessary to disconnect them is caused by the connection process: welding. The reason for the relative depth of most interfaces lies in the central role of the door inner panel. During assembly, all components are consecutively assembled to the inner panel, one on top of the other. At the last assembly step, the inner and the outer panel are connected with a ham flanging process to form a shell-shaped structure, with all other components on the *inside* of this shell. Consequently, the interfaces formed first got 'buried' deepest. Finally, an assessment of the standardization level of the components, measured per function, uncovers that due to their geometric specificity most components exhibit a very low level of standardization (lower left corner in Figure 11-2). The likelihood is very low that each of them can be used in another product of its class or family, or that it can use components from other members of the product family.

Figure 7-23 presents the product architecture map for the conventional door. It adds a graphic illustration of the interface assessments (z-axis), aggregated for each function but separate for each interface characteristic, to the function-component allocation scheme (x-y plane).

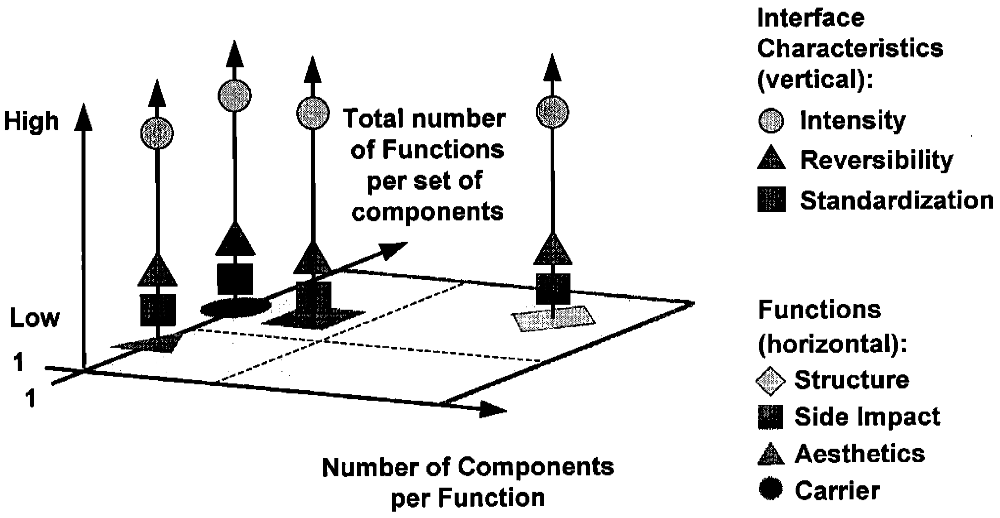


Figure 7-23: Product architecture map for conventional door

7.4.2.2 Case 2: The cruciform door

Looking at the first view, the cruciform door overall does not appears too different from the conventional door. The cruciform door, however, does exhibit some noteworthy differences (Figure 7-24). With respect to the function-component allocation scheme two major observations need to be pointed out. First, the function *structure,* albeit still in the integral-complex region, has moved much closer to the integral-consolidated region of the allocation map. This feature increases its chances for reusability across a product family. Second, the function *aesthetic appearance* is located at precisely the same spot as it is in case of the conventional door. The reason for this is that the function providing element, the door outer panel, is almost identical for both architectures. As it will be shown below, however, the interface characteristic of this function is very different for the cruciform architecture compared to the conventional architecture. This is a good example why it is advantageous to assess function-component allocation schemes and interface characteristics separately.

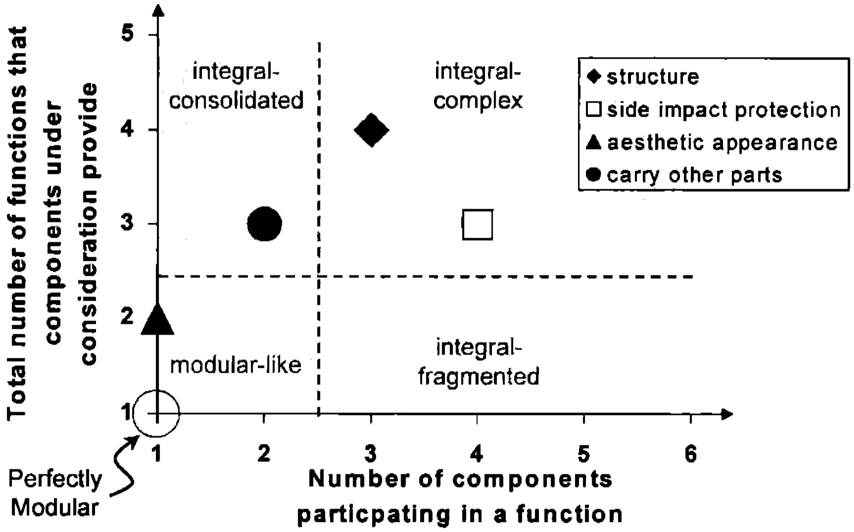


Figure 7-24: Function-component allocation scheme for cruciform door

The interface analysis reveals no significant differences with respect to the nature of the interfaces. Again, exclusively spatial nature, and high intensity due to mechanical force transmission, are the reasons for this assessment. In contrast, some interfaces show major dissimilarities to the ones of the conventional door with respect to their reversibility. This has two reasons. First, the main interface between the two groups of components (door outer panel plus frame on one hand, and the four components forming the cruciform on the other) is mechanically fastened, i.e., it clearly requires a lower level of effort to disconnect than a welded bond. Second, the structure of this architecture enables the components to be pre-assembled into subassemblies, i.e., assembly is not necessarily a string of operations as it is the case with the conventional door. As a consequence, most interfaces are assessed shallow or medium with respect to the interface depth.

Finally, while overall somewhat similar, the level of standardization is assessed slightly higher for the structure of the cruciform door, compared to the conventional door. Reason behind this assessment is that the cruciform structure connects to the other major components (outer panel plus frame) only at four points, and not over extended surfaces, as in the case of the conventional door. This increases somewhat the chances that the cruciform could be used in next-generation or sister products. Figure 7-25 summarizes the architecture assessment for the cruciform door.

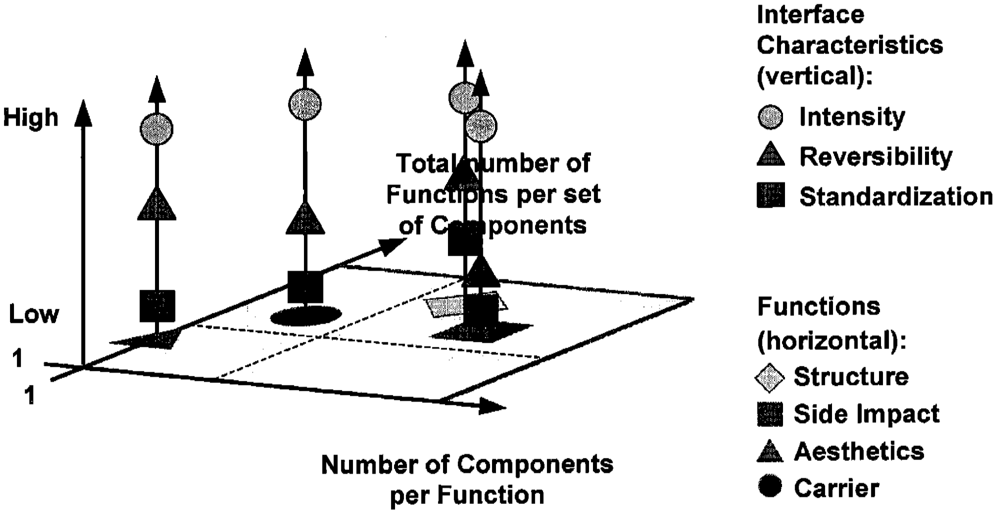


Figure 7-25: Product architecture map for cruciform door

7.4.2.3 Case 3: The cast frame door

The observation that all functions of the cast frame door are located further to the left in the map compared to the base case, the conventional door, is an effect caused by this design having overall fewer components. This architecture is – on average – more consolidated (Figure 7-26). But in addition to this generic information, the function-component allocation scheme also reveals another piece of information. The function *aesthetic appearance* is located in the lower left corner of the map, i.e., in the location of a perfectly modular one-to-one notion between function and component. This effect is caused by the rigid structure of the cast frame that allows this component to solely provide the structure for the door, to carry additional parts, and to contribute to side impact protection. However, functional isolation is for some business strategies necessary but not necessarily sufficient, and the interface analysis needs to fill these gaps – another example for the analysis superiority when the dimensions *elements* and *components* are assessed independently.

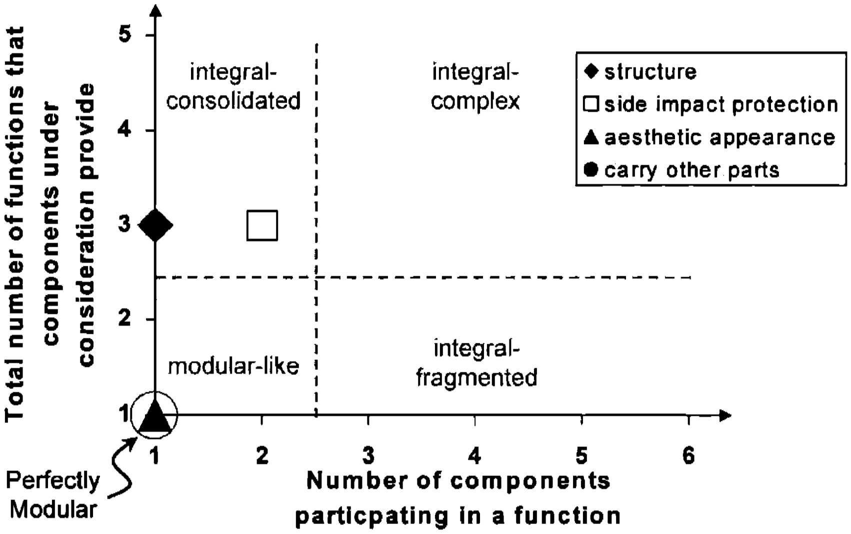


Figure 7-26: Function-component allocation scheme for cast frame door

Like the previous two architectures, the cast frame door also displays interfaces that are exclusively spatial in nature and rather high in intensity due to the requirements to transmit mechanical forces.

Although three components necessarily have to be assembled as a string, for this design the technology used to execute the last assembly operation has a strong impact on the architecture's overall reversibility. During assembly, first the anti-intrusion beam is bolted to the cast frame. In a second step, the door outer panel is attached to the cast frame using a flange hamming process. This process places a high effort requirement on any disconnection attempt.

Finally, the high level of parts integration that the three components of this architecture exhibit, makes re-using any of them in products that deviate more than minimal relatively unlikely. Accordingly, the standardization level is assessed as minimal for all function carrying components. Figure 7-27 illustrates the complete product architecture analysis graphically.

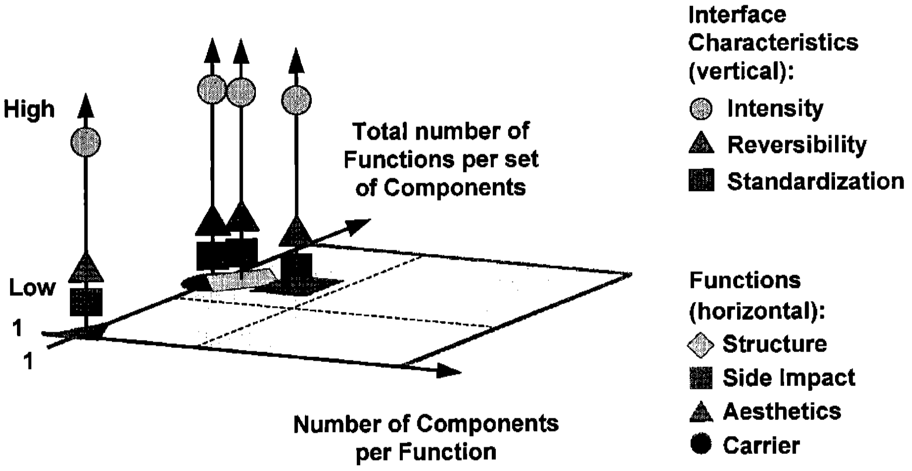


Figure 7-27: Product architecture map for cast frame door

7.4.2.4 Case 4: The extrusion frame door

Like the cast frame architecture, the extrusion frame architecture also exhibits a small number of components (on the hierarchy level under consideration) – compared to the base case (Figure 7-28). Thus, the functions of this architecture are also – on average – located closer to the left of the function-component map, i.e., closer to consolidated regions. There is, however, a significant difference between this architecture and the one previously analyzed. The extrusion frame architecture shows perfect function separation of one function from the rest – and the remaining functions are consolidated in one component. While the function *aesthetic appearance* is completely – and exclusively – provided by the components 'door outer panel' and 'window frame,' the functions *structure, side impact protection,* and *carry other parts* are completely – and exclusively – provided by the 'frame.'

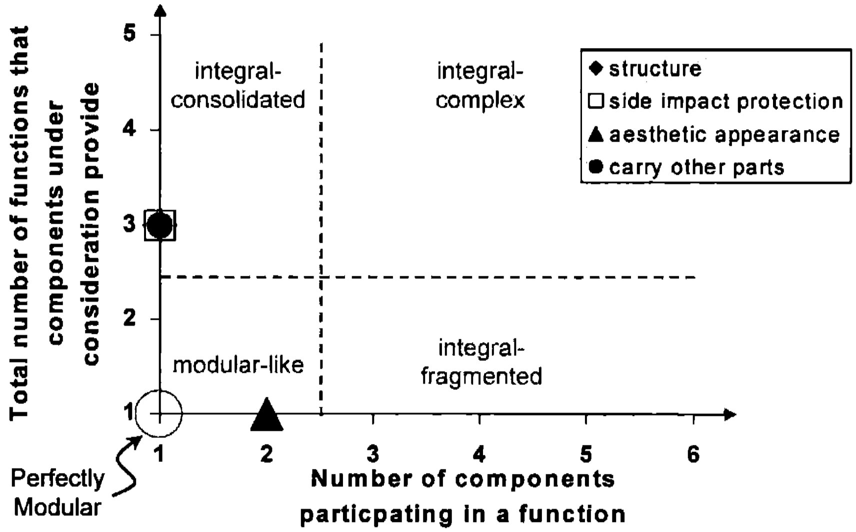


Figure 7-28: Function-component allocation scheme for extrusion frame door

Although similar on the first view, the extrusion frame architecture differs also significantly from the cast frame architecture with respect to interface characteristics. First, while generally similar in nature to all other architectures, spatial in category and high in intensity, this architecture exhibits one interface that is somewhat different. The sole purpose of connection between frame and outer panel is to hold the outer panel in place. All this connection needs to carry is the weight of the outer panel. As a result, its intensity is assessed lower compared to most others in this set of cases. Second, precisely this interface represents also the lowest effort to disconnect it – it is a simple snap-fit connection. And third, this easy-to-disconnect interface is also the last one formed in the assembly process, and thus reduces on the disassembly path the depth of the other interfaces of the architecture.

The technical details of this interface place its participants, in particular the outer panel, also somewhat higher on the standardization map. Due to the difference in thermal expansion coefficients between the frame material and the outer panel material, the connection between these components consists of only three points. Consequently, it is conceivable that a slightly different panel (e.g., next generation) is used with the same frame. Figure 7-29 summarizes the product architecture analysis for the extrusion frame door.

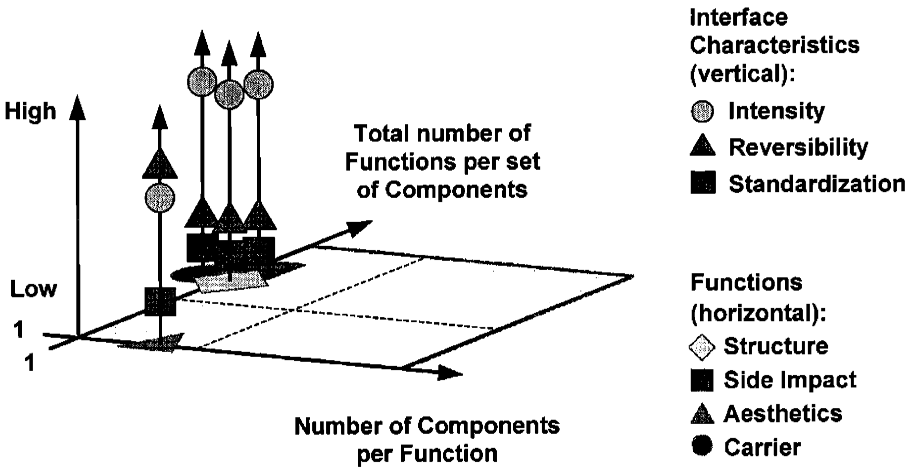


Figure 7-29: Product architecture map for extrusion frame door

7.4.3 Cost Analysis (Baseline scenario)

In chapter 4, the importance of how the analysis is framed was emphasized. Since the boundary setting has a strong impact on the results, it is paramount to be explicit and unambiguous where the boundaries are drawn for the analysis. For this reason, the boundaries for the case analyses presented here are specified with respect to product life cycle and supply chain phases, external environment, direct/indirect costs, and modeling tools before the individual cases are presented.

As mentioned earlier, cost occurrence curves look different for products that differ along characteristics such as product value, production volume, and product lifetime. The product of these experimental case studies, an automobile door, is a product of medium value, relatively large production volume, and medium lifetime. It represents a product category whose members are predominantly sold after production and who compete on selling price.[142] Also, for most products of higher production volumes, production costs dominate development costs. Therefore, this cost analysis focuses on the life cycle phase *production* (Figure 7-30).

---

[142] The doors themselves, obviously, do not compete in their own market, but it is conceivable that if they did, the market would look similar to the one for whole vehicles.

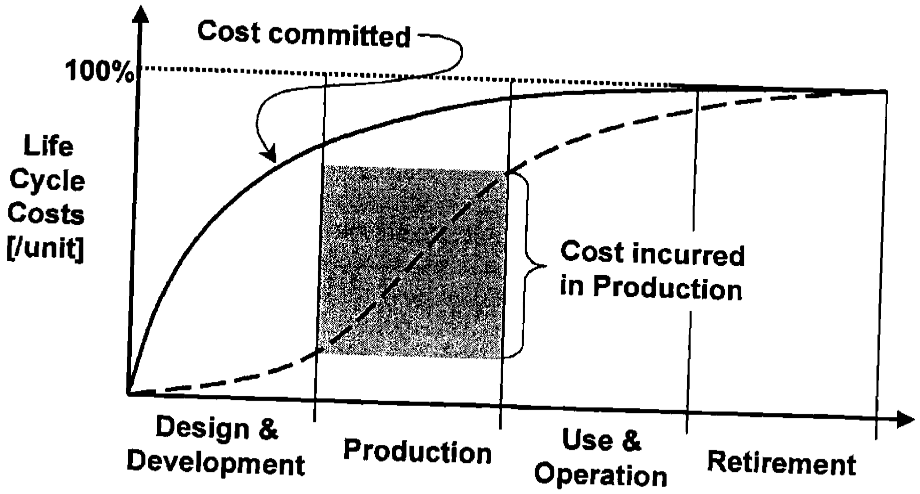


Figure 7-30: Focus of this analysis: the production costs

The choice of the product life cycle phase also determines somewhat the choice of the supply chain. Together with the product choice, i.e., painted door structures, the chosen life-cycle phase determines the end of the supply chain: when the doors structures are completed in shape and color. The beginning of the supply chain is placed at a point before which there is little evidence that architectural choices have any impact on costs: materials. Therefore, the beginning of the supply chain model is placed on parts fabrication. With respect to its granularity, the supply chain consists of three steps, parts fabrication, assembly, and paint[143]. The production steps are connected with stages for transportation and/or storage. Figure 7-31 depicts the supply chain used for this study.

---

[143] Of the total cost of the paint process, only the variable costs are considered in the analysis below. This is the result of the difficulty of assigning fractions of capital equipment of today's large industrial painting facilities to doors as subsystems of the body.

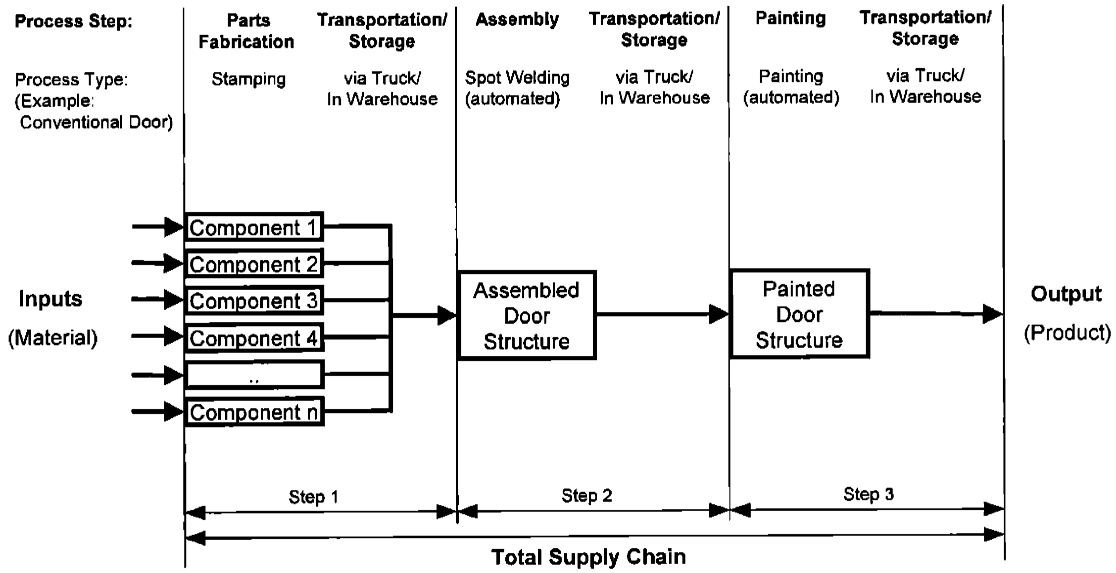


Figure 7-31: Supply chain definition

The external environment of this production scenario consists of two major topics. The first one is concerned with external factors such as location, wage and interest rates, or labor regulations. For this analysis, it is assumed that no facility exists, that is, a so-called green field approach is chosen. In order to isolate the cost effects caused by architectural differences, a dilution of the cost effects through constraints via existing facilities should be avoided. Further, it is assumed that the supply chain is located in a developed country. Storage, transportation and labor costs typical for this environment are applied. The second topic of the baseline scenario is the product strategy. For the baseline scenario, a single product program and a product program lifetime of 5 years are assumed. The production volume is 100, 000 units per year. The exogenous factors for the baseline scenario that are applied to the analysis are listed in Table 7-2.

While the equipment and tooling selection follows a green-field approach, the activity-based costing philosophy considers most machinery as non-dedicated, i.e., the product is charged only for the fraction of time it uses the resource. Some machinery, e.g., in assembly, however, is considered dedicated, i.e., product-specific.

Variable Description	Baseline Scenario
Annual Production Volume	100,000 units/year
Production Program Life	5 years
Assembly and Machine Operator Wage	30 \$/hr
Supervisor	40 \$/hr
Factory Days per Year	230
Shifts	3 /day
Distance between process steps	400 km
Average storage time per step	10 days
Building Costs	120 \$/m^2
Equipment Life	10 years
Building Life	25 years
Interest Rate	10%
Fixed Overhead Rate	35%

Table 7-2: Baseline scenario

With respect to the direct/indirect cost question, this analysis remains close to the production process, i.e., it focuses on costs closely associated with the production process. Aiming at comparing cost effects of alternative architectures, the analysis employs a bottom-up costing approach. Consequently, secondary and tertiary cost effects would cause the number of scenarios to explode. Therefore, to maintain comparability across the cases and to make cost effects visible that are caused by architectural differences, this analysis incorporates costs of manufacturing components, assembling and painting them, and storage and transportation processes in between. Costs included in the analysis are variable costs for labor, materials, and energy as well as fixed costs for machinery, tooling, facilities, maintenance, and an overhead portion that reflects the requirement for experienced personnel to run sophisticated machinery.

Finally, process-based cost modeling tools are applied to model the various process steps. To model costs of parts fabrication processes, such as stamping, casting, or extrusion, different technical cost models are used. Technical cost models incorporate first principle engineering knowledge of how product and process choice affect process requirements, speed, and yield, and therefore, costs. (For a more detailed description of technical cost models see chapter 5.) The logistics costs are represented in straightforward spreadsheet-based cost models. For modeling of cost effects on inventory, a model developed by Baker et al. (Baker et al. 1986) is adopted and modified to the case specifics. All cost modeling has been conducted with neutrality towards ownership, that is, all non-product related factors are modeled as exogenous economic factors, and second order effects like bargaining or monopoly power are not considered.

The results of the cost modeling efforts for the baseline scenario are depicted in Figure 7-32 through Figure 7-39 and discussed below.

7.4.3.1 Case 1: The conventional door

The unit costs of the conventional door across all three supply chain steps are dominated by processing costs followed by logistics cost. The work-in-process (WIP) costs are relatively small for a product of this value (~ $100) and this supply chain configuration (total 30 days). Of the three supply chain steps, the step *parts fabrication* consumes the largest share of total cost (Figure 7-32). The reason for this effect is the dedicated tooling: expensive stamping dies that can only be used to manufacture the parts specific for the door under consideration. The process costs of the assembly step represent primarily machine cost for the assembly line, which is assumed to be dedicated.

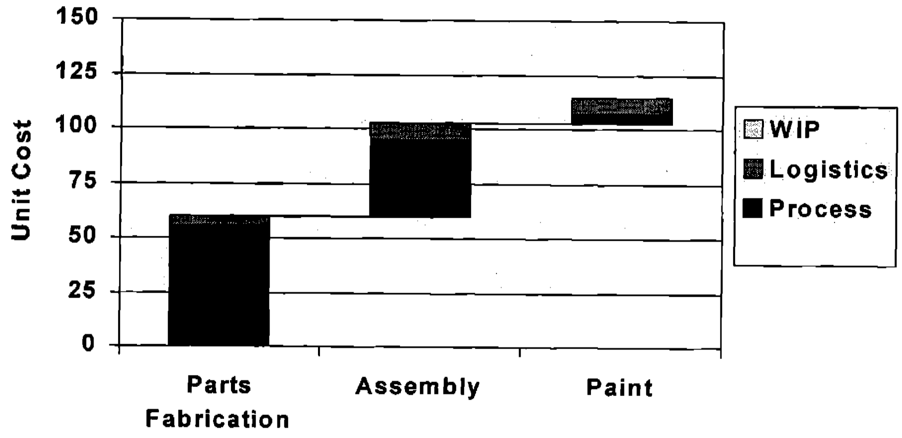


Figure 7-32: Unit cost for conventional door (baseline scenario)

As a consequence, total fixed costs (combined for all three steps) when focused only on process costs are almost evenly distributed between tooling and machine costs (Figure 7-33). Overhead on these two factors complements the fixed cost block. The variable costs are dominated by the costs for material. Labors costs are comparatively small reflecting the high degree of automation. Finally, energy costs are negligible.

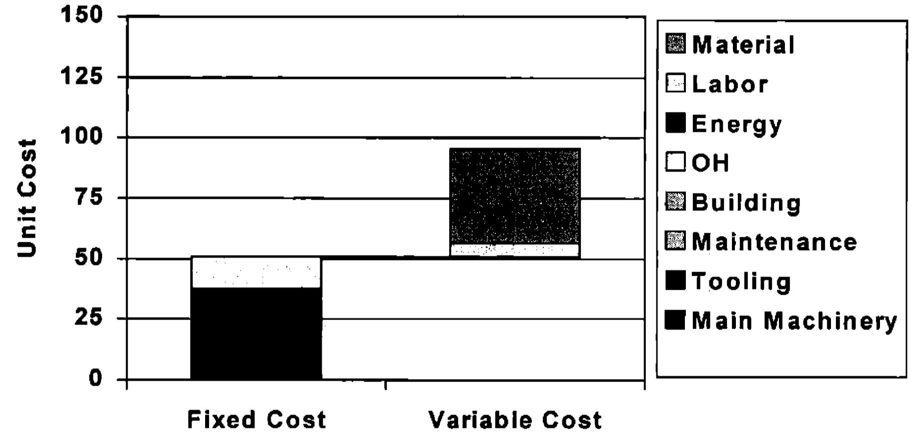


Figure 7-33: Fixed and variable cost for conventional door (process cost only)

7.4.3.2 Case 2: The cruciform door

The overall cost structure of the cruciform door is very similar to the one of the conventional door, because for the cruciform's production the same materials and manufacturing processes are used as in case of the conventional door. The total costs are slightly higher for the cruciform design, caused by additional fastening components to assemble cruciform to the door frame and a small penalty for a reinforced inner trim panel that this door design requires (Figure 7-34).

The relative ratios between processing costs, logistics costs and WIP are comparable to those of the conventional door, for each of the three supply chain steps. Also, the cost distribution across the three steps is very similar to the one of the conventional door.

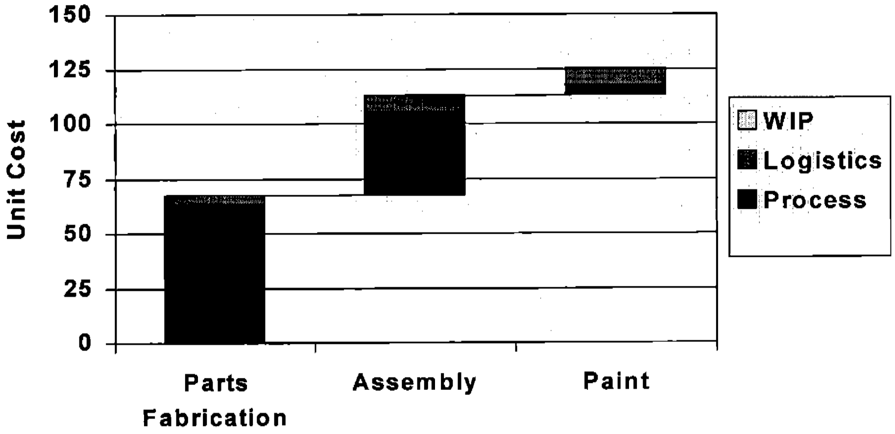


Figure 7-34: Unit cost for cruciform door (baseline scenario)

Similarly, the cost breakdown of the processing costs into fixed and variable costs almost equals the cost breakdown for the conventional door design (Figure 7-35). As for the base case, fixed costs are represented by machine, tooling and overhead costs, while maintenance and building costs are very small. Material dominates variable costs; labor occupies only a small share.

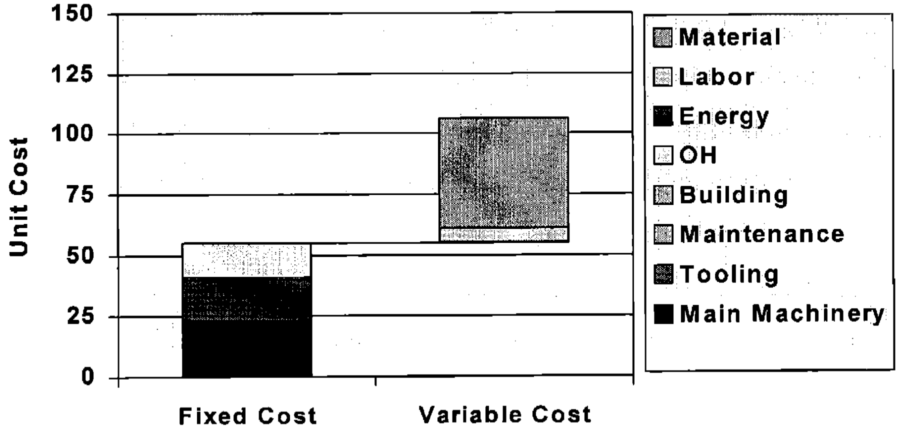


Figure 7-35: Fixed and variable cost for cruciform door (process cost only)

7.4.3.3 Case 3: The cast frame door

As it is the case with the two previous designs, the cast frame door design's processing costs also are significantly larger than logistics and WIP costs. In contrast to the previous two designs, however, this door design exhibits a different cost structure across the supply chain steps: *parts fabrication* represents almost three quarter of the total processing costs (Figure 7-36). The reasons for this effect are twofold. First, this design employs magnesium as material for its main part, i.e., the door frame, aluminum for its outer panel, and a composite for the anti-intrusion beam. All these materials are more expensive on a unit base than the material used for the two previous designs: steel. Second, the application of a complex casting replaces a number of reinforcing components. As a consequence, the costs for the second step, assembly, are reduced.

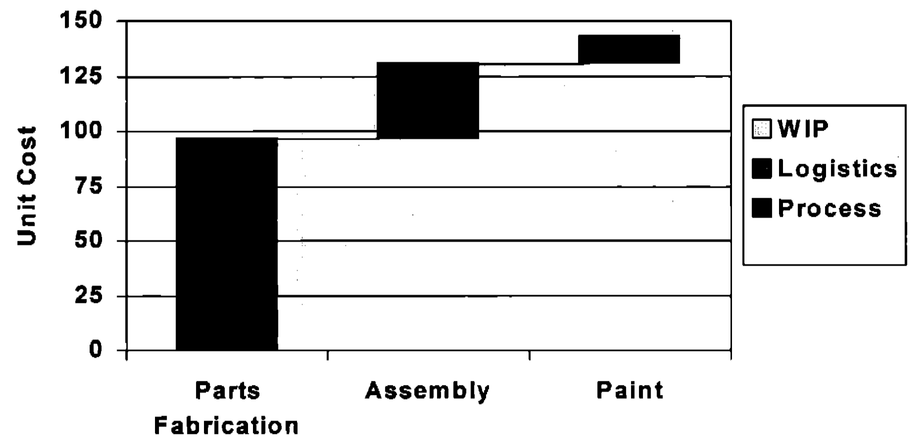


Figure 7-36: Unit cost for cast frame door (baseline scenario)

The cost breakdown in fixed and variable costs (of the processing costs only) also exhibits a different structure relative to the two previous designs. Due to the slow run rate of the die casting machine, machine costs for the cast frame design occupy a higher fraction of the total costs than the costs for stamping in case 1 and 2. In addition, variable costs are even more dominated by material costs, a consequence of the use of expensive materials (Figure 7-37).

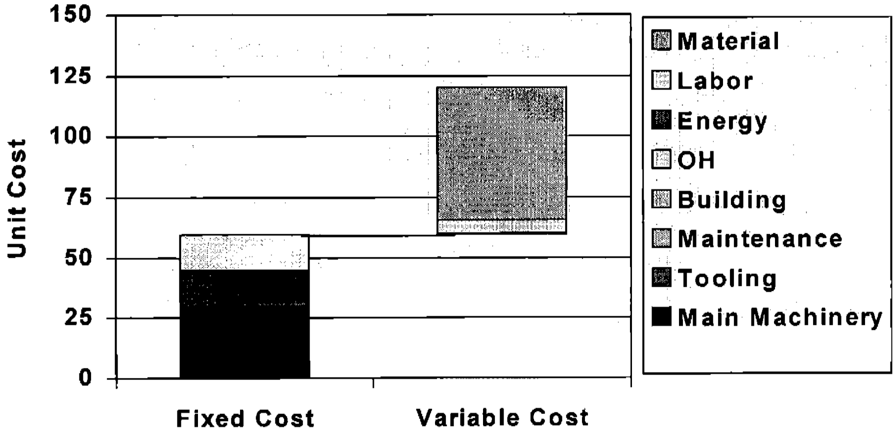


Figure 7-37: Fixed and variable cost for cast frame door (process cost only)

7.4.3.4 Case 4: The extrusion frame door

The cost distribution across the three supply chain steps is again different for the fourth design. Contrary to the cast frame design, the extrusion frame door shows that the assembly stage is responsible for a large portion of the cost, while the one caused by parts fabrication appears reduced (Figure 7-38). The reason lies in the product design and the employed manufacturing processes. For parts fabrication, this design uses extrusion as the main manufacturing process. This process employs relatively cheap tooling and is very fast. It creates, however, profiles that are – while possibly complex in cross-section – relatively simple in their outer geometry, i.e., mostly straight parts. Consequently, this design requires more individual parts to assemble which in turn causes the relatively high costs of the assembly stage.

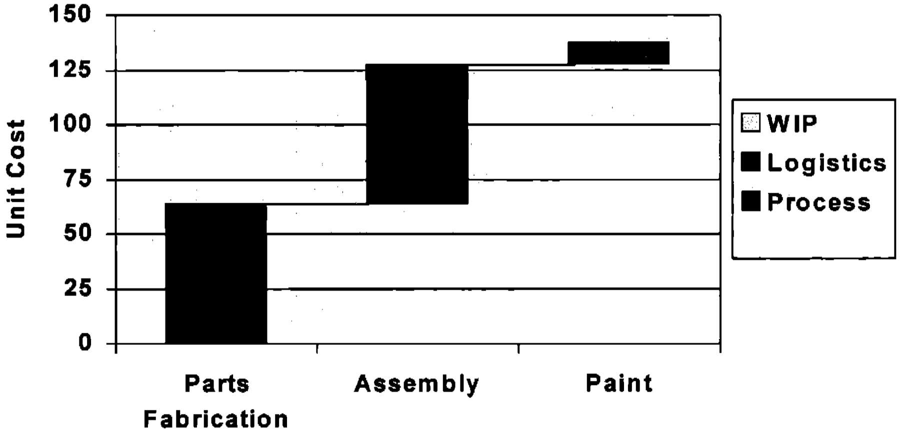


Figure 7-38: Unit cost for extrusion frame door (baseline scenario)

The manufacturing processes employed for this design are also reflected in the cost breakdown structure. The fixed costs are dominated by machining costs (e.g., inexpensive tooling) while the variable costs show that four fifths of them are materials costs. Aluminum, the material used for the extrusions, is relatively expensive compared to steel.

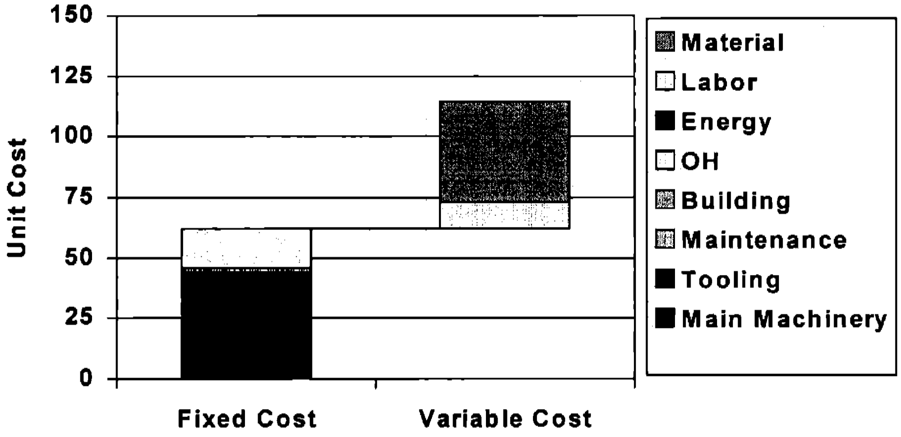


Figure 7-39: Fixed and variable cost for extrusion frame door (process cost only)

7.4.4 Cost Analysis (Sensitivity Analyses)

The previous section discussed the cost structures of all four cases, i.e., product architectures, for a defined baseline scenario. In this section, this baseline scenario is varied (a) to test the robustness of the cost results and (b) to investigate the impact of architectural features on cost.

7.4.4.1 Sensitivity to changes in demand

The previous section's cost analyses are built on a baseline scenario as it is described in section 7.4.3. This baseline scenario assumes an annual production volume of 100,000 units. Since manufacturing processes differ in their fixed cost intensity, they react differently to variations in the annual production volume across which these fixed costs can be spread.

Expensive and dedicated tooling is a well-known factor of high fixed cost intensity. Typically, it is tried to counter this intensity with fast machinery to achieve high production volumes. The door designs 1 and 2, i.e., the conventional and the cruciform door designs, require expensive stamping dies and consequently exhibit cost curves that are very steep for low production volumes, but arrive at low unit costs for high production volumes.

Conversely, inexpensive tooling with short lifetimes tends to resemble variable cost behavior. The resulting cost curves turn flat at comparatively small production volumes. In addition, if the tool lifetime is much shorter (measured in parts manufactured) than the production volume over the program life, the resulting cost curve exhibits steps that represent the addition of new tool sets. The cost curve of the extrusion frame door design is an example for these two effects.

Finally, another explanation for a rather flat cost curve are high variable costs to begin with. For example, expensive materials tend to push the entire curve upwards, independent of the production volume (neglecting the possibility of a volume discount for purchasing large quantities). The cast frame door design demonstrates this effect.

Figure 7-40 compares the volume sensitivity for all four door architectures. Due to their assumed variable behavior, logistics and WIP costs are not considered in this figure.

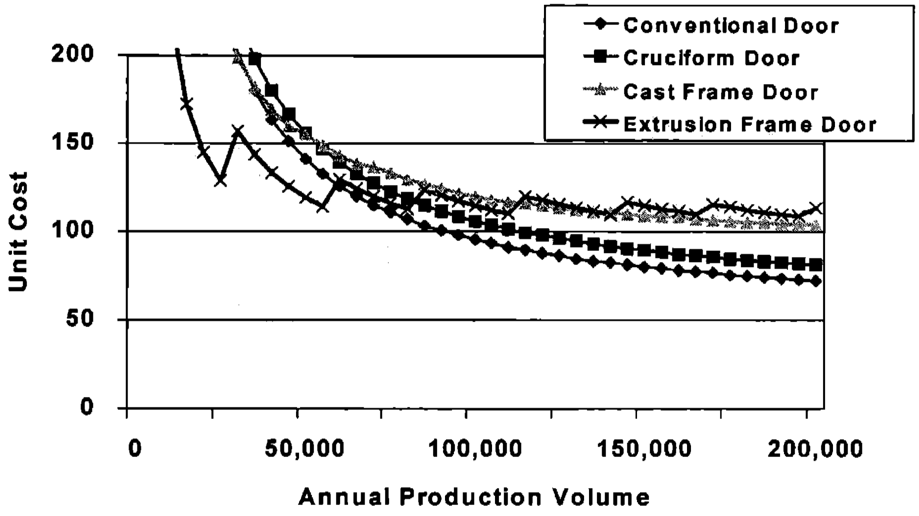


Figure 7-40: Sensitivity to changes in annual demand

A variation of how production volumes can change over the life of a product program is by a change of the length of the program itself. The observable cost effects are very similar to those discussed above. The lower the dependency of fixed costs, the more robust the results of the cost analysis. Figure 7-41 to Figure 7-44 demonstrate the effects for all four door architectures.

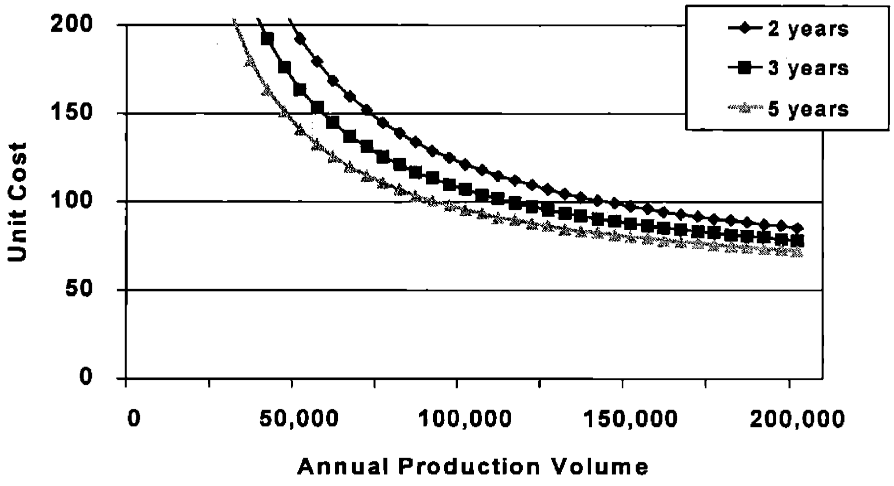


Figure 7-41: Sensitivity to changes in production program life (conventional door)

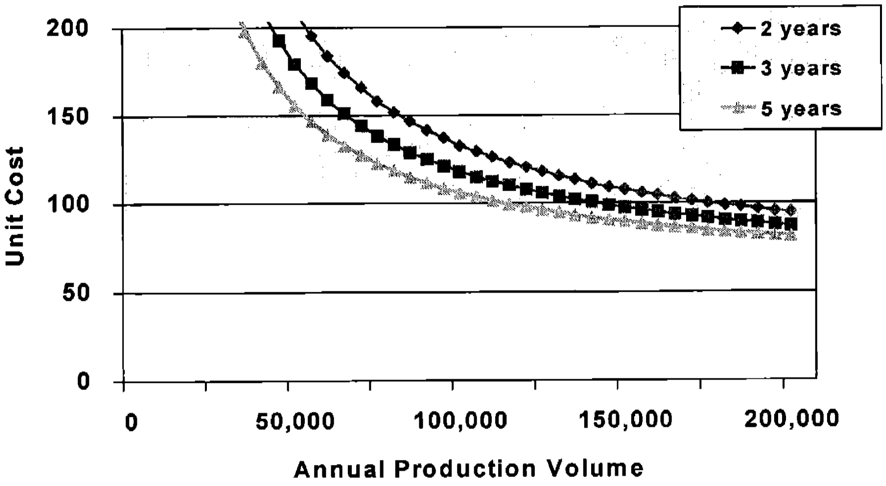


Figure 7-42: Sensitivity to changes in production program life (cruciform door)

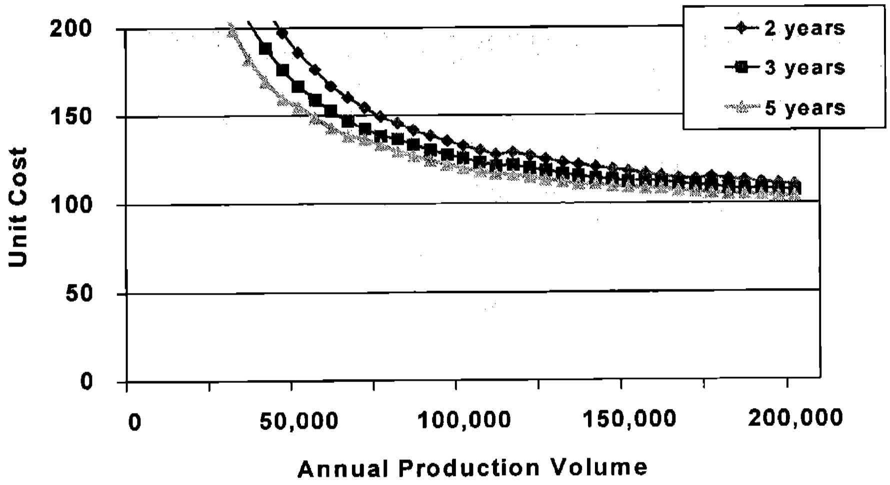


Figure 7-43: Sensitivity to changes in production program life (cast frame door)

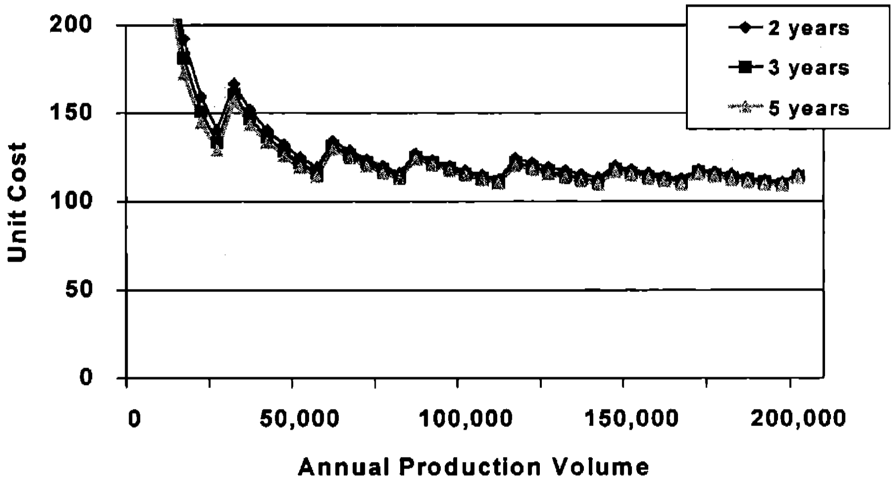


Figure 7-44: Sensitivity to changes in production program life (extrusion frame door)

7.4.4.2 Scenarios to test impact of architectural features on costs

Scenarios are typically divided in two types: Future Histories and Snapshots. While scenarios of the first type stress the dynamic of a system over time, those of the second type present "a cross-sectional view at a single point in time, and structural, rather than dynamic, characteristics are emphasized" (Porter et al. 1991, p.261). In this sense, the sensitivity analyses conducted here with the experimental cases can be seen as scenarios of the second type, as scenarios focusing on the structural understanding of the concept.

Below two scenarios are defined to investigate how architectural differences detected by the framework are associated with cost differences. The two scenarios resemble business strategies often named as benefits in association with modularity.

7.4.4.2.1 Scenario 1: Architectural effects on production costs ('parts commonality')

One of the ideas behind 'modularity' has been the potential for using one module in several members of a product family, i.e., in multiple variants or configurations of a product, or in several generations of a product. As the product architecture analysis has demonstrated, this kind of re-use requires primarily two aspects to be fulfilled. First, the function/part allocation scheme has to allocate the function whose features will not be varied into one chunk. Second, at least on one side of the interface under consideration must be a population that provides alternatives. From a manufacturer's perspective, interface intensity and reversibility are less relevant.

Assuming that the aesthetic appearance on the outside is what drives changes of the door, the structure on the inside is what one would want to keep identical across a product family or subsequent product generations. Using the product architecture mapping method from chapter 3 and focusing on the two relevant aspects reveals the differences in the architectures.

In case of the conventional door, the function *structure* is provided by several parts which simultaneously also contribute to other functions, i.e., the function/component allocation is complex. In addition, the function's level of standardization is very low due to the geometric constraints of this design (see section 7.4.2.1 for details). In contrast, the cruciform door shows a higher level of consolidation of the components providing the function *structure*. Given only a few connection points between the structure and the reinforced outer panel, the standardization level is comparatively higher. Both the cast frame door and the extrusion frame door exhibit even higher level of consolidation than the cruciform door. In case of the cast frame door, however, the level of standardization is almost zero; for the same reason as it is for the conventional door: geometric determined complex shapes. The extrusion frame door, similarly to the cruciform door, demonstrates a significantly higher level of standardization, compared to the other door designs.

As a consequence, the re-use of the structure for different model lines or as a 'carry-over' component for a door's next generation with a slightly changed outer panel is really possible only for the cruciform door and the extrusion frame door. In case of the other two door architectures, the function *structure* is much more difficult to isolate, i.e., its distance to the 'ideal' point where there is a one-to-one relationship is larger, or its level of standardization is much lower, or both. Figure 7-45 illustrates the differences in (a) distance from the one-to-one coordinates and (b) in level of standardization for all four door architectures.

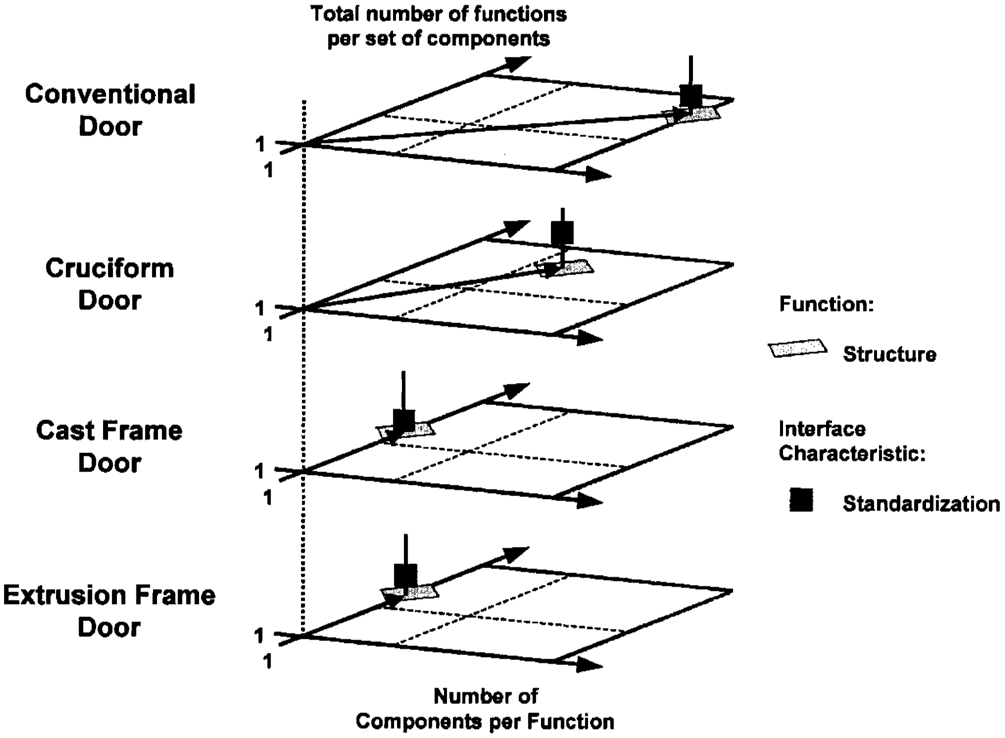


Figure 7-45: Product architecture assessment for 'parts commonality' strategy

The result of this architectural choice is the potential of partial re-use of the product. In other words, for a fraction of the product the production volume can be increased. Since the cost curve of the extrusion frame door is relatively flat, changes in production volume are unlikely to change its competitive position (at least above very small production volumes). For this reason, this scenario analysis investigates the cost impact of the difference along the discussed architectural elements for the cruciform door compared with the conventional door.[144]

For the assumption that the demand for the cruciform alone could be doubled, Figure 7-46 shows the cost savings of this partial volume increase (the solid line represents the production costs of the cruciform alone) for both low and high production volume regimes. The low volume regime is set at 50,000 units; the high volume regime at 150,000 units per year. If the production volume for the cruciform could be increased from 50,000 to 100,000 units, the accompanying

---

[144] This choice offers the additional benefit of controlling for alternative reasons for cost differences such as different materials or manufacturing processes.

savings made the cruciform door as a whole more cost effective than the conventional door, i.e., in this case the savings are larger than the cost difference between the two door architectures. In contrast, for large production volumes (150,000 and above), the savings from doubling the cruciform production volume to 300,000 are too small to shift the cost leadership between the two door architectures. Since the production costs are a non-linear function of the production volume, the benefit of 'parts commonality' is itself volume dependent.

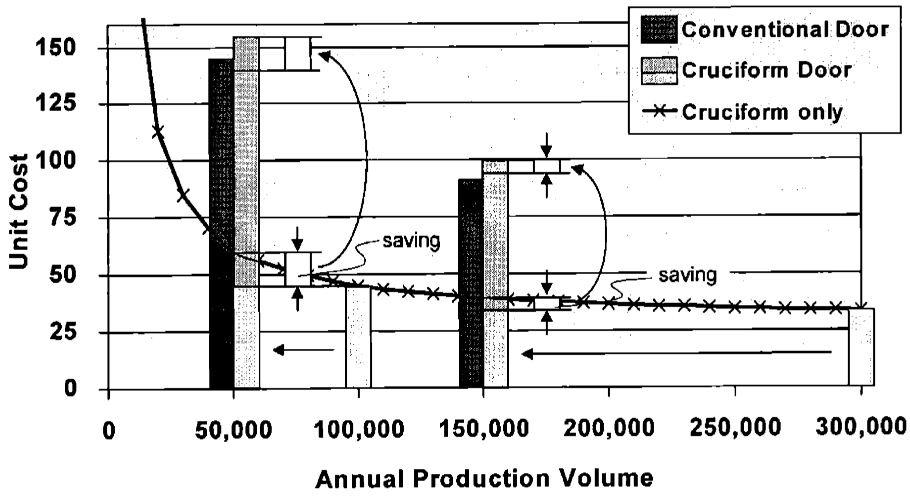


Figure 7-46: Saving effects of parts commonality vary with production volume

This first scenario analysis provides two major insights. First, indeed only a few features of the bundle described in a complete product architecture mapping are responsible for a cost saving effect for a strategy like 'parts commonality.' Second, the cost saving effect itself is dependent on the fixed-cost intensity of the chosen processes, in this case the manufacturing process. In other words, while the mapping process guides one to the relevant individual architecture characteristics, the magnitude of the concrete cost effects are context dependent and need to be analyzed on a case by case basis.

7.4.4.2.2 Scenario 2: Architectural effects on logistics and WIP costs ('postponement' or ' late customization')

The second scenario focuses on a different part of the costs of the value chain: logistics costs. Logistics costs are affected by a variety of factors (Figure 7-47). Storage and transportation costs are primarily affected by a product's size and weight, the process duration and transportation distance. As discussed earlier, these costs have to be evaluated on a case-by-case basis because they are case specific (as are exogenous factors like wages or taxes). Consequently, this scenario focuses on how the product architecture affects WIP costs, in particular with respect to product variety.

Influencing Factors Cost types	Product Size	Product Weight	Process Duration	Distance / Transport mode	Product Variety
Storage	yes	- -	yes	- -	yes
Transportation	yes	yes	- -	yes	yes
WIP	- -	- -	yes	- -	yes

Figure 7-47: Factors influencing logistics costs

The idea to reduce WIP costs through modularity is that some fraction of a product can be used for other products as well. As a result, the total amount of stocked items can be reduced through demand pooling of the 'multi-application' component. To test the extent to which the architectural differences can influence the costs for WIP, I vary the baseline cases by introducing demand uncertainty and variations of a product feature. Consider that the door structure is now offered in two variants, in particular concerning the aesthetics. This scenario can be envisioned as offering the product in two different colors. For demonstrative purposes, the scenario is limited to step 3 only, i.e., WIP of painted door structures.

With respect to the product architecture, two features require particular attention for this strategy (Figure 7-48). First, it should be possible to isolate the function to be varied in a component, and second, the interface characteristic *reversibility* is very important because it enables the re-sequencing of production processes. Low interface reversibility often determines a certain order of production processes (e.g., welded interface requires assembly to be completed before the painting process starts to avoid damaging the paint).

Concerning the function-component allocation, the product function *aesthetics* is carried in all architectures by the component *outer panel.* In case of the conventional door, the outer panel also contributes to *structure,* as it does in case of the cruciform door as the *door frame.* In contrast, the cast frame door presents *aesthetics* at the corner of the map, i.e., at the point of perfectly modular function-component allocation. Finally, the extrusion frame door provides the function *aesthetics* with two components: door panel and window frame.

While fairly similar with respect to function-component allocation, the product architectures show significant differences along the architectural feature *reversibility.* Both the conventional door and the cast frame door connect outer panel to the rest of the door with spot welds and hem flanging. Both processes have to be completed *before* paint is applied. In contrast, the cruciform door and the extrusion frame door provide mechanical fastener as mechanisms to join outer panels and structural components. As a consequence, the joints can be formed *after* the components receive the paint. Note that the bolt joint mechanism for the cruciform door is assessed on a medium level regarding reversibility, whereas the simple snap-fit connection of the extrusion frame door is considered on a high level with respect to this interface characteristic.

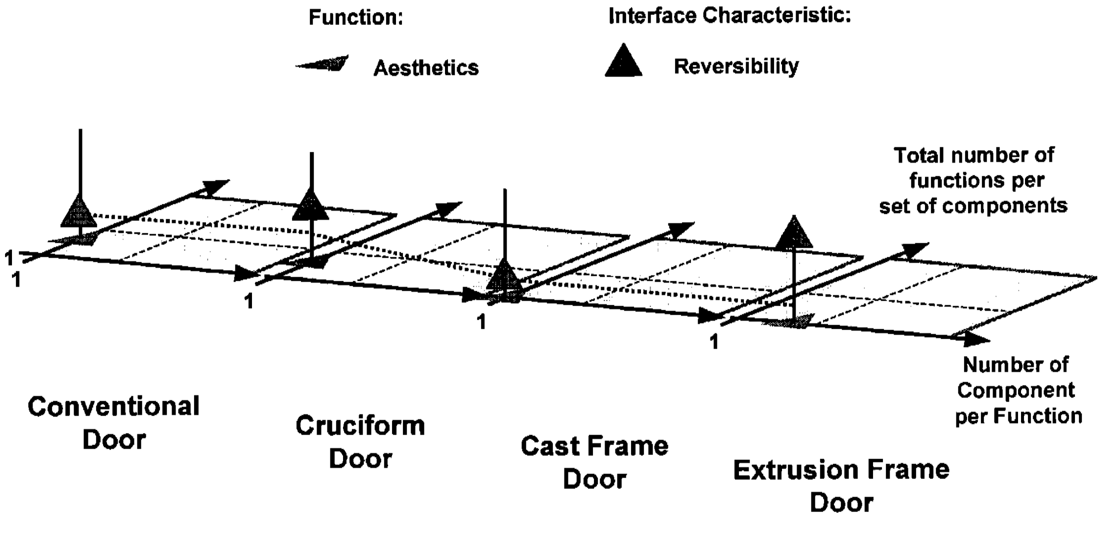


Figure 7-48: Product architecture assessment for 'late customization' strategy

To model the WIP cost effects of this product strategy, the door structures are modeled as products composed of two components. One component is considered common for both versions of the product; the other differentiates the product with the feature under consideration. In case of the door structures, this means that the door structures are identical while the outer panels provide the distinguishing feature. Following this idea, then the interface characteristic *reversibility* separates the four doors into two categories: A and B. In category A, all colors customers might order need to be on stock to provide supply at a chosen service level. The conventional door and the cast frame door fall into this category. In category B, only the outer panels need to be stocked according to the demand for individual colors, while the inventories for door structures can be pooled, because they can be painted in any color.[145] This system can now be viewed as an assembly-to-order system. Figure 7-49 illustrates this set-up for two customer features, i.e., colors.

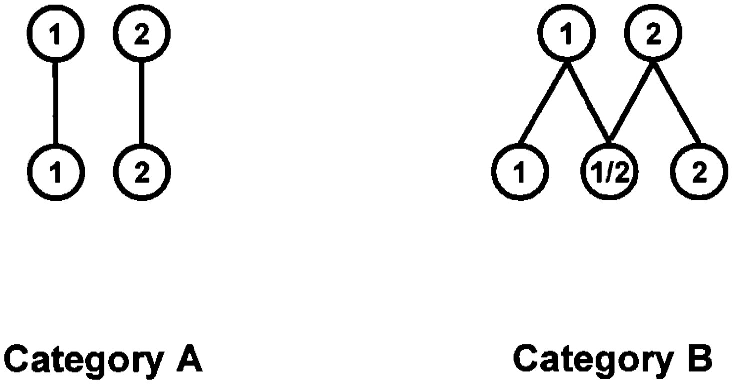


Figure 7-49: Setup for 'postponement' strategy

The potential for savings in WIP can be modeled by applying a model developed by Baker et al. (1986). For the demand, independent uniform distributions are assumed for color 1 and color 2; between 0 and b1, and 0 and b2, respectively. In addition, the aggregate service level (ASL) is applied as a measure of the probability that all demand is met. The different product architectures present different opportunities to minimize the stock required to provide products at the chosen ASL. Figure 7-50 illustrates the extent to which the stock level can be reduced for either category. For example, for a ratio of 0.75 between b2 and b1, and an ASL of 0.94, product

---

[145] I assume equal production costs for any color and batch size.

architectures of category A allow to reduce the stock with respect to total demand, i.e., b1+b2, by 3.4%, whereas product architectures of category B permit to reduce the stock by 8.6%.

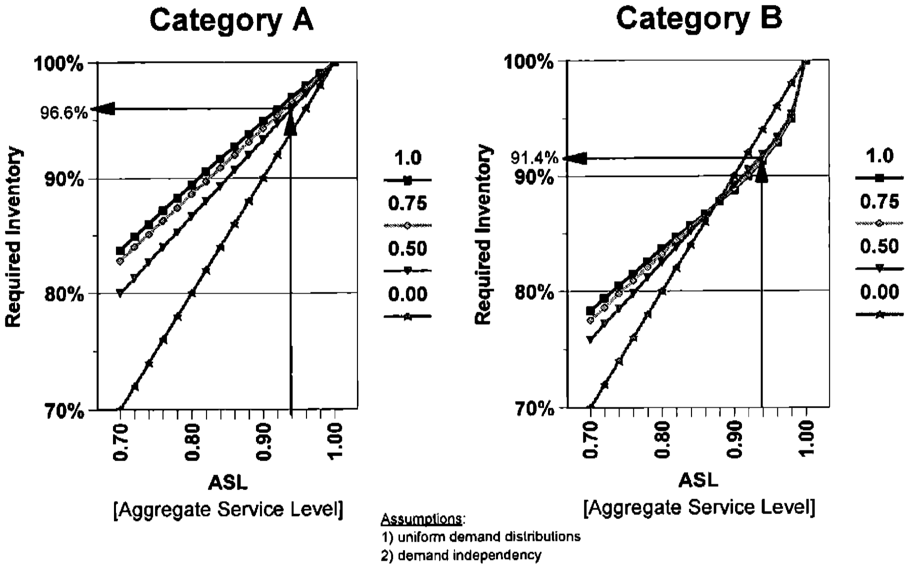


Figure 7-50: Inventory requirements for category A and B

The calculations for Figure 7-50 are unit based. They simply 'count' the total number of components required to be stocked, that is, the common and variant-specific components. To find the true cost effects, the calculations need to be adjusted for the real value that the individual components represent.[146]

In case of the cruciform door, the cruciform structure represents 51.9% and the painted outer panel with frame 48.1% of the total value. For ASL=0.94 and b2/b1=0.75, this results in an inventory adjustment factor of 0.996. This equals lowering the required inventory to 91.0% of b1+b2. Consequently, the per unit WIP cost fall to the same level.

The case of the extrusion frame door is an example of the effect of concentrating a large fraction of the total value in the common component. The extrusion frame represents 83.3% of

---

[146] This is particularly important because while the stock of the common part can be lowered, the stock for the product specific component increases (Baker et al. 1986, Gerchak et al. 1988). The cost savings calculated in the above scenario represent the net savings.

the total value, while the variety specific component, the door outer panel, stands for only 16.7%. Again, assuming ASL=0.94 and b2/b1=0.75, the value adjustment is equal to lowering the inventory down to 57.2%. In other words, the WIP cost for the final step of the supply chain can be lowered by 42.8% through risk pooling in this scenario.

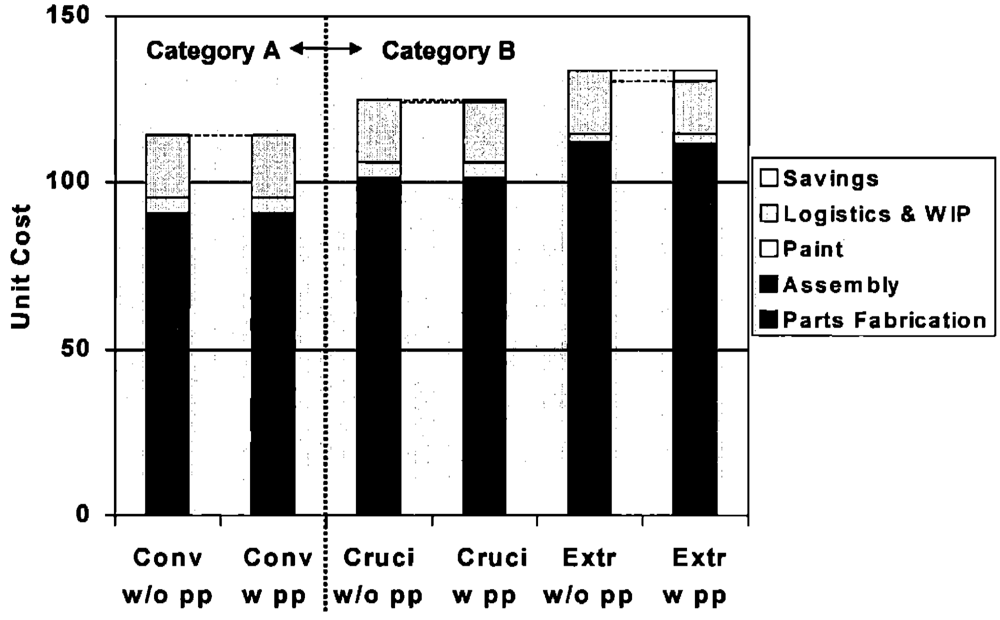


Figure 7-51: Real inventory saving effects (adjusted for value)

Although the savings for the extrusion frame door are recognizable, they are still small compared to the total supply chain costs. Main reason for this effect in these case experiments is the dominance of the manufacturing and assembly costs over all other costs (see also section 7.4.3).

In more general terms, however, WIP cost savings that are enabled through the architectural features function containment and interface reversibility can amount to significant portions of the total costs under certain conditions. Their absolute and relative size depends on (a) the value of the product, (b) the number of variants and thus uncertainty in demand, and (c) the speed of the value chain, i.e., the time the products remain in the supply chain (e.g., long distance transport).

Finally, this scenario has only tested the potential inventory saving effects due to product architecture features in step 3 of the particular supply chain. Increasing upstream volatility of supply chains ('bullwhip-effect') makes additional inventory benefits upstream likely for architectures of category B. Generally, high levels of uncertainty, particularly together with long time lags, i.e., slow supply chains, can amplify this effect and create significant additional costs in the supply chain (Levy 1994).

7.5 Chapter Conclusion

With help of the product architecture framework developed in chapter 3 and process-based cost models, in this chapter I conducted case studies as controlled experiments to test linking the two concepts, i.e., modularity and cost.

The case studies demonstrate how different features of a product architecture can have different cost effects along the supply chain, both in size and location. For example, while function containment and a minimum of alternatives for interface participants in case of the cruciform door can help reducing production costs through partial volume increase, it is the interface reversibility that permits a postponement strategy to reduce inventory costs. On the other hand, the particular value represented by common and variety-specific components, can amplify or reduce the real savings effects.

One insight of these case studies is that in cases where total supply chain costs are strongly dominated by production costs, caution should guide local optimization approaches in areas other than production, such as WIP. For example, it is possible that some architectural features that are beneficial to logistics costs create cost penalties in the production stage that outweigh the benefits. Also, depending on the ownership of the different supply chain steps, these shifted costs may be borne by different players.

In the remaining chapter I will discuss the case study results in greater detail and interpret the implications for strategy, business policy and design. Finally, I will provide potential avenues to expand this work.

8. IMPLICATIONS FOR PRODUCT STRATEGY AND DESIGN STRATEGY

8.1 Chapter Introduction

To assess the implications of this research for product strategy and design strategy, the chapter will review the foregoing analyses on multiple levels and from several perspectives (Figure 8-1).

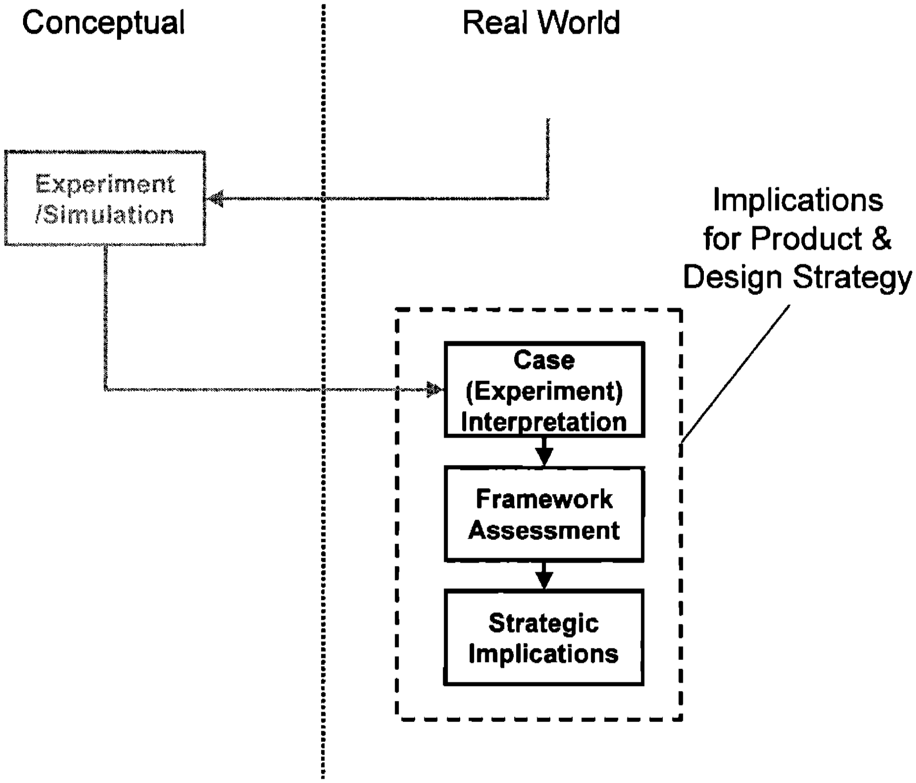


Figure 8-1: Content of chapter on implications for product & design strategy

It begins with an analysis of the results of the experimental case studies and points out the type of insights the studies allow as well as those they do not allow. Next, the analysis moves one level up, and discusses the usefulness (and limitations) of the descriptive product architecture framework. Through a process of distinguishing and measuring the different architectural features, the framework provides a tool to unbundle *modularity* and – together with a costing procedure – to link individual aspects of the product architecture to costs.

This section also provides a brief review of merits and limitations of using cost as a product design decision variable. Next, the perspective is switched to a business view and the range of potential applications of the framework in a business setting is discussed.

Finally, the chapter closes with a discussion of some of the future research avenues that could extend this work around the product architecture framework.

8.2 Case (Experiment) Interpretation

The experimental case studies have demonstrated the applicability of the product architecture description methodology to explore the linkages between individual architectural characteristics and various costs along the value chain. Selected for illustrative purposes, the case study results cannot be used to derive statistically significant conclusions.

However, these case studies do provide valuable insights in how early design decisions with respect to product architecture affect costs at various stages in the product life cycle. There are several reasons for this research approach. First, and as explained in detail in chapter 6, by selecting extreme cases, it is possible to explore the impact of parameter variation along individual dimensions of the descriptive framework. Without requiring statistical representation of the population, the case samples rather push the envelope of the framework's power in defining and characterizing *possible* designs, instead of *representative* ones.

Second, the experimental case study approach enables to freely choose also environmental conditions to reflect possible conditions rather than representative ones. Preferably, scenario definitions are defined as boundaries (worst case, best case) for the space of possible environment conditions.

Finally, case studies that are conducted as controlled experiments help isolating the factors of interest. Akin to the process for controlled laboratory experiments, this approach permits one to test the internal validity of the findings in that the control for all other competing explanations of the observed effects has been achieved. As a consequence, the approach helps exploring the selected effects in great detail. With respect to the modularity debate, the approach presented in this work has made it possible to remove some of the ambiguity associated with the term 'modularity' and its claimed effects and, instead, to demonstrate links of finer granularity between product architectural features and costs. Figure 8-2 illustrates this higher level of precision on both sides of the link, i.e., for the product architecture as well as for the costs. Figure 8-2 replaces Figure 1-1 in the search for how to link modularity and cost.

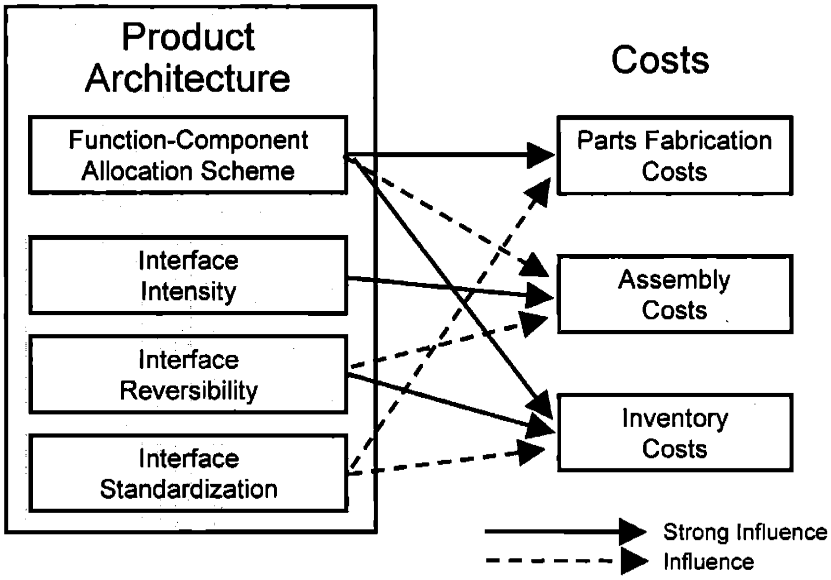


Figure 8-2: New model for how to link modularity and cost[147]

At least for this exploratory phase of research, the experimental case study approach demonstrated its advantages. It is important to note, however, that a limited number of case studies is unlikely to cover *all* possibilities. Cost consequences identified in the case studies are not independent from the choice of material/process combinations. Therefore, more case studies will be needed to refine the understanding of the implications of different kinds of combinations. Ultimately, these results could be collected to build a repository of material/process combinations' effects on costs or other consequences of interest.

8.3 Assessment of the Product Architecture Framework

In order to assess the framework's contribution to a better understanding of modularity and its effects on costs, a brief discussion on cost as a decision variable is helpful.

---

[147] The arrows indicate the relationships found in the case studies presented in chapter 7. They may vary for other material/process combinations.

As the discussion in chapter 4 about various costs that occur throughout a product's life cycle has shown, the act of focusing on the costs of a particular phase is already a selection and thus, constitutes a boundary setting for all consecutive analyses. If more than one phase is to be taken into account, trade-offs between costs of one phase versus those of another are to be expected.

But even if the life-cycle horizon is limited to, say, the producer's point of view (i.e., only costs of production are incorporated in the analyses), costs represent only a subset of the decision variables. For example, a profit maximizing enterprise also needs to consider the revenue effects of product architecture decisions, in addition to the cost consequences. These revenue effects can be directly consumer related, i.e., they reflect her willingness to pay for a certain product (or a certain version of it), in other words they reflect the trade-off curves for individual product attributes. Alternatively, the revenue effects can be linked to the market via a time component. In this case the assumption is that market demand for the product under consideration decreases over time and a delayed market introduction reduces the total revenue potential.[148]

One way of looking at this question is to consider the profit margin of a product. Cooper and Slagmulder, for example, recommend a focus on activities that increase sales when the margins are high, while directing efforts towards cost (reductions) when the margins are low (Cooper and Slagmulder 1997, p. 229). When cost and revenues show large differences in magnitude or in sensitivity, or both, a focus on the wrong decision variable can have untoward effects. For instance, evaluating the economic effects of Design-for-Manufacturing, Ulrich et al. find in a case of redesigning a housing of an instant film camera that the revenue penalties due to delayed market introduction caused by the redesign vastly outweighed the cost savings. In their particular case example, however, the profit/revenue ratio was over 95% and, consequently, a delay in sales had a devastating effect on revenues compared to the minor cost savings achieved through redesign (Ulrich et al. 1993). A similar example is presented by Hu and Poli, who compare the time-to-market of two similar products, one made from several stamped parts, the other made as a single injection molded part (Hu and Poli 1997b). Hu and Poli include tool making and assembly as well as the manufacturing of the first batch in their analysis. They

---

[148] In addition to the market decrease as measured in units, there is an effect of margin dilution over time when competition sets in.

suggest that lowest manufacturing cost should not be the only decision variable, but rather be accompanied by an understanding of the effects of design decisions on time-to-market. Even so, their results are highly sensitive to some basic assumptions such as lot size, time value, and operation conditions.

In sum, while revenue and time are also important parameters in addition to cost, a thorough understanding of the relative size and sensitivity of the parameters is required before one should draw strategic conclusions.

There have been models developed that incorporate revenues and costs in their optimization, i.e., profit maximization, approaches. However, while conceptually powerful, those models mostly reduce the architectural question to one of combinatorial configuration to limit the problem's complexity. For example, Blackenfelt chooses this approach in determining the optimal configuration of a family of lift tables (Blackenfelt 2000). Similarly, Ramdas and Sawhney in their analysis of the case of analog quartz wristwatches limit the architectural differences that they consider to a configuration problem: "We assume that each product uses one unit of each component." (Ramdas and Sawhney 2001, p. 27)

In order to keep the optimization task across revenues and costs manageable, most of these existing models have strong limitations with respect to the product architecture differences they consider. They are powerful tools to assess cost and merits of product variety - under the assumption that the products are more or less identical in structure, and variety is determined only by variation of components. This constraint, however, means the products' architectures are close to identical. Essentially, most of these models test the profit potential of various variety options *within* a (given) product architecture. As a consequence, it is very difficult to translate their insights into design advice for product architectures that differ along multiple dimensions.[149]

This is precisely the reason why this thesis (a) argues for a separation of description and evaluation tasks with respect to modularity, and (b) develops a descriptive approach to analyze and compare product architectures. The only way to understand the precise impact of early design choices with respect to product architectures is to disentangle the bundle 'modularity' and

---

[149] However, comparative testing of multiple, different architectures with help of these models might allow to asses each architecture's effects on variety.

unambiguously describe the differences a product can exhibit along each of the multiple dimensions. Once the comparability is established, then in a second step the product architectural features can be linked to other decision variables of interest, cost being an important one of them, albeit not the only one.

As such, the product architecture framework can help to bridge the disconnect between the engineering and management worlds. By linking strategic business goals to required product characteristics and those in turn to economic consequences, it supports the communication between these two world-views. It can translate business strategies into concrete design guidelines by directing the focus onto the most relevant product characteristics. With the measures developed in this thesis for the individual dimensions on levels below modularity, the framework enables the analyst to characterize the key design dimensions and to connect them to consequences of interest.

In this work, the product architecture description framework has been applied to products that are similar in functionality, that is, across cases it was possible to control for the parameter 'functionality'), and the results were satisfactory. An open question at this point is the extent to which the framework can be used to compare architectures of products that are very different. First tests have been conducted with products whose functionalities differ beyond any meaningful comparison, e.g., computer and automobiles. Although difficult to compare on an individual function base, the product architecture description methodology produced promising results by making differences in overall patterns visible (Fixson and Sako 2001). Nevertheless, more work is needed to identify the limits and to expand the applicability of the framework.

8.4 Implications for Product Strategy and Product Design

Today's companies are operating in market environments that are becoming increasingly dynamic and fast moving. Customer populations are becoming more heterogeneous and product life cycles are steadily shortening. As a result, decisions that must be made early in the development process have a strong impact on the success or failure of a product. While concurrent engineering, i.e., linking detailed design decisions to cost effects in the manufacturing stage, has been developed over the past decade, an even more holistic approach will be required in the future. An approach is needed that makes the linkages between market strategy (product variety, multiple markets, multiple product generations, etc.), design decision, and their consequences throughout multiple product life cycle stages more visible. The design decisions made in early concept development, i.e., the determination of the product architecture, represent a powerful link between product strategy and business success.

The product architecture framework developed in this research can serve companies in two ways. First, it can be used as a guideline to help focusing the discussion on design decisions critical for the product and company under consideration. For an existing product strategy, it can help to identify architectural characteristics that best serve that strategy (see Figure 8-3 and Figure 8-4 with examples based on the cases in chapter 7).[150] With respect to the cost effects, a company might build over time a knowledge base on material/process combinations and their cost profiles. Such a repository then may support the process to search for solutions meeting certain architectural requirements.

The second way in which the product architecture framework can help is through the improvement of a company's strategy planning capabilities. In conjunction with the costing procedure, the framework can help to make the price visible that is to be paid for a chosen product strategy with a particular product architecture. A broader understanding about the economic consequences of certain product architecture choices may support the strategy development in having better information early on.

---

[150] The graphics show example results from the case studies of chapter 7 and are read as follows: Beginning with the product strategy a life-cycle phase is identified which is of particular interest, with respect to cost (or other decisions variables). The graphics show two examples, one from the manufacturing perspective, one from the inventory of finished goods point of view. For manufacturing processes with strong scaling effects, commonality is very important to achieve economies-of-scale. The product architecture characteristics that need to be focused on then are the function-component alignment and the interface standardization. If, however, the cost focus is on the inventory, a postponement strategy may be advisable. Key for a successful application of a postponement strategy is to focus the design work on interface reversibility.

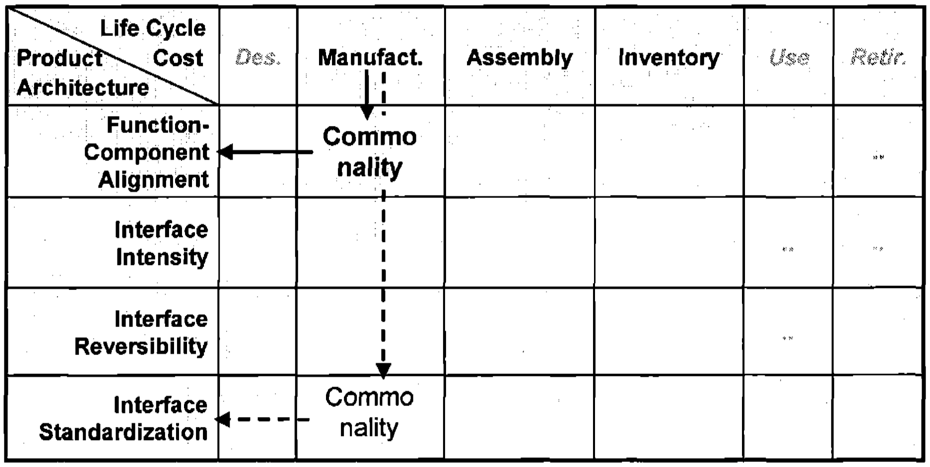


Figure 8-3: Translating strategic goals (e.g., cost reduction) into focused design advice (example taken from scenario 1 of chapter 7)

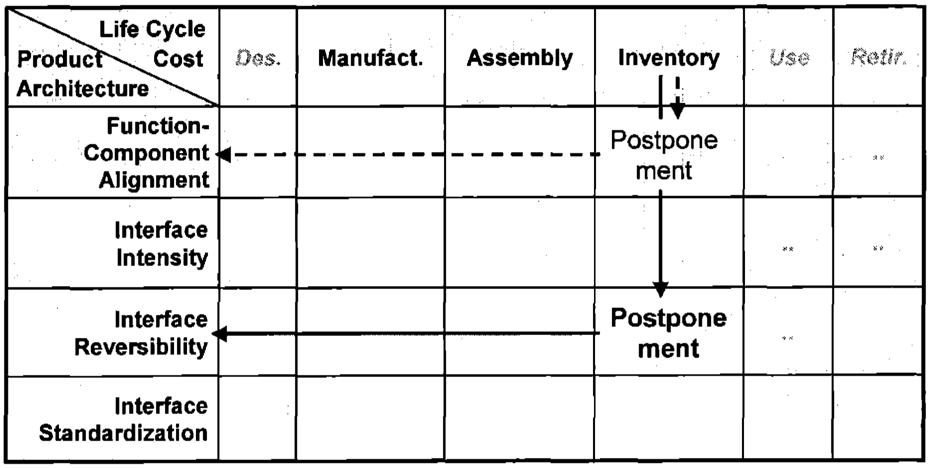


Figure 8-4: Translating strategic goals (e.g., cost reductions) into focused design advice (example taken from scenario 2 in chapter 7)

It seems that, rather than promoting 'modularity,' 'build-to-order,' 'platforms,' or 'late customization' as optimal strategies across the board, it is advantageous to better understand the structure of the underlying relationships between early design decisions and economic and market consequences. Once this knowledge exists, one may conclude which parts of a product to design with 'modular characteristics' and which ones without, given identified environmental conditions.

8.5 Future Work

Every work offers avenues for improvements or extensions, or both. Below four directions are presented which are promising future research directions that build on this dissertation.

First, as discussed earlier, the descriptive product architecture framework can be used for larger empirical studies. These studies could compare products within and across industries, or tie them to performance measures of interest, i.e., cost, revenues, quality, etc. This work could further improve the understanding of type, strength, and conditions of the effects triggered by concepts such as 'modularity.'

Second, the framework could be applied to study product architecture development over time. Successive generations of a product could be described and measured to investigate whether there are patterns in which product architectures evolve. Currently, it is claimed that more and more products will become modular. I conjecture that there are multiple forces working simultaneously (product technology improvement, manufacturing process improvement, ratio of provided to requested product performance, increasing interconnection between formerly distinct products, etc.) and that the resulting direction of product design is not necessarily modular. The framework presented in this thesis could help to uncover some of the ways in which product architectures change.

Third, it has been recently suggested that the notion of concurrent engineering should be expanded to incorporate formal consideration of supply chain operations. Fine calls this approach three-dimensional (3D) concurrent engineering (Fine 1998). The descriptive product architecture framework could inform the development of integrated trade-off models for 3D-concurrent engineering. Potentially, it could even guide the discussion for a multi-dimensional model that incorporates multiple product life-cycle stages.

Finally, the product architecture framework may serve as a stepping-stone to rethink the understanding of business models and product categories. With changing customer demands, the translation of customer needs into functional requirements, and consecutively into products, may shift towards considering complete customer experiences rather than product categories. This in turn may trigger the understanding of products more as 'tools' for larger experience rather than ends for themselves. As a result, the meaning and comparative weight of a product's functionality may change. Viewing products from this perspective could have a profound impact on the way product architectures are understood, developed, and marketed.

9. APPENDIX A: MODULARITY LITERATURE

The two tables below list the references reviewed for this dissertation. The list encompasses articles, conference papers, working papers, and books. Every reference is assessed from the three perspectives *systems, hierarchy,* and *life cycle.* The legend on page 230 explains the meaning of the symbols used in the table. The entire list of articles is split into two groups: one more engineering related, the other more management related (see Table 9-3 on page 231 for a coding list).

Table 9-1: Modularity in the literature - engineering section

Reference		Systems Perspective		Hierarchy Perspective						Life Cycle Perspective				Industry / Product Example
#	Author(s)/ Year	Description/ Variations of Elements	Description/ Variations of Relations	Techn. Modu-larity	Busin. Modu-larity	C	M	P	F	Des.	Prod.	Use	Retir.	
1	Allen and Carlson-Skalak 1998	1-2	(X)	X			X	X				X	X	Video cassette
2	Coulter et al. 1998	1	(X)		X	X		X					X	Automotive center console
3	Dahmus and Otto 2001	2		X		X		X				X		Document handling system of a copy machine
4	Dahmus et al. 2001	2-3		X		X		X	X			X		Family of electric cordless drills
5	Du et al. 2000	1-2		X		X		X			X	X		Office chair

Reference		Systems Perspective		Hierarchy Perspective						Life Cycle Perspective				Industry / Product Example
#	Author(s)/ Year	Description/ Variations of Elements	Description/ Variations of Relations	Techn. Modu-larity	Busin. Modu-larity	C	M	P	F	Des.	Prod.	Use	Retir.	
6	Erixon et al. 1996	2-3	(X)		X		X	X			X	X	X	*Concept only*
7	Eppinger et al. 1994	1-2	X		X	X		X		X				*Concept*
8	Gonzalez-Zugasti et al. 2000	2			X	X		X	X			X		Space craft
9	Gu et al. 1997	2	X	X		X		X				X	X	Vacuum cleaner
10	Huang and Kusiak 1998	2-3	X	X		X		X			X			Desk lamp & Electric motor
11	Ishii et al. 1995	2	(X)		X		X	X	X		X			Refrigerator door
12	Jiao and Tseng 1999	2	(X)		X		X	X	X			X		Power supply units
13	Kota and Sethuraman 1998	2			X	X			X	(X)	X			Walkman
14	Kota et al. 2000	2			X	X			X	(X)	X			Walkman
15	Martin and Ishii 1996	2	(X)		X		X	X	X		X			Refrigerator door

Reference		**Systems Perspective**		**Hierarchy Perspective**						**Life Cycle Perspective**				**Industry / Product Example**
#	Author(s)/ Year	Description/ Variations of Elements	Description/ Variations of Relations	Techn. Modu-larity	Busin. Modu-larity	C	M	P	F	Des.	Prod.	Use	Retir.	
16	Martin and Ishii 2000	2	X		X		X	X	X			X		Ink jet printer; thermoelectric water cooler
17	McAdams et al. 1998	3		X				X	X			X		Beverage brewers & material removal products
18	Newcomb et al. 1998	2	X		X		X	X					X	Automotive center console
19	Siddique et al. 1998	2	X		X			X	X		X			Automotive underbody
20	Siddique and Rosen 2000	2	(X)		X			X	X		X	X		Coffee maker
21	Stone et al. 1998	3	(X)	X		X	X	X				X		Electric screw driver
22	Stone and Wood 2000	3		X			X	X				X		Hot air popcorn popper
23	Stone et al. 2000a	3	(X)	X			X	X				X		Lignite removal system & Electric wok

Reference		Systems Perspective		Hierarchy Perspective						Life Cycle Perspective				Industry / Product Example
#	Author(s)/ Year	Description/ Variations of Elements	Description/ Variations of Relations	Techn. Modu-larity	Busin. Modu-larity	C	M	P	F	Des.	Prod.	Use	Retir.	
24	Stone et al. 2000b	3		X			X	X	X			X		Electro-mechanical devices
25	Sudjianto and Otto 2001	3		X			X	X	X			X		Family of electric cordless drills
26	Tseng and Du 1998	1-2	(X)		X			X	X	X	X	X		Power supply switch
27	Tseng and Jiao 1996	1-2	(X)		X		X	X	X			X		Power supply for pulse width modulation
28	Tseng and Jiao 1998	1-2	(X)		X		X	X	X			X		Power supply device
29	Ulrich and Eppinger 2000	3	(X)	X		X	X	X	X	X	X	X		Motorcycle
30	Whitney 1993	1-2	(X)		X	X		X	X	(X)	X			Automotive-panel meter; - Radiator; - Alternator
31	Wilhelm 1997	1-2	X		X		X	X	X		X			Automobile
32	Yu et al. 1999	1-2			X		X		X			X		Toaster & Instant camera

Table 9-2: Modularity in the literature - management section

Reference		**Systems Perspective**		**Hierarchy Perspective**						**Life Cycle Perspective**				**Industry / Product Example**
#	Author(s) Year	Description/ Variations of Elements	Description/ Variations of Relations	Techn. Modu-larity	Busin. Modu-larity	C	M	P	F	Des.	Prod.	Use	Retir.	
33	Ahmadi et al. 2001	1-2	(X)		X		X	X		X				Rocket Turbopump
34	Baldwin and Clark 2000		X		X		X	X		X				Computer
35	Baker et al. 1986		(X)		X	X		X			X			*Model*
36	Brusoni and Prencipe 1999	2	X		X		X	X		X	X			Aero Engines
37	Chesbrough and Kusunoki 1999		(X)		X			X		X				Read-Write Heads for Disc Drives
38	Collier 1982		(X)		X	X		X			X			*Model*
39	Desai et al. 2001	1-2			X	X		X	X		X			*Model*
40	Ernst and Kamrad 2000	1			X	X		X	X		X			*Model*
41	Feitzinger and Lee 1997	1-2			X		X	X	X		X			Printer, PCs

Reference		**Systems Perspective**		**Hierarchy Perspective**						**Life Cycle Perspective**				**Industry / Product Example**
#	Author(s) Year	Description/ Variations of Elements	Description/ Variations of Relations	Techn. Modu- larity	Busin. Modu- larity	C	M	P	F	Des.	Prod.	Use	Retir.	
42	Fisher et al. 1999	1	(X)		X	X		X	X		X			*Model* & Automotive Brakes
43	Garud and Kumaraswamy 1995	1-2	X		X	X		X		(X)	X			*Concept*
44	Gulati and Eppinger 1996	1-2	(X)	n/a[151]	n/a		X	X		X	(X)			Automotive Control Panel
45	Henderson and Clark 1990	1-2	(X)		X		X	X		(X)		X		Photographic Alignment Equipment
46	Hyer and Wemmerlov 1984	1-2		(X)		X	X			X	X			Elevator; Agricultural machinery
47	Langlois and Robertson 1992	1-2			X		X	X		X	(X)	X		Hi-Fi Stereo equipment; Microcomputer
48	Lee and Tang 1998	1			X			X			X			Sweater
49	MacDuffie et al. 1996	1-2	(X)		X	X		X	X		X			Automobile assembly

---

[151] Exploratory paper that applies two views: from the product architecture on the organization, and from the organization on the product architecture.

Reference		Systems Perspective		Hierarchy Perspective						Life Cycle Perspective				Industry / Product Example
#	Author(s) Year	Description/ Variations of Elements	Description/ Variations of Relations	Techn. Modu-larity	Busin. Modu-larity	C	M	P	F	Des.	Prod.	Use	Retir.	
50	Meyer and Lehnerd 1997	2-3	(X)		X	X		X	X	X	X			Electric Iron
51	O'Grady 1999	2	(X)		X	X	X	X		(X)	X			Computer Appliance
52	Pine 1993	1-2	(X)		X	X	X	X		(X)	(X)	X		Lighting controls
53	Pimmler and Eppinger 1994	2	X	X		X	X	X				X		Automotive climate control system
54	Randall and Ulrich 2001	1			X	X		X			X			Bicycle
55	Robertson and Ulrich 1998	1-2	(X)		X	X	X		X	X	X	X		Automotive Instrument Panel
56	Sako and Murray 1999	2	X		X		X	X		X	X	X		Automobile
57	Sanchez and Mahoney 1996	2	(X)		X		X	X		X	(X)	(X)		*Concept*
58	Schilling 2000	1-2	(X)		X		X	X		(X)	(X)	(X)		*Concept*
59	Starr 1965	1-2	(X)		X		X	X	X	(X)	(X)			*Concept*

Reference		Systems Perspective		Hierarchy Perspective						Life Cycle Perspective				Industry / Product Example
#	Author(s) Year	Description/ Variations of Elements	Description/ Variations of Relations	Techn. Modu-larity	Busin. Modu-larity	C	M	P	F	Des.	Prod.	Use	Retir.	
60	Thonemann and Brandeau 2000	1	(X)		X	X		X	X		X			Automotive Wiring Harness
61	Ulrich 1995	3	X	X	(X)	X	X	X		(X)	X	X		*Concept;* Trailer
62	von Hippel 1990	2-3			X		X	X		X				*Concept*

Legend:

Systems Perspective:

Description/Variation of Elements/Modules: 1 = Module' boundaries are fixed, performance scaling only, 2 = Sub-module components can be arranged and re-arranged, 3 = Function allocation across components is wholly variable

Description/Variation of Relations/Interfaces: X = major focus of the work, (X) = implicitly considered in the work

Hierarchy Perspective:

C = component, M = Module, P = Product, F = Product Family

Life cycle Perspective:

Des. = Design Phase, Prod. = Production Phase, Use = Use Phase, Retir. = Retirement Phase

X = major focus of the work, (X) = implicitly considered in the work

Table 9-3: Coding list to separate Table 9-1 and Table 9-2

	Engineering		Management	
Journals	CIRP Annals	5	Academy of Management Review	1
	Design Studies	3	Administrative Science Quarterly	1
	IEEE Transactions on Systems, Man, and Cybernetics	1	European Journal of Operational Research	2
	Journal of Mechanical Design	4	Harvard Business Review	3
	Research in Engineering Design	3	Management Science	7
			Operations Research	1
			Research Policy	3
			Sloan Management Review	1
			Strategic Management Journal	2
Conference Papers	ASME Design Engineering Technical Conferences	13		
Working Papers	MIT Research report; Mechanical Engineering	1	Working Paper, SPRU, Sussex University	1
			Working Paper, Harvard Business School	1
			Working Paper, MIT Sloan School	2
			Working Paper, IMVP	1
Books / Book sections	Product Design & Development (Ulrich & Eppinger)	1	Design Rules (Baldwin & Clark)	1
	Section in edited book on Automotive Assembly (Wilhelm)	1	Power of Product Platforms (Meyer & Lehnerd)	1
			The Age of Modularity (O'Grady)	1
			Mass Customization (Pine)	1
TOTAL		32		30

10. APPENDIX B: FUNCTION-COMPONENT MATRICES

Product Architecture Case 1: Conventional Door

Components

	Functions	1 Door Inner Panel	2 Door Outer Panel	3 Reinforcement at Hinge	4 Reinforcement at Latch	5 Reinforcement at Belt	6 Anti-Intrusion Beam	Component count	total functions involved with these components
1	structure	1	1	1	1	1		5	4
2	side impact protection	1					1	2	3
3	aesthetic appearance		1					1	2
4	carry other parts	1						1	3
	Function count	3	2	1	1	1	1		
							9	9	

Figure 10-1: Function-component matrix for conventional door (case 1)

Product Architecture Case 2: Cruciform Door

Components

	Functions	1 Door Frame	2 Cruciform Outer	3 Cruciform Inner	4 Secondary Beam	5 Reinforcement	Component count	total functions involved with these components
1	structure	1	1	1			3	4
2	side impact protection		1	1	1	1	4	3
3	aesthetic appearance	1					1	2
4	carry other parts		1	1			2	3
	Function count	2	3	3	1	1		
						10	10	

Figure 10-2: Function-component matrix for cruciform door (case 2)

Product Architecture Case 3: Cast Frame Door

Functions		Components 1 Door Frame	Components 2 Door Outer	Components 3 Anti Intrusion Beam	Component count	total functions involved with these components
1	structure	1			1	3
2	side impact protection	1		1	2	3
3	aesthetic appearance		1		1	1
4	carry other parts	1			1	3
	Function count	3	1	1		
				5	5	

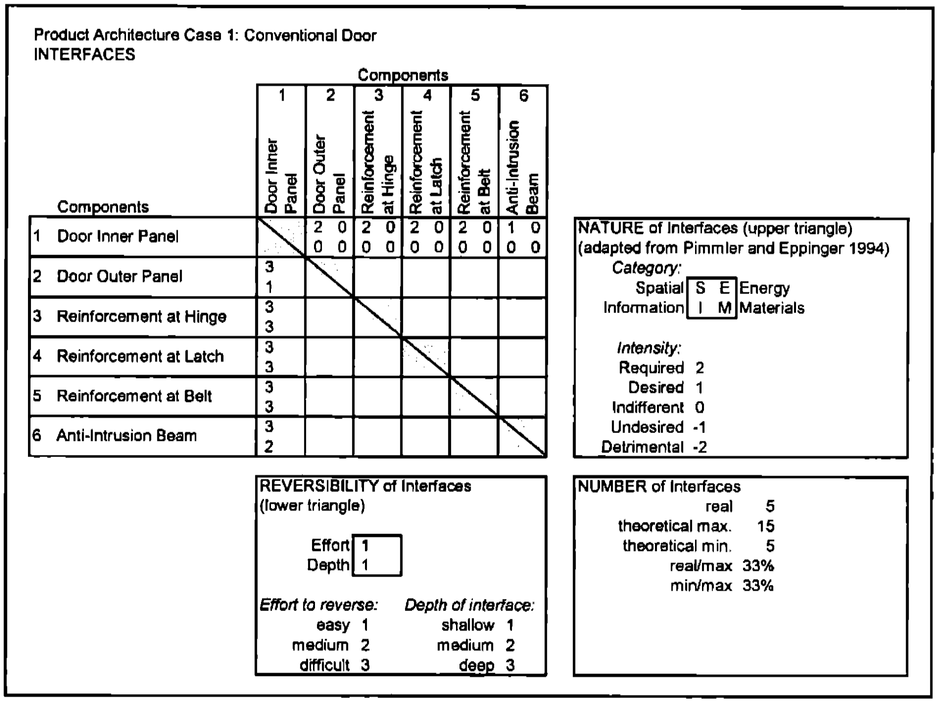
Figure 10-3: Function-component matrix for cast frame door (case 3)

Product Architecture Case 4: Extrusion Frame Door

Functions		Components 1 Door Structure	Components 2 Door Outer	Components 3 Window Frame	Component count	total functions involved with these components
1	structure	1			1	3
2	side impact protection	1			1	3
3	aesthetic appearance		1	1	2	1
4	carry other parts	1			1	3
	Function count	3	1	1		
				5	5	

Figure 10-4: Function-component matrix for extrusion frame door (case 4)

11. APPENDIX C: INTERFACE ASSESSMENTS



Product Architecture Case 1: Conventional Door
INTERFACES

Components	1 Door Inner Panel	2 Door Outer Panel	3 Reinforcement at Hinge	4 Reinforcement at Latch	5 Reinforcement at Belt	6 Anti-Intrusion Beam
1 Door Inner Panel		2 0 0 0	2 0 0 0	2 0 0 0	2 0 0 0	1 0 0 0
2 Door Outer Panel	3 1					
3 Reinforcement at Hinge	3 3					
4 Reinforcement at Latch	3 3					
5 Reinforcement at Belt	3 3					
6 Anti-Intrusion Beam	3 2					

NATURE of Interfaces (upper triangle)
(adapted from Pimmler and Eppinger 1994)
Category:
Spatial S E Energy
Information I M Materials

Intensity:

Required	2
Desired	1
Indifferent	0
Undesired	-1
Detrimental	-2

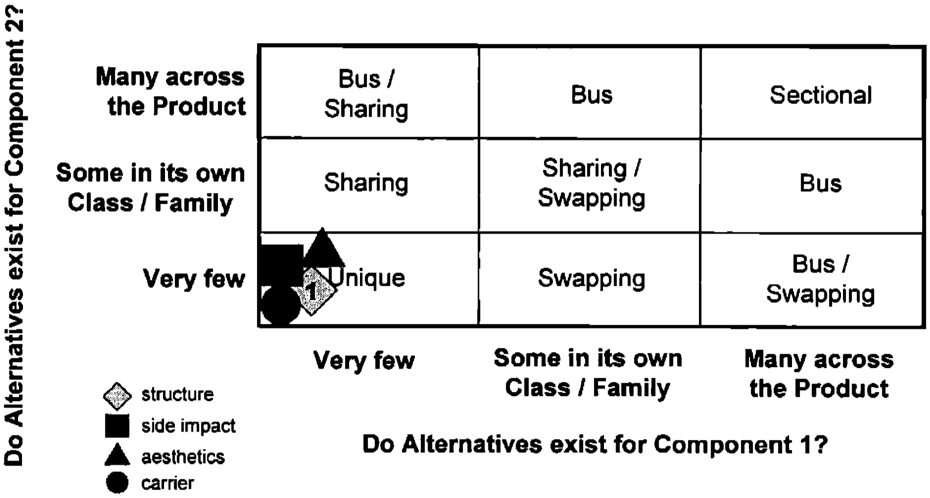
REVERSIBILITY of Interfaces
(lower triangle)
Effort 1
Depth 1

Effort to reverse:		*Depth of interface:*	
easy	1	shallow	1
medium	2	medium	2
difficult	3	deep	3

NUMBER of Interfaces

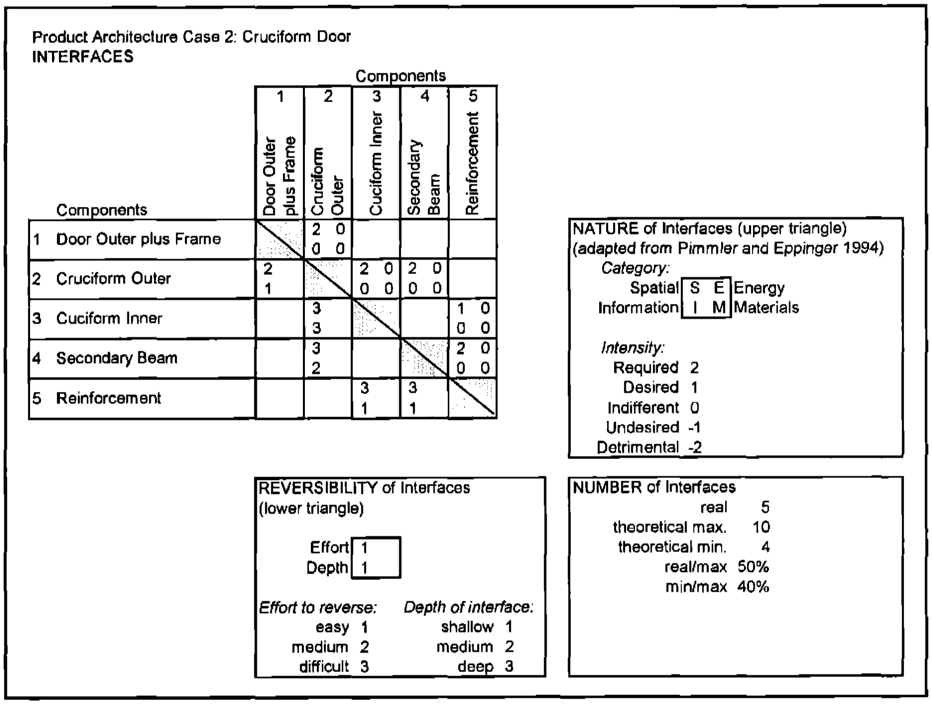
real	5
theoretical max.	15
theoretical min.	5
real/max	33%
min/max	33%

Figure 11-1: Number, nature, and reversibility of the interfaces of the conventional door (case 1)



Do Alternatives exist for Component 2? \ Do Alternatives exist for Component 1?	Very few	Some in its own Class / Family	Many across the Product
Many across the Product	Bus / Sharing	Bus	Sectional
Some in its own Class / Family	Sharing	Sharing / Swapping	Bus
Very few	Unique	Swapping	Bus / Swapping

Figure 11-2: Standardization level of the interfaces of the conventional door (case 1)



Product Architecture Case 2: Cruciform Door
INTERFACES

Components	1 Door Outer plus Frame	2 Cruciform Outer	3 Cuciform Inner	4 Secondary Beam	5 Reinforcement
1 Door Outer plus Frame		2 0 0 0			
2 Cruciform Outer	2 1		2 0 0 0	2 0 0 0	
3 Cuciform Inner		3 3			1 0 0 0
4 Secondary Beam		3 2			2 0 0 0
5 Reinforcement			3 1	3 1	

NATURE of Interfaces (upper triangle)
(adapted from Pimmler and Eppinger 1994)
Category:
Spatial S E Energy
Information I M Materials

Intensity:

Required	2
Desired	1
Indifferent	0
Undesired	-1
Detrimental	-2

REVERSIBILITY of Interfaces
(lower triangle)

Effort 1
Depth 1

Effort to reverse:		*Depth of interface:*	
easy	1	shallow	1
medium	2	medium	2
difficult	3	deep	3

NUMBER of Interfaces

real	5
theoretical max.	10
theoretical min.	4
real/max	50%
min/max	40%

Figure 11-3: Number, nature, and reversibility of the interfaces of the cruciform door (case 2)

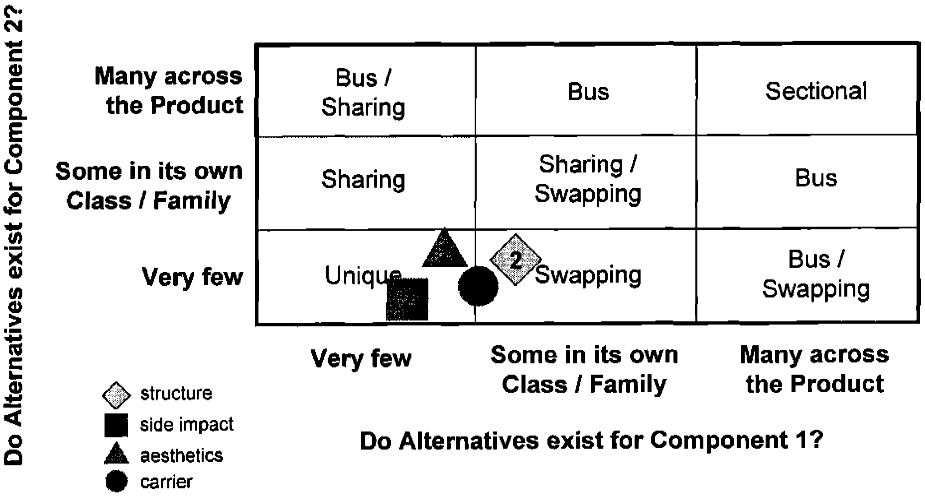
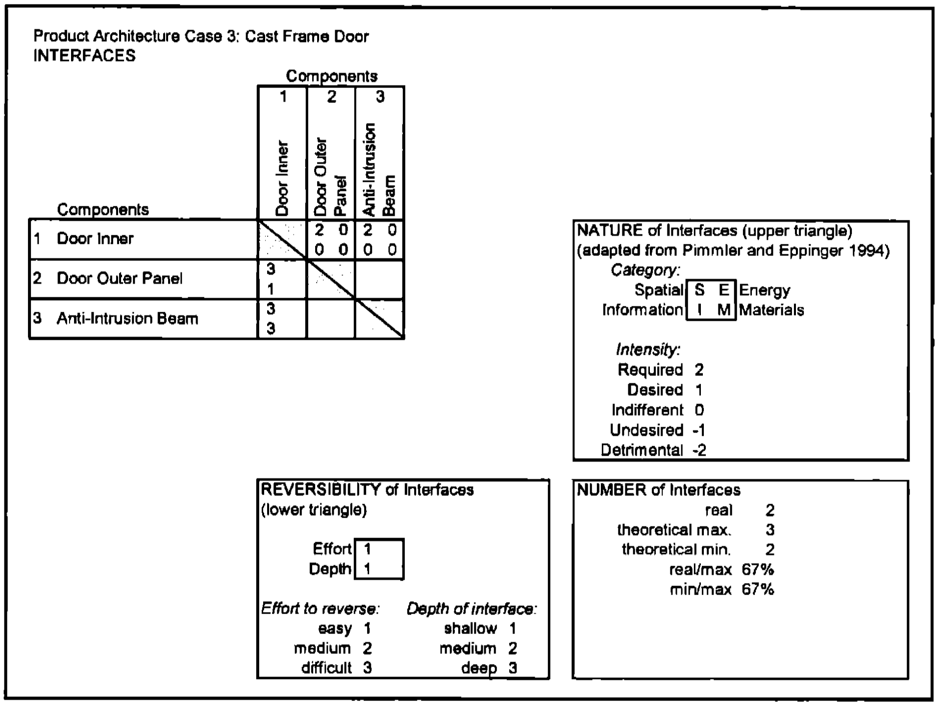


Figure 11-4: Standardization level of the interfaces of the cruciform door (case 2)



Product Architecture Case 3: Cast Frame Door
INTERFACES

Components	1 Door Inner	2 Door Outer Panel	3 Anti-Intrusion Beam
1 Door Inner		2 0 0 0	2 0 0 0
2 Door Outer Panel	3 1		
3 Anti-Intrusion Beam	3 3		

NATURE of Interfaces (upper triangle)
(adapted from Pimmler and Eppinger 1994)

Category:

Spatial	S	E	Energy
Information	I	M	Materials

Intensity:

Required	2
Desired	1
Indifferent	0
Undesired	-1
Detrimental	-2

REVERSIBILITY of Interfaces
(lower triangle)

Effort	1
Depth	1

Effort to reverse:		*Depth of interface:*	
easy	1	shallow	1
medium	2	medium	2
difficult	3	deep	3

NUMBER of Interfaces

real	2
theoretical max.	3
theoretical min.	2
real/max	67%
min/max	67%

Figure 11-5: Number, nature, and reversibility of the interfaces of the cast frame door (case 3)

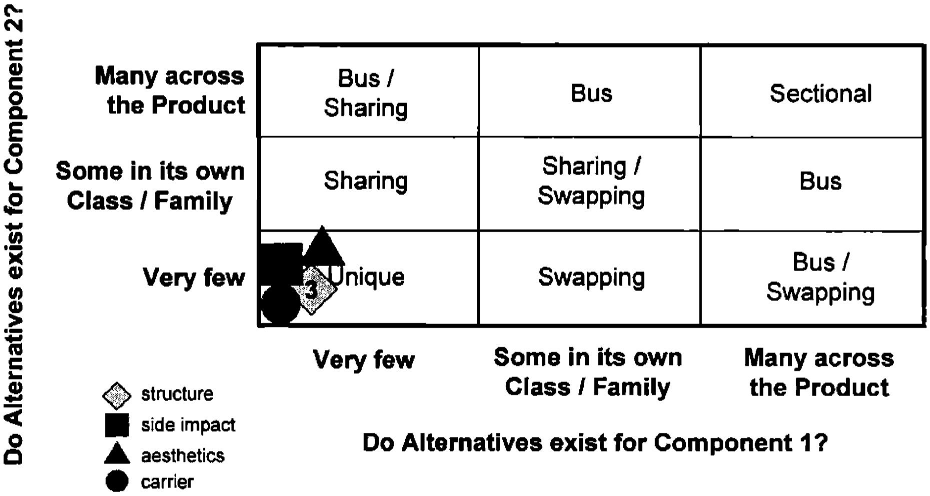
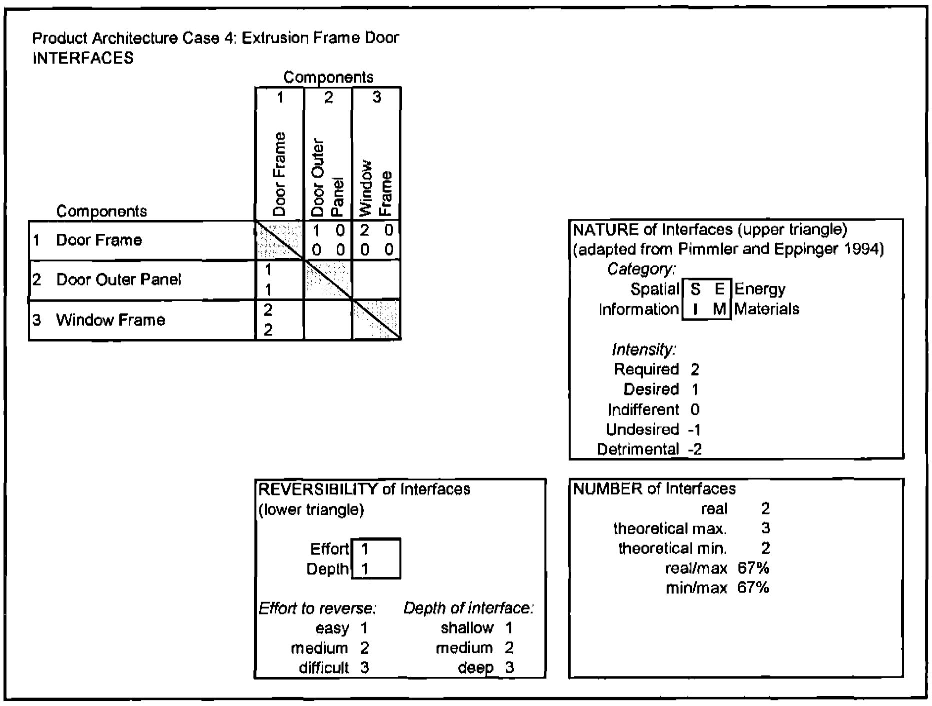


Figure 11-6: Standardization level of the interfaces of the conventional door (case 1)



Product Architecture Case 4: Extrusion Frame Door
INTERFACES

Components	1 Door Frame	2 Door Outer Panel	3 Window Frame
1 Door Frame		1 0 0 0	2 0 0 0
2 Door Outer Panel	1 1		
3 Window Frame	2 2		

NATURE of Interfaces (upper triangle)
(adapted from Pimmler and Eppinger 1994)
Category:
Spatial S E Energy
Information I M Materials

Intensity:
Required 2
Desired 1
Indifferent 0
Undesired -1
Detrimental -2

REVERSIBILITY of Interfaces
(lower triangle)

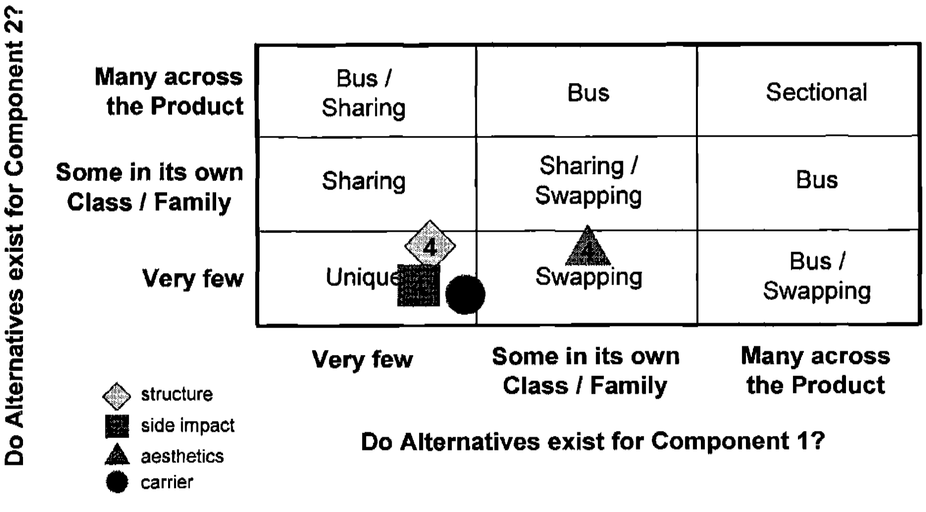
Effort 1
Depth 1

Effort to reverse:		*Depth of interface:*	
easy	1	shallow	1
medium	2	medium	2
difficult	3	deep	3

NUMBER of Interfaces

real	2
theoretical max.	3
theoretical min.	2
real/max	67%
min/max	67%

Figure 11-7: Number, nature, and reversibility of the interfaces of extrusion frame door (case 4)



Do Alternatives exist for Component 2?	Very few	Some in its own Class / Family	Many across the Product
Many across the Product	Bus / Sharing	Bus	Sectional
Some in its own Class / Family	Sharing	Sharing / Swapping	Bus
Very few	Unique	Swapping	Bus / Swapping

Do Alternatives exist for Component 1?

Figure 11-8: Standardization level of the interfaces of the extrusion frame door (case 4)

12. REFERENCES

Agrawal, Mani, T.V. Kumaresh and Glenn A. Mercer (2001). "The false promise of mass customization." McKinsey Quarterly(3).

Ahmadi, Reza, Thomas A. Roemer and Robert H. Wang (2001). "Structuring Product Development Processes." European Journal of Operational Research **130**: 539-558.

Akula, John L. (2000). "Business Crime: What to do when the law pursues you." Sloan Management Review(Spring): 29-41.

Alexander, Christopher (1964). Notes on the Synthesis of Form. Cambridge, Massachusetts, Harvard University Press.

Allen, Kevin R. and Susan Carlson-Skalak (1998). Defining Product Architecture during Conceptual Design. ASME Design Engineering Technical Conference, Atlanta, GA, ASME.

Almgren, Henrik (2000). "Pilot production and manufacturing start-up: the case of Volvo S80." International Journal of Production Research **38**(17): 4577-4588.

Al-Tabtabai, Hashem, Alex P. Alex and Maha Tantash (1999). "Preliminary Cost Estimation of Highway Construction Using Neural Networks." Cost Engineering **41**(3): 19-24.

Anderson, Shannon W. (1995). "Measuring the Impact of Product Mix Heterogeneity on Manufacturing Overhead Cost." The Accounting Review **70**(3): 363-387.

Anderson, Shannon W. and Karen L. Sedatole (1998). "Designing Quality into Products: The Use of Accounting Data in New Product Development." Accounting Horizons **12**(3): 213-233.

Andrade, M.C., R.C. Pessanha Filho, A.M. Espozel, L.O.A. Maia and R.Y. Qassim (1999). "Activity-based costing for production learning." International Journal of Production Economics **62**: 175-180.

Anonymous (1993b). Design of a magnesium/aluminum door frame. Automotive Engineering International. **101:** 57-60.

Anonymous (1997a). High-strain-rate tensile testing of door trim materials. Automotive Engineering International. **105:** 77-79.

Anonymous (1998b). Testing new composite side door concepts. Automotive Engineering International. **106:** 43-48.

Anonymous (1999c). Lear-UTA Deal Redefines Interior Module. WARD's Auto World. **35:** 25-26.

Anonymous (1999d). Meritor showcases roof module. WARD's Auto World. **35:** 50.

Anonymous (1999e). Bosch to Supply Cockpit Modules. WARD's Auto World. **35:** 22.

Anonymous (2000b). DaimlerChrysler launches Internet programme to speed up product development. Automotive World,

Anonymous (2000c). "Plastics Push Modular Market." WARD's Auto World **36**(9): 60-61.

Anonymous (2001a). Wave goodbye to the family car. The Economist: January 13, 2001.

Anonymous (2001b). Automotive modular developments. Automotive Engineering International. **109:** 79-85.

Asiedu, Y. and P. Gu (1998). "Product life cycle cost analysis: state of the art review." International Journal of Production Research **36**(4): 883-908.

Babad, Yair M. and Bala V. Balachandran (1993). "Cost Driver Optimization in Activity-Based Costing." The Accounting Review **68**(3): 563-575.

Baker, Kenneth R., Michael J. Magazine and Henry L.W. Nuttle (1986). "The Effect of Commonality on Safety Stock in a Simple Inventory Model." Management Science **32**(8): 983-988.

Baldwin, Carliss Y. and Kim B. Clark (1997). "Managing in the Age of Modularity." Harvard Business Review **75**(September-October): 84-93.

Baldwin, Carliss Y. and Kim B. Clark (2000). Design Rules. Volume 1: The Power of Modularity. Cambridge, Massachusetts, MIT Press.

Banker, Rajiv D., Srikant M. Datar and Sunder Kekre (1988). "Relevant Costs, Congestion and Stochasticity in Production Environments." Journal of Accounting and Economics **10**: 171-197.

Banker, Rajiv D., Srikant M. Datar, Sunder Kekre and Tridas Mukhopadhyay (1990). Cost of Product and Process Complexity. Measures for Manufacturing Excellence. R. S. Kaplan. Boston, Massachusetts, Harvard Business School Press: 269-290.

Banker, Rajiv D., Gordon Potter and Roger G. Schroeder (1995). "An empirical analysis of manufacturing overhead cost drivers." Journal of Accounting and Economics **19**: 115-137.

Ben-Arieh, D. (2000). "Cost estimation system for machined parts." International Journal of Production Research **38**(17): 4481-4494.

Bhimani, Al and Philip S. Muelder (2001). "Managing Processes, Quality, and Costs: A Case Study." Journal of Cost Management **15**(2): 28-32.

Bielefeld, James R. Jr. and G. David Rucklos (1992). "Cost Scaling Factors: How accurate are they?" Cost Engineering **34**(10): 15-20.

Biersdorfer, J.D. (2001). Module Lineup Gives A Little Organizer Big Ambitions. The New York Times, New York, NY, April 5, 2001, E7.

Birkholz, Eberhard and Rainer G. Stark (1997). Engineering Of Modular Door Systems - Evaluation Of Different Concepts From The Perspective Of Vehicle Manufacturers. International Body Engineering Conference, Detroit, Michigan, Society of Automotive Engineers.

Blackenfelt, Michael (2000). Profit Maximization While Considering Uncertainty by Balancing Commonality and Variety Using Robust Design - The Redesign of a Family of Lift Tables. 2000 ASME Design Engineering Technical Conferences, Baltimore, Maryland, ASME.

Blanchard, Benjamin S. and Wolter J. Fabrycky (1998). Systems Engineering and Analysis. Upper Saddle River, NJ, Prentice-Hall.

Blischke, Wallace R. and D.N.Prabhakar Murthy (1994). Warranty Cost Analysis. New York, Marcel Dekker, Inc.

Blocher, Edward and William L. Berry (1998). "Cost Management with a Strategic Emphasis: Selected Manufacturing Cases." Journal of Cost Management **12**(6): 6-11.

Bode, Juergen (2000). "Neural networks for cost estimation: simulations and pilot application." International Journal of Production Research **38**(6): 1231-1254.

Boer, Michael and Rasaratnam Logendran (1999). "A methodology for quantifying the effects of product development on cost and time." IIE Transactions **31**(4): 365-378.

Boothroyd, Geoffrey, Peter Dewhurst and Winston A. Knight (1994). Product design for manufacture and assembly. New York, Marcel Dekker.

Bradley, Stephen, Matthias Calice, Marc Fischer and Michael Harmening (1997). Design for Purchasing: A Methodology for Reducing the Materials Cost of Engineered Products. ASME Design Engineering Technical Conferences, Sacramento, CA, ASME.

Bradsher, Keith (2000b). Ford reduces Output to Help Recall of Tires. The New York Times, New York, August 22, 2000, C1.

Braha, Dan and Oded Maimon (1998). "The Measurement of a Design - Structural and Functional Complexity." IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans **28**(4): 527-535.

Brimson, James A. (1998). "Feature Costing: Beyond ABC." Journal of Cost Management **12**(1): 6-12.

Brooke, Lindsay (2000a). Head Games - Eaton Corp. and TRW are battling for a new chunk of powertrain supply profits - the fully assembled modular cylinder head. Automotive Industries: 40-41.

Brown, Susan (2001). Build-to-Order. Automotive World, http://www.just-auto.com/features_detail.asp?art=412

Browning, Tyson R. (1998). Modeling and Analyzing Cost, Schedule, and Performance in Complex System Product Development. Doctoral Thesis. Technology, Management and Policy, Massachusetts Institute of Technology, Cambridge, MA: 299.

Brusoni, Stefano and Andrea Prencipe (1999). Modularity in Complex Product Systems: Managing the Knowledge Dimension. Working Paper, SPRU, Sussex University, : .

Buchholz, Kami (1998a). "Pre-assembled door system has it all." Automotive Engineering International **106**(8): 43-44.

Buchholz, Kami (2000). Roof modules: present and future. Automotive Engineering International. **108:** 80.

Buede, Dennis M. (2000). The Engineering Design of Systems - Models and Methods. New York, John Wiley & Sons.

Burman, Deepak (1998). "The Design to Cost (DTC) Approach to Product Development." AACE International Transactions(VE.03): 1-3.

Busch, John V. (1987). Technical Cost Modeling of Plastic Fabrication Processes. Doctoral Thesis. Materials Science and Engineering, Massachusetts Institute of Technology, Cambridge, MA: 164.

Chan, D.S.K. and W.P. Lewis (2000). "The integration of manufacturing and cost information into the engineering design process." International Journal of Production Research **38**(17): 4413-4427.

Chen, Chun-Hsien, L.G. Occena and Sai Cheong Fok (2001). "CONDENSE: a concurrent design evaluation system for product design." International Journal of Production Research **39**(3): 413-433.

Chesbrough, Henry W. and Ken Kusunoki (1999). The Modularity Trap: Innovation, Technology Phase-Shifts, and Resulting Limits of Virtual Organizations. Working Paper, Harvard Business School, February 1999: 49.

Christensen, Clayton M. (1992a). "Exploring the limits of the technology S-curve. Part I: Component Technologies." Production and Operations Management **1**(4): 334-357.

Christensen, Clayton M. (1997). The Innovator's Dilemma - When New Technologies cause Great Firms to Fail. Boston, Massachusetts, Harvard Business School Press.

Chun, Young H. and Kwei Tang (1999). "Cost Analysis of Two-Attribute Warranty Policies Based on the Product Usage Rate." IEEE Transactions on Engineering Management **46**(2): 201-209.

Clancy, Donald K. (1998). "Strategic Design Cost Analysis." Cost Engineering **40**(8): 25-30.

Clark, Joel P., Rich Roth and Frank R. Field (1997). Techno-Economic Issues in Materials Selection. ASM Handbook. G. E. Dieter. Materials Park, OH, ASM International. **20 Materials Selection and Design:** 255-265.

Clark, Kim B. (1985). "The interaction of design hierarchies and market concepts in technological evolution." Research Policy **14**: 235-251.

Clark, Kim B. and Takahiro Fujimoto (1991). Product Development Performance. Boston, Massachusetts, Harvard Business School Press.

Cokins, Gary (2000). "The Changing Face of Cost Management in the Auto Industry." Journal of Cost Management **14**(5): 13-15.

Collier, David A. (1982). "Aggregate Safety Stock Levels and Component Part Commonality." Management Science **28**(11): 1297-1303.

Cooper, Robin and W. Bruce Chew (1996). "Control tomorrow's costs through today's designs." Harvard Business Review **74**(1): 88-97.

Cooper, Robin and Robert S. Kaplan (1988). "How Cost Accounting Distorts Product Costs." Management Accounting(4): 20-27.

Cooper, Robin and Robert S. Kaplan (1992). "Activity-Based Systems: Measuring the Costs of Resource Usage." Accounting Horizons(9): 1-12.

Cooper, Robin and Regine Slagmulder (1997). Target Costing and Value Engineering. Portland, Oregon, Productivity Press.

Coulter, Stewart L., Mark W. McIntosh, Bert Bras and David W. Rosen (1998). Identification of Limiting Factors for Improving Design Modularity. ASME Design Engineering Technical Conferences, Atlanta, GA, ASME.

Crosse, Jesse (1999). Leading Lights. Automotive Engineer: 63-64.

Cunningham, Timothy W. and Daniel E. Whitney (1998). The Chain Method for Identifying Integration Risk During Concept Design. ASME Design Engineering Technical Conferences, Atlanta, GA, ASME.

Dahmus, Jeffrey B., Javier P. Gonzalez-Zugasti and Kevin N. Otto (2001). "Modular Product Architecture." Design Studies **22**(5): 409-424.

Dahmus, Jeffrey B. and Kevin N. Otto (2001). Incorporating Lifecycle Costs into Product Architecture Decisions. Research report, MIT, : 12.

Darlington, John (1999). "Lean thinking and mass customisation: The relationship between production and costs." Management Accounting **77**(10): 18-21.

Das, Sanchoy K., Pradeep Yedlarajiah and Raj Narendra (2000). "An approach for estimating the end-of-life product disassembly effort and cost." International Journal of Production Research **38**(3): 657-673.

Datar, Srikant M., Sunder Kekre, Tridas Mukhopadhyay and Kannan Srinivasan (1993). "Simultaneous Estimation of Cost Drivers." The Accounting Review **68**(3): 602-614.

Datar, Srikant M., Sunder Kekre, Tridas Mukhopadhyay and Eric Svaan (1991). "Overloaded Overheads: Activity-Based Cost Analysis of Material Handling in Cell Manufacturing." Journal of Operations Management **10**(1): 119-137.

de Neufville, Richard (1990). Applied Systems Analysis - Engineering Planning and Technology Management. New York, McGraw-Hill.

Delphi (1999). Form 10K (Annual Report). Delphi Automotive Systems, http://www.sec.gov/Archives/edgar/data/1072342/0001072342-99-000006.txt

Desai, Preyas, Sunder Kekre, Suresh Radhakrishnan and Kannan Srinivasan (2001). "Product Differentiation and Commonality in Design: Balancing Revenues and Cost Drivers." Management Science **47**(1): 37-51.

Dewhurst, Peter (2000). Evaluation of Instrument Panel Designs for Cost of Manufacture and Environmental Impact. , Boothroyd Dewhurst Inc., : .

Doran, David T. and Joe E. Dowd (1999). "Depreciation and Amortization Cost in Activity Based Costing Systems." Journal of Cost Management **13**(5): 34-38.

Du, Xuehong, Mitchell M. Tseng and Jianxin Jiao (2000). Graph Grammar Based Product Variety Modeling. 2000 ASME Design Engineering Technical Conferences, Baltimore, Maryland, ASME.

Duray, Rebecca, Peter T. Ward, Glenn W. Milligan and William L. Berry (2000). "Approaches to mass customization: configurations and empirical validation." Journal of Operations Management **18**: 605-625.

Eisenhardt, Kathleen M. (1989). "Building Theories from Case Study Research." Academy of Management Review **14**(4): 532-550.

Ellis, Michael (2001). Opel drops GM's maligned "brand management" plan. Reuters, http://just-auto.com/news_print.asp?art=24810

Emblemsvag, Jan and Bert Bras (1994). Activity-based costing in design for product retirement. Advances in Design Automation, ASME.

Eppinger, Steven D., Daniel E. Whitney, Robert P. Smith and David A. Gebala (1994). "A Model-Based Method for Organizing Tasks in Product Development." Research in Engineering Design(6): 1-13.

Erixon, Gunnar, Alex von Yxkull and Anders Arnstroem (1996). "Modularity - the Basis for Product and Factory Reengineering." CIRP Annals **45**(1): 1-6.

Ernst, Ricardo and Bardia Kamrad (2000). "Evaluation of supply chain structures through modularization and postponement." European Journal of Operational Research **124**: 495-510.

Eynan, Amit and Meir J. Rosenblatt (1996). "Component commonality effects on inventory costs." IIE Transactions **28**: 93-104.

Fargher, Neil and Dale Morse (1998). "Quality Costs: Planning the Trade-off between Prevention and Appraisal Activities." Journal of Cost Management **12**(1): 14-22.

Feitzinger, Edward and Hau L. Lee (1997). "Mass Customization at Hewlett-Packard: The Power of Postponement." Harvard Business Review **75**(January-February): 116-121.

Fine, Charles H. (1998). Clockspeed - Winning Industry Control in the Age of Temporary Advantage. Reading, Massachusetts, Perseus Books.

Finger, Susan and John R. Dixon (1989a). "A Review of Research in Mechanical Engineering Design. Part I: Descriptive, Prescriptive, and Computer-Based Models of Design Processes." Research in Engineering Design **1**(1): 51-67.

Finger, Susan and John R. Dixon (1989b). "A Review of Research in Mechanical Engineering Design. Part II: Representations, Analysis, and Design for the Life Cycle." Research in Engineering Design **1**(2): 121-137.

Fisher, Marshall L. and Christopher D. Ittner (1999). "The Impact of Product Variety on Automobile Assembly Operations: Empirical Evidence and Simulation Analysis." Management Science **45**(6): 771-786.

Fisher, Marshall L., Kamalini Ramdas and Karl T. Ulrich (1999). "Component Sharing in the Management of Product Variety: A Study of Automotive Braking Systems." Management Science **45**(3): 297-315.

Fixson, Sebastian K. and Mari Sako (2001). Modularity in Product Architecture: Will the Auto Industry Follow the Computer Industry? (An Analysis of Product Architecture, Market Conditions, and Institutional Forces). IMVP Working Paper, International Motor Vehicle Program, September 10, 2001: 66.

Foster, George and Mahendra Gupta (1990). "Manufacturing Overhead Cost Driver Analysis." Journal of Accounting and Economics **12**: 309-337.

Fritzsch, Ralph B. (1997). "Activity-Based Costing and the Theory of Constraints: Using Time Horizons to Resolve Two Alternative Concepts of Product Cost." Journal of Applied Business Research **14**(1): 83-89.

Galsworth, Gwendolyn D. (1994). Smart, Simple Design. Essex Junction, Vermont, Oliver Wight Publications.

Garud, Raghu and Arun Kumaraswamy (1995). "Technological and Organizational Designs for Realizing Economies of Substitution." Strategic Management Journal **16**(Summer): 93-109.

Gerchak, Yigal, Michael J. Magazine and A. Bruce Gamble (1988). "Component Commonality with Service Level Requirements." Management Science **34**(6): 753-760.

Goepfert, Jan (1998). Modulare Produktentwicklung - Komplexitaetsbewaeltigung durch die gemeinsame Modularisierung von Produkt und Entwicklungsorganisation. Innovationsforschung and Technologiemanagement. N. Franke and C.-F. von Braun. Berlin, Springer Verlag: 139-151.

Goldratt, Eliyahu M. and Jeff Cox (1984). The Goal - A Process of ongoing Improvement. Great Barrington, MA, North River Press.

Gonzalez-Zugasti, Javier P., Kevin N. Otto and John D. Baker (2000). "A Method for Architecting Product Platforms." Research in Engineering Design **12**: 61-72.

Greenwood, Allen G. (1997). An Approach to Enhance Cost Estimation During Product/Process Design. Annual Meeting of the Decision Sciences Institute, Atlanta, GA, Decision Sciences Institute.

Gu, P., M. Hashemian and S. Sosale (1997). "An Integrated Modular Design Methodology for Life-Cycle Engineering." CIRP Annals **46**(1): 71-74.

Gulati, Rosaline K. and Steven D. Eppinger (1996). The Coupling of Product Architecture and Organizational Structure Decisions. Working Paper #3906-96, MIT Sloan School of Management, May 1996: 31.

Han, Helen N. (1994). The Competitive Position of Alternative Automotive Materials. Doctoral Thesis. Materials Science and Engineering, Massachusetts Institute of Technology, Cambridge, MA: 131.

Henderson, Rebecca M. and Kim B. Clark (1990). "Architectural Innovation: The Reconfiguration of Existing Product Technologies and the Failure of Established Firms." Administrative Science Quarterly **35**: 9-30.

Hervey, Richard P. (2000). Automotive Modularity: Issues for Automakers, Suppliers, Labor, and Capital Markets. Report combining 3 Presentations, Sigma Associates, Spring 2000: 34.

Hoff, Todd, John Madej, Tom Goral, Chuck Ryntz, Ana Ludvill, Bob Cicala and Steve Hartig (1998). Design and Development of a Generic Door Hardware Module Concept. SAE Paper 980999, Society of Automotive Engineers, 1998: 13.

Holweg, Matthias and Frits K. Pil (2001). "Seeing the Whole - Reconnecting to the Customer through Build-to-Order." Sloan Management Review **43**(1): 74-.

Horngren, Charles T. and George Foster (1991). Cost accounting: a managerial emphasis. Englewood Cliffs, N.J., Prentice Hall.

Hoult, David P. and C. Lawrence Meador (1997). Manufacturing Cost Estimating. ASM Handbook. G. E. Dieter. Materials Park, OH, ASM International. **20 Materials Selection and Design:** 716-722.

Hu, S.J. (1997). "Stream-of-Variation Theory for Automotive Body Assembly." CIRP Annals **46**(1): 1-6.

Hu, Weiyi and Corrado Poli (1997a). To injection mold, to stamp, or to assemble? - Part I: A DFM cost perspective. ASME Design Engineering Technical Conferences, Sacramento, CA, ASME.

Hu, Weiyi and Corrado Poli (1997b). To injection mold, to stamp, or to assemble? - Part II: A Time-to-Market perspective. ASME Design Engineering Technical Conferences, Sacramento, CA, ASME.

Huang, Chung-Che and Andrew Kusiak (1998). "Modularity in Design of Products and Systems." IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans **28**(1): 66-77.

Hundal, Mahendra S. (1997). "Product Costing: A Comparison of Conventional and Activity-Based Costing Methods." Journal of Engineering Design **8**(1): 91-103.

Hyde, Justin (2001). What ever happened to Chrysler - Analysis., Reuters, February 26, 2001: 2.

Hyer, Nancy L. and Urban Wemmerlov (1984). "Group technology and productivity." Harvard Business Review **62**(July/August): 3-12.

Ishii, Kosuke, Cheryl Juengel and C. Fritz Eubanks (1995). Design for Product Variety: Key to Product Line Structuring. ASME International Conference on Design Theory and Methodology, ASME.

Ittner, Christopher D. and David F. Larcker (1997). "Product Development Cycle Time and Organizational Performance." Journal of Marketing Research **34**(1): 13-23.

Ittner, Christopher D. and John Paul MacDuffie (1994). Exploring the sources of international differences in manufacturing overhead. IMVP Research Paper, The Wharton School, University of Pennsylvania, June 1994: 53.

Jasaitis Ennis, Kristina, John J. Dougherty, Thomas Lamb, Charles R. Greenwell and Richard Zimmermann (1998). "Product-Oriented Design and Construction Cost Model." Journal of Ship Production **14**(1): 41-58.

Jiao, Jianxin and Mitchell M. Tseng (1999). "An Information Modeling Framework for Product Families to Support Mass Customization Manufacturing." CIRP Annals **48**(1): 93-98.

Joalto Design (2000). Joalto Design Safety Systems. Southfield, MI, Joalto Design, 20990 Bridge Street, Southfield, Michigan 48034.

Johannesson, Hans L. (1996). On the Nature and Consequences of Functional Couplings in Axiomatic Machine Design. ASME Design Engineering Technical Conferences, Irvine, CA, ASME.

Johannesson, Hans L. (1997). Application of Interaction and Functional Coupling Rules in Configuration Design. ASME Design Engineering Technical Conferences, Sacramento, CA, ASME.

Johnson, Richard F. (1997). Using DFA to enhance value engineering / value analysis workshop outcomes. 12th Annual International Forum on DFMA, Newport, RI.

Jost, Kevin (1995). "Next-generation modular door system." Automotive Engineering International **1995**(May): 59-61.

Kang, Paul J. (1998). A Technical and Economic Analysis of Structural Composite Use in Automotive Body-in-White Applications. Master Thesis. Materials Science and Engineering, Massachusetts Institute of Technology, Cambridge, MA: 170.

Kaplan, Robert S. (1991). "New Systems for Measurement and Control." The Engineering Economist **36**(3): 201-218.

Kaplan, Robert S. and Robin Cooper (1998). Cost and effect: using integrated cost systems to drive profitability and performance. Boston, Massachusetts, Harvard Business School Press.

Kee, Robert (1998). "Integrating ABC and the Theory of Constraints to Evaluate Outsourcing Decisions." Journal of Cost Management **12**(1): 24-36.

Kee, Robert and Charles Schmidt (2000). "A comparative analysis of utilizing activity-based costing and the therory of constraints for making product-mix decisions." International Journal of Production Economics **63**: 1-17.

Kimberley, William (1999). Back to the future. Automotive Engineer: 62-64.

Kinutani, Hiroshi (1997). Modular Assembly in Mixed-Model Production at Mazda. Transforming Automobile Assembly - Experience in Automation and Work Organization. K. Shimokawa, U. Juergens and T. Fujimoto. Berlin, Springer: 95-108.

Kirchain, Randolph E. (1999). Modeling Methods for Complex Manufacturing Systems: Studying the Effects of Materials Substitution on the Automobile Recycling Infrastructure. Doctoral Thesis. Materials Science and Engineering, Massachusetts Institute of Technology, Cambridge, MA: 241.

Kirchain, Randolph E. (2001). "Cost Modeling of Materials and Manufacturing Processes." Encyclopedia of Materials: Science and Technology: 1718-1727.

Kirschman, C.F. and G.M. Fadel (1998). "Classifying Functions for Mechanical Design." Journal of Mechanical Design **120**: 475-482.

Knight, Winston A. (1998). Group Technology, Concurrent Engineering and Design for Manufacture and Assembly. Group Technology and Cellular Manufacturing: A State-of-the-Art Synthesis of Research and Practice. N. C. Suresh and J. M. Kay. Boston, Kluwer Academic Publishers: 15-36.

Kochan, Anna (2001). The future modularity of doors. Automotive World,

Koltai, Tamas, Sebastian Lonzano, Fernando Guerrero and Luis Onieva (2000). "A flexible costing system for flexible manufacturing systems using activity based costing." International Journal of Production Research **38**(7): 1615-1630.

Kota, Sridhar and Kannan Sethuraman (1998). Managing Variety in Product Families through Design for Commonality. ASME Design Engineering Techncial Conferences, Atlanta, GA, ASME.

Kota, Sridhar, Kannan Sethuraman and Raymond Miller (2000). "A Metric for Evaluating Design Commonality in Product Families." Journal of Mechanical Design **122**(4): 403-410.

Krishnan, Viswanathan, Steven D. Eppinger and Daniel E. Whitney (1997). "A Model-Based Framework to Overlap Product Development Activities." Management Science **43**(4): 437-451.

Krishnan, Viswanathan and Karl T. Ulrich (2001). "Product Development Decisions: A Review of the Literature." Management Science **47**(1): 1-21.

Kusiak, Andrew (1999). Engineering Design: Products, Processes, and Systems. San Diego, Academic Press.

Langlois, Richard N. and Paul L. Robertson (1992). "Network and innovation in a modular system: Lessons from the microcomputer and stereo component industries." Research Policy **21**: 297-313.

Lee, Hau L. and Christopher S. Tang (1998). "Variability Reduction Through Operations Reversal." Management Science **44**(2): 162-172.

Lefever, Douglas D. and Kristin L. Wood (1996). Design for Assembly Techniques in Reverse Engineering and Redesign. ASME Design Engineering Technical Conferences, Irvine, CA, ASME.

Leibl, Peter, Mahendra S. Hundal and Guenther Hoehne (1999). "Cost Calculation with a Feature-based CAD System using Modules for Calculation, Comparison and Forecast." Journal of Engineering Design **10**(1): 93-102.

Levy, David L. (1994). "Chaos Theory and Strategy: Theory, Application, and Managerial Implications." Strategic Management Journal **15**: 167-178.

Lieberman, Marvin B., Susan Helper and Lieven Demeester (1999). "The Empirical Determinants of Inventory Levels in High-Volume Manufacturing." Production and Operations Management **8**(1): 44-55.

Liebers, A. and H.J.J. Kals (1997). "Cost Decision Support in Product Design." CIRP Annals **46**(1): 107-112.

Lingnau, Volker (1999). Management Accounting and Product Variety. Optimal Bundling. R. Fuerderer, A. Herrmann and G. Wuebker. Berlin, Springer: 133-155.

Locascio, Angela (1999). "Design Economics for Electronics Assembly." The Engineering Economist **44**(1): 64-77.

MacDuffie, John Paul, Kannan Sethuraman and Marshall L. Fisher (1996). "Product Variety and Manufacturing Performance: Evidence from the International Automotive Assembly Plant Study." Management Science **42**(3): 350-369.

Maier, Mark W. and Eberhardt Rechtin (2000). The Art of Systems Architecting. Boca Raton, FL, CRC Press.

Malhotra, Manoj K. and Varun Grover (1998). "An assessment of survey research in POM: from constructs to theory." Journal of Operations Management **16**: 407-425.

Mapleston, Peter (1999). GE, Delphi renew efforts in next-generation door modules. Modern Plastics. **76**: 27.

Martin, Mark V. (1999a). Design for Variety: A Methodology for Developing Product Platform Architectures. Doctoral Thesis. Mechanical Engineering, Stanford University, Stanford, CA: 172.

Martin, Mark V. and Kosuke Ishii (1996). Design for Variety: A Methodology for Understanding the Costs of Product Proliferation. ASME Design Engineering Technical Conferences, Irvine, CA, ASME.

Martin, Mark V. and Kosuke Ishii (2000). Design for Variety: A Methodology for Developing Product Platform Architectures. 2000 ASME Design Engineering Technical Conferences, Baltimore, MD, ASME.

Martin, Norman (1999b). "Integrating the IP." Automotive Industries(May): 30-35.

McAdams, Daniel A., Robert B. Stone and Kristin L. Wood (1998). Understanding Product Similarity Using Customer Needs. ASME Design Engineering Technical Conferences, Atlanta, GA, ASME.

McAlinden, Sean P., Brett C. Smith and Bernard F. Swiecki (1999). The Future of Modular Automotive Systems: Where are the Economic Efficiencies in the Modular Assembly Concept? Research Memorandum No.1, Office for the Study of Automotive Transportation - University of Michigan Transportation Research Institute, November 1999: 31.

McCutcheon, David M. and Jack R. Meredith (1993). "Conducting case study research in operations management." Journal of Operations Management **11**: 239-256.

Mercer, Glenn (1995). Modular supply in the 1990s: the key to success. Chapter 11 of The Economist Intellligence Unit Limited 1995.

Meredith, Jack R. (1998). "Building operations management theory through case and field research." Journal of Operations Management **16**: 441-454.

Meyer, Marc H. and Alvin P. Lehnerd (1997). The Power of Product Platforms. New York, The Free Press.

Michaels, Jack V. and William P. Woods (1989). Design to Cost. New York, John Wiley & Sons.

Michelena, Nestor and Panos Papalambros (1997). "A hypergraph framework for optimal model-based decomposition of design problems." Computational Optimization and Applications **8**(2): 173-196.

Miller, Jeffrey G. and Thomas E. Vollmann (1985). "The hidden factory." Harvard Business Review **63**(5): 142-150.

Mirapaul, Matthew (2001). Made Especially for You, In Industrial Quantities. The New York Times, New York, NY, March 11, 2001, 11.

Moulin, Jean-Paul, Hugues Cheron, Jean-Louis Vaysse, Pierre Perdoux and Matt Orlando (1999). Modular Vehicle Approach. SAE Paper 1999-01-1255, Society of Automotive Engineers, 1999: 57-61.

Mudge, Arthur E. (1971). Value Engineering - A Systematic Approach. New York, McGraw-Hill.

Murphy, Tom (2000). Jack sets his sights on cockpit modules. WARD's Auto World. **36:** 86.

Murthy, D.N.Prabhakar and Wallace R. Blischke (2000). "Strategic Warranty Management: A Life-Cycle Approach." IEEE Transactions on Engineering Management **47**(1): 40-54.

Nagarur, Nagen and Abdullahil Azeem (1999). "Impact of commonality and flexibility on manufacturing performance: A simulation study." International Journal of Production Economics **60-61**: 125-134.

Nevins, James L. and Daniel E. Whitney (1989). Concurrent Design of Products and Processes: A Strategy for the Next Generation in Manufacturing. New York, McGraw-Hill Publishing Company.

Newcomb, P.J., Bert Bras and David W. Rosen (1998). "Implications of Modularity on Product Design for the Life Cycle." Journal of Mechanical Design **120**(3): 483-491.

NHTSA (2002). Standard No. 214 -- Side Impact Protection. National Highway Traffic Safety Administration, http://www.nhtsa.dot.gov/cars/rules/standards/safstan2.htm

Nobeoka, Kentaro and Michael A. Cusumano (1993). Multi-Project Management: Strategy and Organization in Automobile Product Development. Working Paper, MIT Sloan School of Management, September 11, 1993: 53.

Noreen, Eric and Naomi Soderstrom (1994). "Are overhead costs strictly proportional to activity?" Journal of Accounting and Economics **17**: 255-278.

O'Grady, Peter (1999). The Age of Modularity - Using the new world of modular products to revolutionize your corporation, Adams and Steele Publishing.

Pahl, Gerhard and Wolfgang Beitz (1996). Engineering Design - A Systematic Approach. London, Springer.

Perera, H.S.C., Nagen Nagarur and Mario T. Tabucanon (1999). "Component part standardization: A way to reduce the life-cycle cost of products." International Journal of Production Economics **60-61**: 109-116.

Pimmler, Thomas U. and Steven D. Eppinger (1994). Integration Analysis of Product Decompositions. Working Paper WP# 3690-94-MS, MIT Sloan School of Management, May 1994: 39.

Pine, B. Joseph II (1993). Mass Customization. Boston, Massachusetts.

Porter, Alan L., A. Thomas Roper, Thomas W. Mason, Frederick A. Rossini and Jerry Banks (1991). Forecasting and Management of Technology. New York, John Wiley & Sons.

Pothoven, Alexander (1999). The Smart Body Panel System - Innovative Design of the Exterior Surface of the 'Smart'. Business Briefing: Global Automotive Manufacturing & Technology: 136-142.

Pugh, Stewart (1990). Total Design. Wokingham, England, Addison-Wesley.

Ramdas, Kamalini and Mohanbir S. Sawhney (2001). "A Cross-Functional Approach to Evaluating Multiple Line Extensions for Assembled Products." Management Science **47**(1): 22-36.

Randall, Taylor and Karl T. Ulrich (2001). "Product Variety, Supply Chain Structure, and Firm Performance: Analysis of the U.S. Bicycle Industry." Management Science **47**(12): 1588-1604.

Rehman, Sumaira and Marin D. Guenov (1998). "A Methodology for Modeling Manufacturing Costs at Conceptual Design." Computers & Industrial Engineering **35**(3-4): 623-626.

Reinertsen, Donald G. (1997). Managing the Design Factory. New York, The Free Press.

Robertson, David and Karl T. Ulrich (1998). "Planning for Product Platforms." Sloan Management Review(Summer): 19-31.

Roemer, Thomas A. (2000). "Time-cost trade-offs in overlapped product development." Operations Research **48**(6): 858-865.

Rosa, Jose Antonio, Joseph F. Porac, Jelena Runser-Spanjol and Michael S. Saxon (1999). "Sociocognitive Dynamics in a Product Market." Journal of Marketing **63**(Special Issue 1999): 64-77.

Rosenbloom, Richard S. and Clayton M. Christensen (1994). "Technological Discontinuities, Organizational Capabilities, and Strategic Commitments." Industrial and Corporate Change **3**(3): 655-685.

Sako, Mari and Fiona Murray (1999). Modules in Design, Production and Use: Implications for the Global Auto Industry. Working paper prepared for the IMVP Annual Sponsors Meeting, IMVP, 5-7 October 1999: 32.

Sako, Mari and Max Warburton (1999). Modularization and Outsourcing Project - Preliminary Report of European Research Team. Preliminary Report, International Motor Vehicle Program, October 1999: 60.

Samlip (2002). Development History of Door Module Systems. Samlip, Inc., http://samlip.co.kr/eng/products/

Sanchez, Ron and Joseph T. Mahoney (1996). "Modularity, Flexibility, and Knowledge Management in Product and Organization Design." Strategic Management Journal **17**(Winter Special Issue): 63-76.

Sanderson, Susan and Mustafa Uzumeri (1995). "Managing product families: The case of the Sony Walkman." Research Policy **24**(5): 761-782.

Sands, Jan, William Loughlin and Frank Lu (1998). "Equipment Standardization under Acquisition Reform." Journal of Ship Production **14**(2): 110-123.

Schilling, Melissa A. (2000). "Towards a general modular systems theory and its application to interfirm product modularity." Academy of Management Review **25**(2): 312-334.

Schilling, Melissa A. (2002 forthcoming). Modularity in Multiple Disciplines. Managing in the Modular Age: Architectures, Networks and Organizations. R. Garud, A. Kumaraswamy and R. N. Langlois. Oxford, Blackwell Publishers.

Schonberger, Richard J. (1986). World Class Manufacturing - The Lesson of Simplicity Applied. New York, The Free Press.

Shtub, Avraham and Ronen Versano (1999). "Estimating the cost of steel pipe bending, a comparison between neural networks and regression analysis." International Journal of Production Economics **62**: 201-207.

Siddique, Zahed and David W. Rosen (2000). Product Family Configuration Reasonings Using Discrete Design Spaces. 2000 ASME Design Engineering Technical Conferences, Baltimore, Maryland, ASME.

Siddique, Zahed, David W. Rosen and Nanxin Wang (1998). On the Applicability of Product Variety Design Concepts to Automotive Platform Commonality. ASME Design Engineering Technical Conferences, Atlanta, GA, ASME.

Simon, Herbert A. (1962). "The Architecture of Complexity." Proceedings of the American Philosophical Society **106**(6): 467-482.

Smith, Preston G. and Donald G. Reinertsen (1991). Developing Products In Half The Time. New York, Van Nostrand Reinhold.

Smith, Robert P. and Steven D. Eppinger (1997b). "A Predictive Model of Sequential Iteration in Engineering Design." Management Science **43**(8): 1104-1120.

Sodhi, M. and Winston A. Knight (1998). "Product Design for Disassembly and Bulk Recycling." CIRP Annals **47**(1): 115-118.

Sosa, Manuel, Steven D. Eppinger and Craig M. Rowles (2000). Designing Modular and Integrative Systems. 2000 ASME Design Engineering Technical Conferences, Baltimore, Maryland, ASME.

Starr, Martin K. (1965). "Modular Production - A New Concept." Harvard Business Review **43**(November-December): 131-142.

Steward, Donald V. (1981). System Analysis and Management: Structure, Strategy, and Design. New York/Princeton, Petrocelli Books.

Stone, Robert B. and Kristin L. Wood (2000). "Development of a Functional Basis for Design." Journal of Mechanical Design **122**(4): 359-370.

Stone, Robert B., Kristin L. Wood and Richard H. Crawford (1998). A Heuristic Method to Identify Modules from a Functional Description of a Product. ASME Design Engineering Technical Conferences, Atlanta, GA, ASME.

Stone, Robert B., Kristin L. Wood and Richard H. Crawford (2000a). "A heuristic method for identifying modules for product architectures." Design Studies **21**: 5-31.

Stone, Robert B., Kristin L. Wood and Richard H. Crawford (2000b). "Using quantitative functional models to develop product architectures." Design Studies **21**: 239-260.

Sudjianto, Agus and Kevin N. Otto (2001). Modularization to Support Multiple Brand Platforms. ASME Design Engineering Technical Conference, Pittsburgh, PA, ASME.

Suh, Nam P. (1990). The Principles of Design. New York, Oxford, Oxford University Press.

Suresh, Nallan C. and John M. Kay (1998). Group Technology & Cellular Manufacturing: Updated Perspectives. Group Technology and Cellular Manufacturing: A State-of-the Art Synthesis of Research and Practice. N. C. Suresh and J. M. Kay. Boston, Kluwer Academic Publishers: 1-14.

Suzue, Toshio and Akira Kohdate (1990). Variety Reduction Program - A Production Strategy for Product Diversification. Cambridge, MA, Productivity Press.

Swan, Willis A. (1914). "Proposes Standardization of Car Sizes." The Automobile **31**: 76-77.

Szykman, Simon (1996). Improving the Design Process by Predicting Downstream Values of Design Attributes. ASME Design Engineering Technical Conferences, Irvine, CA, ASME.

Tajima, Takamitsu and Masahiko Yasugahira (1999). "Development of front-end module." JSAE Review **20**(2): 223-228.

Tanaka, Takao (1993). "Target Costing at Toyota." Journal of Cost Management 7(1): 4-11.

Ten Brinke, E., Eric Lutters, Ton Streppel and H.J.J. Kals (2000). "Variant-based cost estimation based on Information Management." International Journal of Production Research **38**(17): 4467-4479.

Terwiesch, Christian and Christoph H. Loch (1999). "Managing the Process of Engineering Change Orders: The Case of the Climate Control System in Automobile Devlopment." Journal of Product Innovation Management **16**: 160-172.

The Economist Intelligence Unit (2000a). Developments in the global sports car sector. EIU Motor Business International: 43-69.

The Economist Intelligence Unit (2000b). Global niche vehicle markets: prospects to 2005. EIU Motor Business Journal: 150-164.

The Economist Intelligence Unit (2000c). New technology: alternative materials. EIU Components Business International: 70-84.

The Economist Intelligence Unit (2000f). Automotive supply chain: the revolution begins. EIU Components Business International: 130-150.

Thomas, Marlin U. (1999). "Some Economic Decision Problems in Warranty Planning." The Engineering Economist **44**(2): 184-196.

Thomke, Stefan H. (1997). "The role of flexibility in the developmnet of new products: An empirical study." Research Policy **26**: 105-119.

Thomke, Stefan H. and Takahiro Fujimoto (2000). "The Effect of "Front-Loading" Problem-Solving on Product Development Performance." Journal of Product Innovation Management **17**: 128-142.

Thomke, Stefan H. and Donald Reinertsen (1998). "Agile Product Development: Managing Development Flexibility in Uncertain Environments." California Management Review **41**(1): 8-30.

Thonemann, Ulrich W. and Margaret L. Brandeau (2000). "Optimal Commonality in Component Design." Operations Research **48**(1): 1-19.

Tikal, Franz and Christian Vollmer (1997). "Pkw-Tuerrahmen aus Magnesium." Werkstoffe im Automobilbau (Materials in the Automotive Industry)(Sonderausgabe von ATZ und MTZ): 30-32.

Townsend, John, Michael Kaczmar and Mohamed El-Sayed (2001). Modular Door System for Side Impact Safety of Motor Vehicles. 17th International Technical Conference on the Enhanced Safety of Vehicles (ESV), Amsterdam, Netherlands.

Tseng, Mitchell M. and Xuehong Du (1998). "Design by Customers for Mass Customization Products." CIRP Annals **47**(1): 103-106.

Tseng, Mitchell M. and Jianxin Jiao (1996). "Design for Mass Customization." CIRP Annals **45**(1): 153-156.

Tseng, Mitchell M. and Jianxin Jiao (1998). Design for Mass Customization by Developing Product Family Architecture. ASME Design Engineering Technical Conferences, Atlanta, GA, ASME.

Uduma, Kalu (2000). "Innovations in Auto Safety Design, a Key to Quality Improvement." Technological Forecasting and Social Change **64**: 197-208.

Ulrich, Karl T. (1995). "The role of product architecture in the manufacturing firm." Research Policy **24**: 419-440.

Ulrich, Karl T. and Steven D. Eppinger (2000). Product Design and Development. Boston, McGraw-Hill.

Ulrich, Karl T. and Scott A. Pearson (1993). Does product design really determine 80% of manufacturing cost? Working Paper WP# 3601-9, MIT Sloan School of Management, : 31.

Ulrich, Karl T. and Scott A. Pearson (1998). "Assessing the Importance of Design through Product Archaeology." Management Science **44**(3): 352-369.

Ulrich, Karl T., David Sartorius, Scott A. Pearson and Mark J. Jakiela (1993). "Including the Value of Time in Design-for-Manufacturing Decision Making." Management Science **39**(4): 429-447.

Ulrich, Karl T. and Karen Tung (1991). Fundamentals of Product Modularity. Working Paper WP# 3335-91-MSA, MIT Sloan School of Management, September 1991: 14.

Uppal, Kul B. (1996). "Estimating Engineered Equipment Costs." AACE Transactions **40**: EST.10.1-EST.10.6.

van Hoek, Remko I. and Harm A.M. Weken (1998). "The Impact of Modular Production on the Dynamics of Supply Chains." The International Journal of Logistics Management **9**(2): 35-50.

Vollrath, K. (2001). Flugzeug der Zukunft fliegt mit Feinguss. VDI nachrichten, Duesseldorf, March 30, 2001, 19.

von Hippel, Eric (1990). "Task partitioning: An innovation process variable." Research Policy **19**(5): 407-418.

Weustink, I.F., E. Ten Brinke, A.H. Streppel and H.J.J. Kals (2000). "A generic framework for cost estimation and cost control in product design." Journal of Materials Processing Technology **103**: 141-148.

Whitney, Daniel E. (1988). "Manufacturing Design." Harvard Business Review **66**(July-August): 83-91.

Whitney, Daniel E. (1993). "Nippondenso Co. Ltd: A Case Study of Strategic Product Design." Research in Engineering Design **5**: 1-20.

Whitney, Daniel E. (1996). Why mechanical design cannot be like VSLI design. Working Paper, Massachusetts Institute of Technology, April 1996: 32.

Whitney, Daniel E. (2002 forthcoming). Assembly.

Wilhelm, Bernd (1997). Platform and Modular Concepts at Volkswagen - Their Effects on the Assembly Process. Transforming Automobile Assembly - Experience in Automation and Work Organization. K. Shimokawa, U. Juergens and T. Fujimoto. Berlin, Springer: 146-156.

Womack, James P., Daniel T. Jones and Daniel Roos (1990). The Machine that changed the World - The Story of Lean Production. New York, Harper Perennial.

Yin, Robert K. (1994). Case Study Research - Design and Methods. Thousand Oaks, CA, Sage Publications.

Young, James A. III (1997). "Design Phase Cost Control." AACE International Transactions(CC.01.1): 24-27.

Yu, Janet S., Javier P. Gonzalez-Zugasti and Kevin N. Otto (1999). "Product Architecture Definition based upon Customer Demands." Journal of Mechanical Design **121**: 329-335.

Yu-Lee, Reginald Tomas (2001b). "The Reality of Costs." Industrial Management **42**(4): 29-33.

Zentrum Wertanalyse (1995). Wertanalyse: Idee - Methode - System. Duesseldorf, VDI-Verlag GmbH.

Zipkin, Paul (2001). "The Limits of Mass Customization." Sloan Management Review **42**(3): 81-87.