Michael Marti

Complexity Management

Optimizing Product Architecture
of Industrial Products

DUV GABLER EDITION WISSENSCHAFT

Michael Marti

Complexity Management

GABLER EDITION WISSENSCHAFT

Michael Marti

Complexity Management

Optimizing Product Architecture of Industrial Products

With a foreword by Prof. Dr. Thomas Friedli

Deutscher Universitäts-Verlag

Bibliografische Information Der Deutschen Nationalbibliothek
Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der
Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über
<http://dnb.d-nb.de> abrufbar.

Dissertation Universität St. Gallen, 2007

1. Auflage September 2007

Lektorat: Frauke Schindler / Sabine Schöller

Der Deutsche Universitäts-Verlag ist ein Unternehmen von Springer Science+Business Media.
www.duv.de

Umschlaggestaltung: Regine Zimmer, Dipl.-Designerin, Frankfurt/Main
Gedruckt auf säurefreiem und chlorfrei gebleichtem Papier
Printed in Germany

ISBN 978-3-8350-0866-3

Foreword

The discussion about increased product complexity in Western European companies has been omnipresent for decades. This is due to ever new demands (real or conceived) from the market and new possibilities (technical and others) open to the providers of products and services to come up fast with very specific solutions. The management of this complexity inside the company so as to achieve a balance between the market benefits of more customer specific solutions and the internal costs induced by this is still a big challenge. After years of research in this area there is still a lack of tools to help practitioners to take decisions about the right degree of complexity for a product or the right design of a product architecture.

Mr. Marti brings together three perspectives on this topic in an easily understandable and communicable way: The strategy of the company (combined with the maturity of the product), the functionality, and the physical complexity on a component level. The visualization in a complexity matrix is an important new tool especially for fostering the understanding for complexity issues in a company!

With this book Mr. Marti has come up with an outstanding contribution to product complexity management, which is crucial for the future competitiveness of companies in Western Europe. I hope that this book will find a broad distribution as well in practice as in theory for the sake of our countries!

Prof. Dr. Thomas Friedli

Preface

During the period of conducting the research for this thesis, I worked as a product manager for Siemens. I found the many interactions between theory and practice intriguingly fruitful, and by actively participating in both worlds I was also shown the different attitudes and needs of theorists and practitioners. I learned that it is of prime importance to companies to cope with the complexity surrounding them and ensure the competitive edge of their products without causing excessive complexity inside the firm. Doing research in the field of complexity management was a very rewarding task to me as it is a subject strongly related to applications in industry, all the while requiring a good theoretical understanding of complexity and its effects on enterprises.

This work would never have been possible without the contributions and intellectual support I received from many sides. Therefore, I have quite a long list of people to acknowledge for helping me with my dissertation.

First and foremost, I owe special thanks to Prof. Dr. Fritz Fahrni, who was my advisor and contributed decisively to the successful accomplishment of my dissertation. I benefited heavily from his many decades of industry experience, and he was able to give me a feel of what matters in management and what does not. I would also like to thank him for encouraging me to participate in a triathlon competition.

The next person I wish to thank is my co-advisor Prof. Dr. Thomas Friedli, who has always been there to give advice and guidance. I very much appreciated his supportive attitude and the excellent comments, which tremendously enhanced my dissertation.

I would also like to acknowledge all the people at my industrial partners who I worked with while conducting my case studies. I am especially grateful to Dr. Dirk Brusis, Dr. Jan Göpfert, Dr. Werner Hälg, Dr. Axel Hoynacki, Dr. Michael Ilmer, Dr. Klaus Mecking, and Dr. Thomas Rapp.

Special thanks go to Dr. Rolf Wohlgemuth, who enabled me to travel to Taiwan and attend the R&D Management Conference in Taipei and Hsinchu, where I presented my research results.

Next are the people with whom I had numerous discussions about product platforms, modularization, product architecture, my model and its application in the case studies, and complexity management in general. I am grateful to Dr. Björn Avak, Christoph Baur, Dr. Luca Bongulielmi, Noëlle Jufer, Michael Furrer, Rahel Parnitzki, and Katrin Tschannen.

I would also like to thank Barbara Heck for proofreading my thesis from a linguist's perspective and Patrick Fuchs for reading through the text and giving welcome comments from a fellow PhD candidate's point of view.

To my colleagues back at Siemens who had to handle an extra load of work during my absence of several months to finish my thesis: special thanks therefore also go to Dirk Bödeker, Jeanette Mai, and Walter Wögerer for their great support.

My wonderful fiancée Nina Schilling receives my deep gratitude for her loving support and her patience.

Winterthur, May 2007 Michael Marti

Brief Contents

Table of Contents

List of Figures

List of Tables

List of Acronyms

ABC	Activity-based costing
CAM	Computer aided manufacturing
CAS	Computer aided selling
CODP	Customer order decoupling point
CPV	Customer perceived value
DFA	Design for assembly
DFC	Design for configuration
DFM	Design for manufacturing
DFV	Design for variety
DFX	Design for X
DSM	Design structure matrix
IMVP	International Motor Vehicle Program
LHD	Left hand drive
METUS	Management Engineering Tool for Unified Systems
MFD	Modular function deployment
MIM	Module indication matrix
NBIC	National Bicycle Industrial Company
OPP	Order penetration point

PIMS	Profit impact of market strategies
PLC	Product life cycle
QFD	Quality function deployment
RHD	Right hand drive
ROI	Return on investment
SBU	Strategic business unit
SRS	System requirement specification
URS	User requirement specification
USB	Universal serial bus
USP	Unique selling proposition
VCR	Video cassette recorder
VMEA	Variant mode and effects analysis
VW	Volkswagen

Management Summary

In the field of complexity management, the two dimensions of external and internal complexity receive special attention from theorists and practitioners alike. The two complexity dimensions pose a major challenge to enterprises because they require different and often conflicting treatment. External complexity (customer requirements, competitive forces, technological changes, etc.) pushes companies to broaden their product portfolios and introduce product variety, which in turn increases the enterprise-internal complexity (such as product complexity, organizational complexity, production complexity, etc.). Efforts to reduce internal complexity and slash the corresponding complexity costs typically require compromising the customization of products. This in turn complicates the task of differentiating oneself from competitors.

This difficult situation calls for a procedure that investigates the two dimensions of external and internal complexity and provides guidelines for action as to how the two can be balanced. The complexity management model introduced in this work is based on the reasoning that product architecture determines to a considerable extent how external complexity is translated into physical products. The model exhibits a three-step procedure to optimize a product's architecture: (1) strategic and product life cycle aspects are assessed; (2) the product's complexity is assessed quantitatively by means of the complexity matrix, which considers the product's functionality and physical complexity; (3) based on the previous two steps, guidelines for action are derived as to how product architecture can be optimized.

The model was applied to four industrial products and was able to shed light on the sources of complexity. Product architecture was optimized according to the functionality and physical complexity of the products, and it was shown that the same or even increased customer benefit can be delivered while causing less internal complexity. As less internal complexity is associated with lower complexity costs, the complexity management model supports companies in their quest to increase product competitiveness.

Management Summary (Deutsch)

Auf dem Gebiet des Komplexitätsmanagements erfährt das Spannungsfeld zwischen externer und interner Komplexität eine erhöhte Aufmerksamkeit sowohl von der Theorie als auch der Praxis. Die beiden Dimensionen der Komplexität stellen eine grosse Herausforderung für Unternehmen dar, weil sie unterschiedliche und sich oft widersprechende Massnahmen erfordern. Externe Komplexität (Kundenanforderungen, Wettbewerbskräfte, technologische Veränderungen etc.) drängt Unternehmen dazu, ihr Produktsortiment auszuweiten und neue Produktvarianten einzuführen, was wiederum die unternehmensinterne Komplexität (Produktkomplexität, Organisationskomplexität, Produktionskomplexität etc.) steigert. Typischerweise bedingen Anstrengungen, die interne Komplexität zu verringern und die damit einhergehenden Komplexitätskosten zu senken, eine weniger stark ausgeprägte Individualisierung der Produkte. Dies erschwert aber die Aufgabe, sich von Wettbewerbern zu differenzieren.

Diese schwierige Situation verlangt nach einem Vorgehen, das externe und interne Komplexität untersucht und Handlungsempfehlungen abgibt, wie eine Balance zwischen den beiden Dimensionen erreicht werden kann. Das in dieser Arbeit vorgestellte Komplexitätsmanagement-Modell geht von der Erkenntnis aus, dass die Produktarchitektur zu einem massgeblichen Teil bestimmt, wie externe Komplexität in physische Produkte übersetzt wird. Das Modell besteht aus drei Schritten, um die Produktarchitektur zu optimieren: (1) Strategische Aspekte und solche des Produktlebenszyklus werden beurteilt; (2) die Produktkomplexität wird mit der Komplexitätsmatrix quantitativ untersucht, indem Funktionalität und physische Komplexität bewertet werden; (3) basierend auf den beiden vorhergehenden Schritten werden Handlungsempfehlungen abgeleitet, wie die Produktarchitektur optimiert werden kann.

Das Modell wurde bei vier Industrieprodukten angewendet und konnte die Ursachen von Komplexität aufzeigen. Die Produktarchitektur wurde anhand der Funktionalität und der physischen Komplexität der Produkte optimiert, und es wurde gezeigt, dass derselbe oder sogar ein erhöhter Kundennutzen bereitgestellt werden kann, wäh-

rend weniger interne Komplexität erzeugt wird. Weil weniger interne Komplexität weniger Komplexitätskosten bedeutet, unterstützt das Komplexitätsmanagement-Modell Unternehmen im Bestreben nach erhöhter Wettbewerbsfähigkeit ihrer Produkte.

1 Introduction

If one does not begin with a right atti-
tude, there is little hope for a right ending.

Kung Fu meditation.[1]

1.1 Problem Statement

A successful product must satisfy customer requirements and preferences. As this bundle of market needs has many facets and is highly complex in its nature, it is called *external complexity* here. To comply with these diverse demands, companies design their product portfolios accordingly, i.e. they introduce variety to their products. This, in turn, increases not only the product's complexity but affects the complexity within the entire company. This enterprise-internal complexity spreads to all functional areas (product development, logistics, production, and sales, to name a few) and is called *internal complexity.*[2] The products of an enterprise are exposed to external complexity and cause internal complexity. Therefore, products must be designed to cope with the implications of both external and internal complexity because they are a very important instrument for achieving sustained profits and assuring long-term survival.

Complexity is not an evil per se, though. Both the benefit created by product variants and the costs they cause must be weighed against each other in order to find the optimum combination (Rathnow, 1993, pp. 1-4 and pp. 41-42). The benefit side is explained by the purpose of product variety, which is to match the product with custom-

---

1 As cited in Klir and Elias (2003, p. 1)

2 The terms of external and internal complexity are widely used in literature about complexity management. A sample of sources is given here. Schuh and Schwenk (2001, pp. 13-17) emphasized the effects of excessive customer orientation (i.e. responding to external complexity) on internal complexity and complexity costs. Kaiser (1995) used the terms external (exogenous) and operative (endogenous) complexity (pp. 16-18) as well as external and internal complexity (pp. 100-101). Bliss (2000, pp. 5-7) introduced exogenous and endogenous complexity drivers.

ers' requirements as closely as possible and to acquire new customers, which increases sales, and retain existing ones. On the cost side, introducing product variants entails additional complexity costs that are effective initially (when the product is launched) as well as continuously over the product's life cycle. As the product variety benefits cannot be harvested without a rise in complexity costs, the goal is not to reduce product complexity as far as possible but to find the optimum level of complexity that takes into account the benefits as well as the costs generated by product variety.

As the product portfolio grows and variants proliferate, complexity costs do not spread equally among all product variants (Schuh & Schwenk, 2001, pp. 17-19). Due to a lack of economies of scale, low-sales variants generate more per unit costs than the high-sales variants, which are produced in larger numbers. A problem of traditional cost accounting systems lies in their insufficient capability of transparently tracing back all costs to the respective variants. As a result, low-sales variants are priced too low,[3] effectively being subsidized by the high-sales variants (Cooper & Kaplan, 1988a).

The *product architecture* inherently determines the nature of the complexity costs generated by all the variants of that product. It is a very important element in defining the internal complexity necessary to respond to the external (market) complexity. Depending on how the architectures of its products are structured, an enterprise can take advantage of a high degree of commonality – which keeps costs low – while still maintaining a sufficiently high level of distinctiveness – what customers care about.[4] Bearing in mind that complexity costs affect virtually all enterprise functions over the entire product life cycle, one can appreciate the importance of well-founded decisions concerning the product architecture.

---

[3] The price of low-sales variants depends on the pricing strategy. Because the costs of these variants appear to be lower than they actually are, the price tends to be too low to be profitable as well, no matter what the pricing strategy.

[4] Robertson and Ulrich (1998, p. 21) gave an excellent overview of how the product architecture influences the trade-off between commonality and distinctiveness.

Many methods exist that attempt to reduce complexity in product portfolios. The underlying rationale in all concepts is to trade off cost-cutting standardization and sales-increasing customization. Such methods include, among many others, the product platform (Meyer & Lehnerd, 1997; Robertson & Ulrich, 1998), mass customization[5], modularization (Baldwin & Clark, 1997; Ulrich & Tung, 1991), design for variety (Martin & Ishii, 1996), and modular function deployment (Erixon, 1998). These and other concepts will be presented in more detail in Chapter 3. However, none of them addresses product complexity explicitly and in a quantitative way and investigates the dependencies between a product's complexity and its architecture.

It can be said, therefore, that no method so far has been developed that attempts to quantify the complexity of a product in order to optimize the product architecture. Such a method is characterized by its potential to give valuable advice about how to structure a product's architecture, which reduces its complexity and the costs associated with complexity. Provided that the product's attractiveness from a customer perspective can be maintained, the product's competitiveness is increased and, as a result, the company's profits rise.

1.2 Research Objectives and Research Question

As can be seen from the current situation described in the previous section, a product's costs and its sales potential depend strongly on the product architecture. A means must be found to describe the complexity of a product and, based on such an evaluation, design the product architecture in such a way so as to decrease complexity costs as much as possible while at the same time providing as much customer value as possible. Such an optimization procedure must be complemented by product and enterprise strategy aspects. This ensures that a product's broader surroundings and the company's long-term direction are taken into account. Only in such a way can quantitative, "hard" factors be balanced with qualitative, "soft" aspects.

---

[5] See Pine II (1993a), Pine II (1993b), Pine II, Victor, and Boynton (1993), Gilmore and Pine II (1997), Piller (2003), and Levering (2003) for an introduction to the subject.

This work assumes the complexity of a product to be determined by essentially two dimensions, which will be explained in more detail in the subsequent chapters:

- *Functionality.* Describes customer requirements towards the product; provides customer value; represents the external (market) complexity encountered by the product and the enterprise.[6]

- *Physical complexity.* Accounts for how the market requirements are translated into the physical product; drives costs; represents a product's enterprise-internal aspects of complexity.[7]

The objective of the model presented in this work is to increase the competitiveness of the product. Therefore, the following research question lies at the center of this thesis:

> Can a product's competitiveness be increased by designing the product architecture according to functionality and physical complexity?

Because the objective of this thesis is to provide a model that can be applied in an industry context and optimizes the complexity of a product's architecture, it must take into account the very situation of the individual enterprise the model is applied to. Besides practical relevance, however, the model must show testability – which will be considered by action research conducted as case studies. Once such a model has been developed and proven valid, theorists as well as practitioners have at their disposal a powerful means to manage product complexity.

---

[6] In some manufacturing companies, the document describing customer requirements is referred to as user requirement specification (URS). The term used in German is *"Lastenheft."*

[7] In some manufacturing companies, the document describing the details of translating market requirements into an actual product is referred to as system requirement specification (SRS).

1.3 Reference Frame

The thesis' research is confined to industrial products as all case studies are performed in the machinery and process equipment industries. Electronics, software, and services are excluded from the research. Furthermore, all case studies are European-based. However, the literature considered in this work has a worldwide focus.

1.4 Methodological Approach

As opposed to "pure" basic science, where theories are developed to explain observed phenomena, *applied science* employs hypotheses and explanations that are provided by basic science and aims at applying them to practical problems (Ulrich, 1981, pp. 3-5). Business economics as an applied science provides the foundation of this work and should – following the St. Gallen management model – be perceived as a discipline that is concerned with forming, directing, and developing purpose-oriented social systems.[8] Ulrich and Hill (1979, pp. 165-168) divided the research process of management science into an *explorative,* an *explicative,* and an *application* context. Based on these fundamental considerations, the objective of this work is to investigate and describe a problem occurring in business reality, give explanations by means of developing a model, and test the practical applicability of the model and show the benefit it provides.

The research performed in this work is *qualitative.* For a quantitative investigation, a larger and more homogeneous sample would be needed (e.g. many comparable products in the same industry) to acquire the necessary data. Qualitative researchers maintain a tight relationship with the research object because they feel "a strong urge to 'get close' to the subjects being investigated – to be an insider" (Bryman, 1999, p. 38). As the application of this thesis' model in practice requires the researcher to take part in optimizing the product architecture, I consider qualitative research the more

[8] See Dyllick and Probst (1984, pp. 10-11) for an introduction to the system-oriented concept of management science.

suitable method for the purposes of this work. The research procedure followed by this thesis is *explicit* (or *deductive),* i.e. the model is developed in a first step and tested thereafter.[9] The existing work in the research area is abundant and provides a sufficient basis to derive a model. An exploratory investigation previous to the model development is therefore not considered necessary.

The model presented in this work is developed and tested by *action research* conducted as *case studies.* I believe relying both on action research and case study research is a viable combination as their underlying principles reinforce each other. They both emphasize the research object's real-life context and the research's relevance for practitioners.[10] According to Susman and Evered (1978, pp. 589-590), the characteristics of action research can be summarized as follows:

- *Future oriented.* As action research deals with the practical problems of people, it is oriented toward creating a more desirable future for them.
- *Collaborative.* Interdependence between researcher and practice is an essential feature of action research. Therefore, the interests of both sides take part in the research process.
- *Action research implies system development.* The system under investigation is enabled to develop itself within a cyclical process of diagnosing, action planning, action taking, evaluating, and specifying learning.
- *Action research generates theories grounded in action.*
- *Action research is agnostic.* The action researcher's recommendations for action are themselves the product of previously taken action, and the consequences of the actions cannot be fully predicted.

---

[9] An implicit (or inductive) procedure would imply that an introductory case is investigated in a first step. A model or theory is then developed based on the findings of that case. In a third step, the model is verified (or falsified) using additional cases.

[10] As an example of combining action research and case studies, see the cases presented by Greenwood and Levin (1998, pp. 33-49 and pp. 129-148), which they termed "action research cases."

- *Action research is situational.* The research object is a function of the situation as it is currently defined and, therefore, not free of its context.

I actively take part in the case studies and my research influences the architecture and complexity of the products I consider. Once I have applied the model I compare the situations before and after. Action research provides a very well suited methodological frame for such a type of investigation. Furthermore, the characteristics of action research as listed above provide considerable leverage for this work's research, especially the first, second, and fourth points.

It was said above that I first develop the complexity management model and then test it in real-life cases. This should not, however, obscure the fact that the model has been improved greatly based on just those very applications, i.e. the model has been partly developed and optimized thanks to the research performed. In that sense, this work's research has many aspects in common with *grounded theory,* which is "a research strategy whose purpose is to generate theory from data" (Punch, 2005, p. 155). Also, the work here does not attempt to verify some existing theory but aims at developing a new model and applying and testing it in practice. This objective is somewhat similar to the grounded theory approach, which uses deduction as well and does not solely employ inductive techniques.[11]

The advantage of building theories from cases comes from the increased likelihood of generating novel theory that is empirically valid and whose hypotheses prove testable (Eisenhardt, 1989, pp. 546-547). Table 1.1 summarizes the strengths and weaknesses of case study research. The research performed in this work draws on four case studies, which allows for a very direct and intimate connection to empirical reality. This, in turn, enables the proposed complexity management model to be mirrored very closely with business reality.

[11] Punch (2005, p. 158) argued that while the primary objective of grounded theory is to create a theory, "it is not long into the theorizing process before we are also wanting to test theoretical ideas which are emerging."

Table 1.1 Strengths and weaknesses of case study research[12]

Strengths	**Weaknesses**
• Likelihood of generating novel theory: the complex reality forces the researcher to "unfreeze" thinking and abandon his / her bias. • Increased testability: hypotheses can be proven false, results are measurable. • Empirical validity: theory-building process is intimately tied with evidence.	• Lacks simplicity: empirical evidence leads to overly complex theory that tries to capture everything. • Narrow and idiosyncratic theory.

Yin (2003, p. 13) defines a case study as "an empirical inquiry that investigates a contemporary phenomenon within its real-life context, especially when the boundaries between phenomenon and context are not clearly evident."[13] Therefore, conducting case studies ensures very close ties to the real-life frame of the research and increases its relevance for practice. Multiple (four) cases will be considered in this thesis,[14] which gives a sufficient breadth of research material. The data collected is qualitative and quantitative evidence.[15] While the case study research process proposed by Eisenhardt is divided into eight steps,[16] Yin (2003, p. 2) outlined four case study phases: design, data collection, analysis, and reporting. The case study research performed in this work follows the latter procedure.

The research procedure pursued in this thesis is based on Ulrich's (1981, p. 20) conception of systematically conducting applied research. Figure 1.1 outlines the re-

---

[12] Developed from Eisenhardt (1989, pp. 546-547).

[13] When considering case studies, it must be distinguished between case studies for research purposes and case studies as teaching devices (Yin, 2003, p. 2 and p. 10). Leenders and Erskine (1989) give an introduction of the case method used for teaching purposes.

[14] Case study research can be classified either as single-case or multiple-case design (Yin, 2003, pp. 39-40).

[15] Eisenhardt (1989, pp. 534-535) distinguishes qualitative (e.g. words) and quantitative (e.g. numbers) evidence.

[16] The respective steps are: getting started, selecting cases, crafting instruments and protocols, entering the field, analyzing data, shaping hypotheses, enfolding literature, and reaching closure (Eisenhardt, 1989, p. 533).

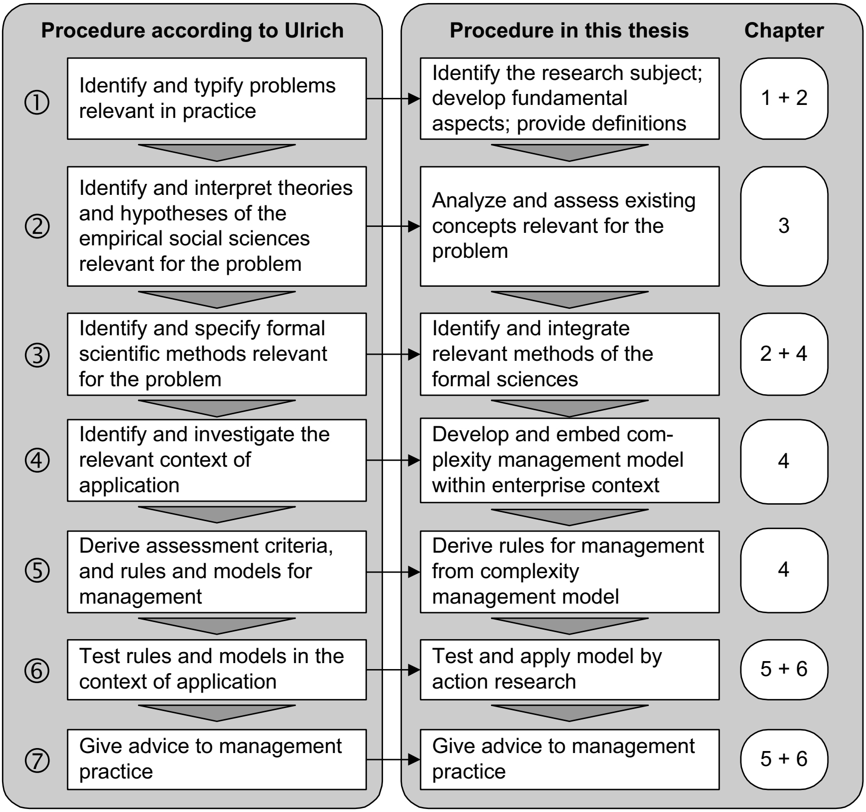


Figure 1.1 Research procedure and corresponding chapters

spective steps and how they are implemented in this work. The corresponding chapters are indicated as well.

1.5 Thesis Structure

The thesis structure is shown in Figure 1.2, starting with the present Chapter 1 followed by Chapter 2, which provides fundamental aspects on complexity in an enterprise setting and introduces the concept of external and internal complexity. A short

1 Introduction
Problem statement; research objectives and research question; reference frame; methodological approach; thesis structure

2 Background and Fundamental Concepts
- 2.1 Complexity as a Challenge for Enterprises
- 2.2 The Complexity of Systems
- 2.3 The Importance of Product Architecture
- 2.4 Concluding Remarks

3 Literature Review: Existing Concepts
- 3.1 Assessment Criteria
- 3.2 Managing Complexity on a Conceptual Level
- 3.3 Tools for Managing Complexity
- 3.4 Assessment Summary

4 Complexity Management Model
- 4.1 Overview
- 4.2 Strategy and Product Life Cycle Assessment
- 4.3 Product Complexity Assessment
- 4.4 Deriving Guidelines for Action
- 4.5 Summary of Complexity Management Model

5 Case Studies
- 5.1 Introduction
- 5.2 Railroad Signal
- 5.3 Liquid Handling Platform
- 5.4 Process Industry Compressor
- 5.5 Railroad Switch Lock

6 Conclusion
Reflecting on the research achievements;model limitations; reflecting on the research methodology; suggestions for future work

Figure 1.2 Structure of the thesis

overview of complexity within any system in general is given, and the importance of product architecture when managing product complexity is pointed out.

The current research status in the field is described in Chapter 3. Several models developed to cope with product complexity are presented and evaluated with respect to what they contribute to the thesis' research subject.

Chapter 4 develops the complexity management model proposed in this work. In the course of the chapter, the explanations draw on a simple and insightful example to present the model. The two major steps of the model – strategy and product life cycle assessment (Section 4.2) and product complexity assessment (Section 4.3) – form the basis for deriving the guidelines for action presented in Section 4.4.

In Chapter 5 the model is tested by applying it to four real-life products. The cases are all set in the machinery and process equipment industries. At the end of each section covering one case study, a critical evaluation is given showing the benefits and limits of the model.

Chapter 6 concludes this thesis and summarizes the major findings gathered in the course of the research. Based on a list of the most important open questions, suggestions for future research are given.

2 Background and Fundamental Concepts

2.1 Complexity as a Challenge for Enterprises

The full degree of variety potentially demanded will not, in general, be supplied because scale economies (even to a small degree) mean that the potential welfare or revenue gain from greater variety must be balanced against the lower unit production costs with fewer variants.

Kelvin Lancaster.[1]

Introductory Example: Shimano

When Shimano, one of the leading manufacturers of racing bicycle components, considers an extension of one of its product lines or even attempts to launch a completely new product, it must cater to a wide range of customer expectations. Cycling professionals all the way to occasional cyclists purchase their racing bicycles equipped with cranksets, brakes, hubs, derailleurs, chains, and cassette sprockets manufactured by Shimano. It is clear that a professional user, who sits on his / her bicycle for several hours per day, has very different requirements considering quality and functionality as compared to the occasional user, who is much more price sensitive.

Shimano grouped its product portfolio for racing bicycle components around the five brands Dura Ace, Ultegra, 105, Tiagra, and Sora, each designed for one specific customer segment. Even more variety is added to the portfolio by allowing for varying components, such as double or triple cranksets, different crank arm lengths, a variety

1 Lancaster (1990, p. 189)

of cassette sprocket combinations, etc. Browsing Shimano's product catalogue reveals interesting insights on how the company decided to respond to the large diversity of market needs: it provides a certain level of product variety which it believes to match with customers' preferences to a large extent. While this strategy certainly is more costly than producing one single standardized product, it allows Shimano to appeal to a wide range of customers. Source: Shimano (2006).

2.1.1 The Two Sides of Complexity

The above example illustrates two fundamental dimensions an enterprise is confronted with: on the one hand, it offers a product[2] on the marketplace that must fulfill certain customer requirements and preferences. On the other hand, the company chooses to develop and produce its product in its very specific way in order to respond to these market needs. It is obvious that the diverse demands of a large number of customer segments is difficult to cope with, especially when keeping in mind that customer requirements are dynamic. Furthermore, supplying a product to the market must also take into account the competitors, suppliers, legal regulations, technological developments etc.

As this bundle of market requirements and the other external factors are highly complex, the term of *external complexity* is introduced here to describe all influences on a product external to the company. The way in which the enterprise-internal value chain is formed strongly depends on the external complexity. The R&D department – to highlight the extreme positions – either develops a product to be sold several ten thousand times or designs it to one single customer's specifications. The production process might boast fully automated manufacturing equipment geared to an output of a large number of standardized goods or, alternatively, could be based on highly skilled workers manufacturing and assembling products in small lot sizes. External complex-

[2] I use the term "product" to include both products in the common sense as well as services.

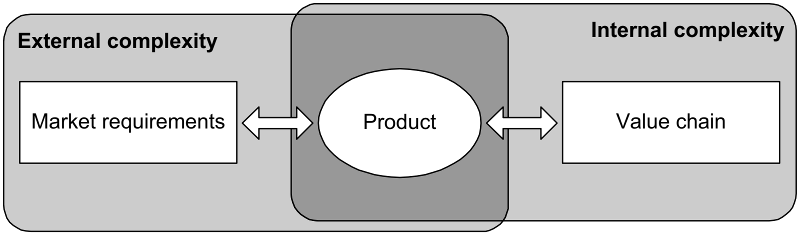


Figure 2.1 External and internal complexity from a product perspective

ity also affects the way in which the product's architecture is designed, i.e. what modules it consists of, how much variety it offers, which components are standardized, etc. This cluster describing the translation of market requirements into a physical product is called *internal complexity*. Figure 2.1 illustrates the situation described above.

Very similar to the above concept of external and internal complexity, Bliss (2000, pp. 5-7) introduced exogenous and endogenous complexity drivers. He identified three exogenous complexity drivers determining *market complexity:*

- *Demand complexity.* The increasingly individualized demand leads to fragmented markets with decreasing customer target group sizes and fast changing customer needs.

- *Competitive complexity.* Global and deregulated markets, powerful competitors and the shift from seller to buyer markets increase the market intensity and dynamics. These factors often cause a necessity for competitive differentiation and a broad and individualized product portfolio.

- *Technological complexity.* New technologies based on formerly distinct technologies merging into one discipline and shortening product life cycles cause a high degree of technological complexity.

These market complexity drivers, combined with *society complexity* – including aspects such as politics, economics, and legal issues as well as ecological and cultural aspects (Kirchhof, 2003, p. 39) – entail a certain degree of external complexity which the enterprise must adapt to by forming its internal complexity accordingly. Therefore, the above complexity drivers are complemented by a set of endogenous complexity drivers determining the enterprise-internal complexity:

- *Customer complexity.* Companies choose to serve a large number of heterogeneous customers and customer groups (e.g. different industries and / or different geographic segments), often with weak demand.
- *Product portfolio complexity.* Wide and diversified product portfolios are based on a large number of product variants.
- *Product complexity.* Product concepts are characterized by a large variety of raw materials, components, subassemblies, etc.

As the above complexity drivers are directly affected by the exogenous complexity drivers, they describe what Bliss (2000) termed *correlated enterprise complexity.* Bliss introduced four additional endogenous complexity drivers describing what he called *autonomous enterprise complexity.* They do not directly reflect the company's environment:

- *Production complexity.* The production is based on the philosophy of producing a considerable number of components and piece-parts in-house and is characterized by an order penetration point (OPP)[3] at a very early stage of the value chain.
- *Organizational complexity.* Enterprise processes become highly fragmented due to a strong orientation along functional lines and due to specialization. The interface density and fragmented responsibilities generate a high degree of organizational complexity.

[3] The order penetration point describes the location within the value chain where the production is no longer standardized but determined by a specific customer order. The terms customer order decoupling point (CODP) and point of variegation (Ramdas, 2003, p. 83) are used as synonyms.

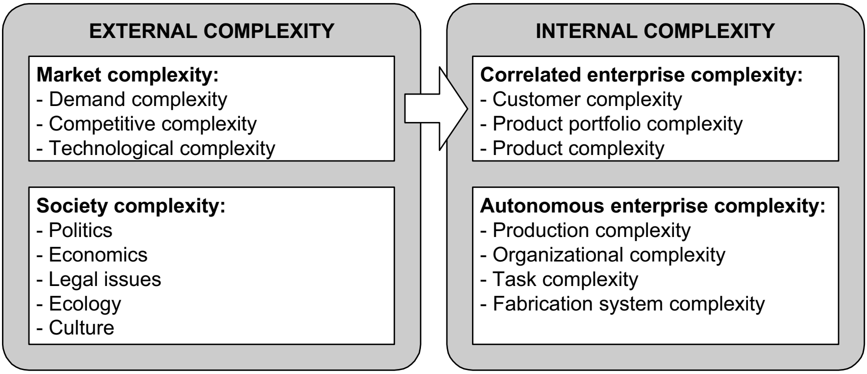


Figure 2.2 Complexity drivers forming external and internal complexity; slightly altered from Sekolec (2005, p. 15)

- *Task complexity.* Enterprises pursue a large variety of objectives in parallel.[4]
- *Fabrication system complexity.* Manufacturing systems adhering to a horizontally and vertically undifferentiated value chain are directed by a central and deterministic control system.[5]

Figure 2.2 summarizes the complexity drivers introduced above and depicts their allocation to external and internal complexity.

In a 1991 study, Cummings presented a list of what he called symptoms of complexity. They underscore the effects of the external complexity drivers on the enterprise. Among them are, according to Cummings, a large and increasing number of products or customers per sales dollar (e.g. 20 percent of the products generate 80 per-

4 See Campbell (1988) for details on task complexity. Campbell identifies four sources rendering a task complex: (1) presence of multiple paths to a desired end-state, (2) presence of multiple desired end-states, (3) presence of conflicting interdependence, and (4) presence of uncertainty or probabilistic linkages. Finally, Campbell presents a classification of complex tasks: decision tasks, judgment tasks, problem tasks, and fuzzy tasks.

5 Fabrication system complexity and production complexity are somewhat similar. Bliss (2000, pp. 7-8) argues that differentiating between the two complexity drivers is necessary because, for instance, a firm with a short value chain can avoid production complexity while still suffering from high fabrication system complexity.

cent of sales), a large and increasing number of unique inputs and suppliers, a high labor content (job shop operations rather than continuous batch processing), and large inventory pools (pp. 60-61). One further very common response of firms to cope with external complexity is introducing *product variety.* According to Kaiser (1995, pp. 100-101), the enterprise's task consists of designing appropriate output clusters (i.e. product variety) to fit with the heterogeneous market requirement clusters in the best possible way. The optimum state is achieved by matching the level of internal complexity to the degree of external complexity.

As shown above, complexity has many facets and cannot be fully described by one or two aspects. To illustrate this point, let's assume that a complexity level of 1 is defined by 50 customers, 145 product variants, 950 components, and 60 suppliers. If the number of components is reduced to 900, the complexity is unequivocally reduced. In this case, the number of components can be viewed as a measure of complexity – when applying the ceteris paribus condition. When the number of components is reduced by 50, the product variants by 10, and the customers by 4, the complexity is reduced, too, but a value for the complexity reduction cannot be determined. If some of the above complexity indicators are reduced and some increased, it is not even possible to decide whether the complexity has been raised or lowered (Adam & Johannwille, 1998, pp. 10-11).

It is of great importance to an enterprise to describe the effects on the costs (i.e. enterprise-internal complexity) and the benefits (i.e. responding to market requirements) associated with complexity. The example in the previous paragraph shows that one indicator (or, if possible, several) must be chosen as a measure of complexity. Rathnow (1993) based his considerations on product variety as a complexity indicator, leading to the concept of optimum variety, which considers the benefits and costs associated with product variety. It is based on the premise of increasing marginal costs and decreasing marginal benefits of variety. Conceptually, the optimum variety is defined by the point where the marginal benefit equals the marginal costs (see Figure 2.3). Rathnow pointed out that cost and benefit must be considered simultaneously to solve the

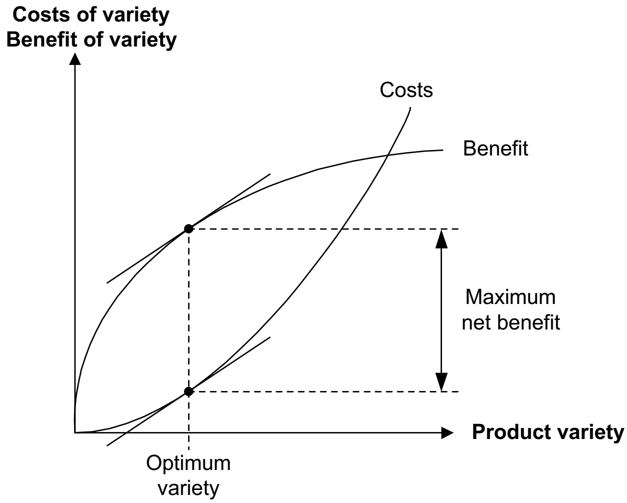


Figure 2.3 Conceptual description of costs and benefit associated with product variety; source: Rathnow (1993, p. 44)

optimization problem. This view is shared by Child, Diederichs, Sanders, and Wisniowski (1991), who contended:

> In order to optimize variety, a company must assess the level of variety at which consumers will still find its offering attractive and the level of complexity that will keep the company's costs low. Key to this decision is understanding the distinction between internal complexity and external variety. (p. 74)

Now that the fundamental concept of external and internal complexity has been introduced, the following two subsections cover the two dimensions in more detail. Subsection 2.1.2 on external complexity presents several existing concepts to assess market requirements and customer needs. The subject of subsection 2.1.3 on internal complexity is assessing the costs incurred by complexity.

2.1.2 External Complexity – Understanding the Market Needs

A product must be designed to match the target market's customer requirements[6] as closely as possible. These requirements mainly reflect – in terms of Figure 2.2 – the demand aspect of external complexity. The functionality offered by the product must therefore be compared with the customers' expectations, which allows the determination of the overlap of product offer and requirements. An under-engineered product (i.e. less functionality than required) compromises its competitive edge, while over-engineering (i.e. more functionality than required) causes costs that cannot be turned into profits (Figure 2.4). When the requirements are fulfilled at least to a large extent, the customers are satisfied and will stick with the product in the future – provided that price and quantity are in a favorable range and delivery is on-time (Seghezzi, 2003, p. 83).

The Kano model of customer satisfaction outlines a very useful classification of customer requirements. The three quality attributes that are identified by Kano, Seraku, Takahashi, and Tsuji (1984) include:

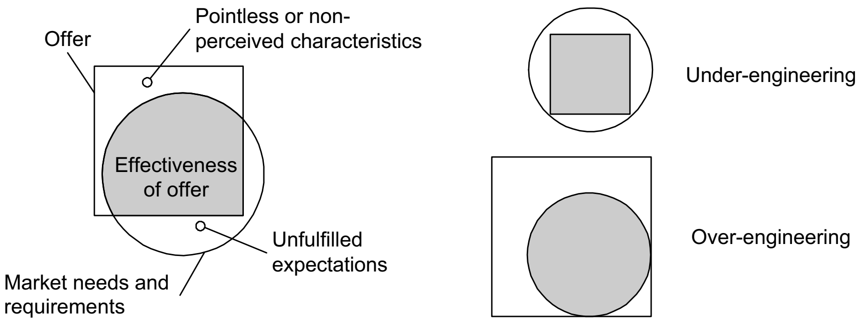


Figure 2.4 Overlap of offer and market requirements; sources: Teboul (1991, pp. 29-47) and Seghezzi (2003, p. 83)

---

[6] Following Ulrich and Eppinger (1995, p. 35), I choose to use the terms customer requirements, customer needs, and customer attributes as synonyms. They all label any attribute of a potential product that is desired by the customer.

- *Basic requirements* must necessarily be fulfilled because they are taken for granted. Customers do not normally spend much thought on basic requirements and, therefore, do not express them. Their presence does not result in customer satisfaction, but their absence causes strong dissatisfaction. An example of a basic requirement is providing toilet paper in a hotel room. Kano et al. (1984) call this type of quality attribute “must-be.”

- *Performance requirements* are at the top of customers’ minds when deciding on which product to buy. Hence, they will typically speak about them. Performance requirements can both satisfy and dissatisfy customers, depending on how well they are executed. A car’s fuel economy is an example of this type of customer need, termed “one-dimensional” quality attribute by Kano et al. (1984).

- *Excitement requirements* are unarticulated by customers and – when executed properly – delight customers and differentiate a company from its competitors. They mostly yield higher margins and are often referred to as USPs (unique selling propositions). While excitement requirements fascinate the customer (e.g. providing a 110 or 220 volt outlet in a car), they do not result in any dissatisfaction when they are absent. “Attractive quality” is the term Kano et al. (1984) coined for this type of market need.

Additionally, Kano et al. (1984) introduced the two quality attributes “indifferent” and “reverse.” Indifferent quality refers to aspects that are neither good nor bad and, therefore, do not result in either customer satisfaction or dissatisfaction. Reverse quality causes a high degree of dissatisfaction when included in the product (and vice versa). For example, some customers prefer the basic model of a product and are annoyed when too many features are included (Löfgren & Witell, 2005, p. 10). Depending on the dynamics of a market, a customer requirement will change from excitement to performance to basic. Kano provided empirical evidence for the dynamics of the television remote control, which has followed such a life cycle: Remote controls were an excitement requirement in 1983, a performance requirement in 1989, and a basic requirement in 1998 (as cited in Löfgren & Witell, 2005, p. 10). Figure 2.5 illustrates the Kano model and depicts the three types of customer requirements.

While the Kano model is able to classify customer requirements in general, it does not prove useful when simultaneously considering all potential customers. *Market segmentation*[7] jumps into this gap as it is a very powerful instrument for analyzing markets and coping with external (demand) complexity. Kotler and Keller (2006, pp. 240-242) introduced a typology of market segmentation based on customer preferences (see Figure 2.6):

- *Homogeneous preferences.* All consumers have roughly the same preferences;

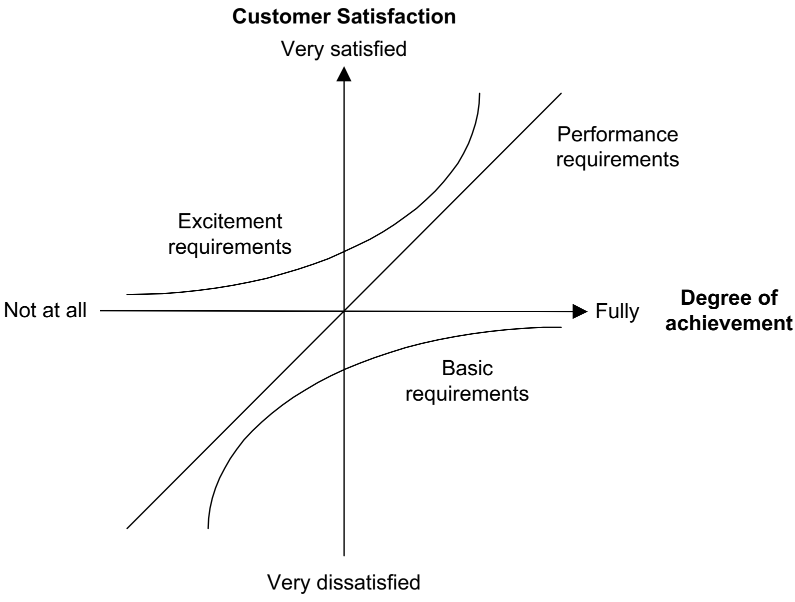


Figure 2.5 The Kano model of customer satisfaction (based on Löfgren & Wittell, 2005, p. 9)

---

[7] A market segment is defined here as "a group of customers who share a similar set of needs and wants" (Kotler & Keller, 2006, p. 240). Market segmentation can be performed in many different ways: geographical segments, preference segments, demographic segments, etc.

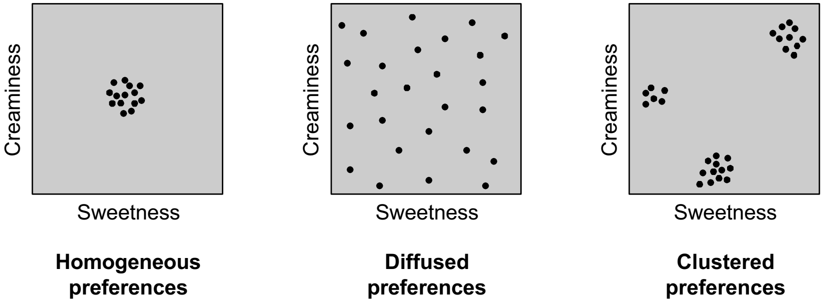


Figure 2.6 Basic market-preference patterns of ice cream buyers for the two product attributes creaminess and sweetness (Kotler & Keller, 2006, p. 242)

- *Diffused preferences.* Consumers vary greatly in their preferences; and
- *Clustered preferences.* The market reveals distinct preference clusters.

From a customer perspective, it is important to differentiate between customer benefit and customer value. *Customer benefit* refers to what the buyer receives by purchasing the product: functionality, assistance, warranty, brand name, etc. The costs (or *total customer costs)* consist of all costs a customer incurs to evaluate, buy, use, and dispose of the market offering (including monetary, time, energy, and psychic costs). *Customer value* (or customer perceived value, CPV) is the difference between benefits and costs (Kotler & Keller, 2006, p. 141). Therefore, a product will only be sold if the sum of all benefits is valued higher than all costs (see Figure 2.7).

A powerful concept to achieve a competitive edge and provide customer value was introduced by Clark and Fujimoto (1990), who reasoned that *product integrity* is the key to success. The extent to which a new product manages to balance basic functions and economy with more subtle characteristics is a measure of its integrity. Product integrity has both an external and an internal dimension. Internal integrity is characterized by the consistency of a product's functionality and its structure, while external integrity refers to the consistency between a product's performance and customers' expectations. In Clark and Fujimoto's understanding, a company that develops successful products is itself coherent and integrated. The strength of the product integrity

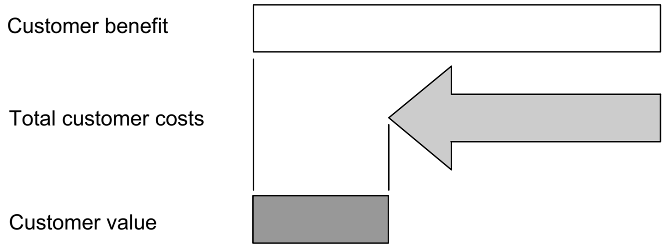


Figure 2.7 The concept of customer value; source: Rathnow (1993, p. 12)

concept resides, on the one hand, with the integration of both listening to customer needs as well as finding ways to actually organize the development of market-oriented products. On the other, the difficulty of capturing the full complexity of customers' requirements and expectations with all their facets is addressed, too, and ways to deal with such challenging situations are shown.

A quantitative approach widely used by marketing managers to assess customer preferences is provided by *conjoint analysis*[8], a set of techniques for measuring buyers' trade-offs among multi-attributed products (Green & Srinivasan, 1990, p. 3).[9] Customers commonly choose product alternatives by weighing characteristics that fall along more than one single dimension – they are multi-attribute (Green & Wind, 1975, p. 108). For example, one's preference for various houses may depend on the joint influence of such attributes as nearness to work, tax rates, quality of school system, and anticipated resale value (Green & Rao, 1971, p. 355). Conjoint analysis aids marketing managers in determining the relative importance of a product's multidimensional attributes, revealing to what extent they contribute to the product's overall attractiveness. Vriens (1994, pp. 39-40) provides an excellent example of how conjoint analysis is applied. Coffee-makers can be defined by the following attributes: price, brand

[8] The terms conjoint analysis and conjoint measurement are used as synonyms here.

[9] According to Green and Srinivasan (1978, p. 103), it is generally agreed that the start of conjoint measurement is marked by the work of Luce and Tukey (1964).

name, capacity, color, and the presence / absence of a flavor cap. Each of these attributes can adopt several values, shown in Table 2.1. The values of one attribute can be combined with all values of the other attributes, which adds up to 512 ($4 \times 4 \times 4 \times 4 \times 2$) possible variations of the coffee-maker.[10] The strength of conjoint analysis now unfolds: Respondents to a customer survey need only be asked to evaluate a limited number out of the complete set of 512 full profiles[11] (in this case, only 16 profiles were sufficient). The computation that then follows ranks the attributes according to their importance from a customer perspective, providing valuable information about how the different product characteristics are balanced against each other.

Table 2.1 Attributes and corresponding values of coffee-makers; as an example, one full profile is indicated by the line; source: Vriens (1994, p. 39)

Attribute	Value 1	Value 2	Value 3	Value 4
Capacity	Max. 6 cups	Max. 8 cups	Max. 10 cups	Max. 14 cups
Price	$20	$30	$40	$50
Brand	Philips	Moulinex	Rowenta	Ismet
Color	White	Black	Brown	Red
Flavor cap	Present	Absent		

Conjoint measurement can even go one step further and assign relative importance levels to the product attributes. Such a comparison is given in Figure 2.8, which depicts the relative importance of five attributes for a spot remover for carpets and upholstery. As can be seen from these examples, the fields of application for conjoint

10 Displaying product attributes and their values as shown in Table 2.1 is called an *attribute-value matrix.*

11 A full product description by one possible combination of its attributes and values is called *full profile.* An example of a full profile for the coffee-maker is indicated by the line in Table 2.1: max. 10 cups (capacity), $50 (price), Philips (brand), white (color), present (flavor cap).

measurement are wide and include marketing segmentation[12], product decisions[13], competitive analyses, pricing decisions, promotional decisions, and distribution purposes (Vriens, 1994, p. 41).

This subsection has shown a selection of the many facets of external complexity surrounding enterprises. The focus has clearly been placed on the demand aspects of external complexity as this will be one of the main subjects in the remainder of this work. The basic concepts and models presented above offer a first valuable assistance for the task of structuring the complex reality of market requirements and understanding what customers want. The next subsection covers the enterprise-internal aspects of complexity – i.e. how the market needs are reflected by products and processes and what costs are incurred thereby.

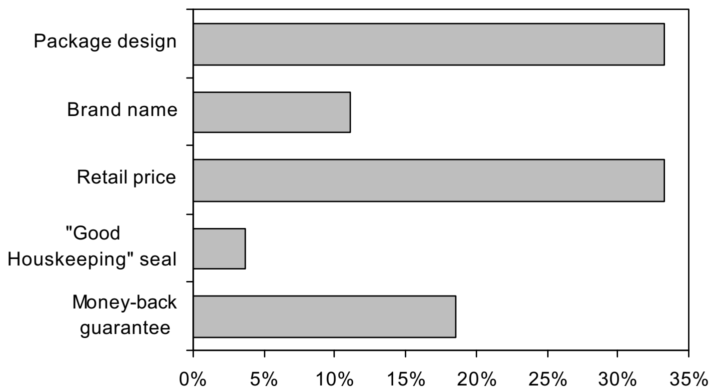


Figure 2.8 Relative importance of product attributes of a spot remover for carpets and upholstery; source: Green and Wind (1975, p. 110)

---

12 See Green and Krieger (1991) for a conceptual framework describing market segmentation in the context of conjoint analysis.

13 Page and Rosenbaum (1987) presented a case study showing how decisions on redesigns of product lines can be supported by conjoint analysis. Moore, Louviere, and Verma (1999) applied conjoint analysis to the design process of entire product platforms and argued this to be an approach that is superior to considering products individually.

2.1.3 Internal Complexity – The Cost Side of Complexity

The complex market environment and the diversity of customer requirements described in the previous subsection results in one very common and – especially in the context of this work – very important effect on enterprises: they generate product variety. In the strict terms of Figure 2.2, the complexity driver mainly considered here is product complexity. It will be shown shortly, though, that product variety affects all of a firm's functional areas. The strategic move of increasing the number of product variants undoubtedly has many beneficial aspects such as covering customers' needs more closely and increasing sales by tapping segments that were not catered to previously. However, the cost side of increased product variety must necessarily be considered, too, in order to be able to achieve the optimum level of variety that was introduced above.

The definition of *complexity cost* provided by Thonemann and Brandeau (2000, p. 1), "the cost of indirect functions at a company and its suppliers that are caused by component variety," will also be followed here, except that I consider variety at the product and not component level. The above authors reiterated the point that variety "affects almost every aspect of a company's costs, including those in accounting, logistics, material handling, production planning, purchasing, documentation, and research and development" (p. 4). When introducing a new product variant, the design team has to produce new drawings, a new article code must be assigned to the variant by the logistics department, and investments for new manufacturing tools might become necessary. Furthermore, marketing and sales are forced to adapt the product documentation if the new product variant adds a significant extension to the existing line, and often spare parts must be held on inventory for customers even if the product has ended its life cycle.[14] Figure 2.9 lists a selection of potential sources of complexity costs in detail.

---

[14] Rathnow (1993, pp. 20-23) differentiates between complexity costs that are incurred once (when the new product variant is introduced) and that occur continuously (during the entire life cycle of the variant).

Product life cycle \ Functional area	R&D	Procurement	Production	Marketing & Sales	Customer Service
Development cycle	• Drawings • Bills of material • Tests	• Search for and evaluation of additional suppliers	• Additional tooling • Additional work plans	• Additional training • More complex pricing	• Additional documentation • Additional training
Market cycle	• Adapting variants to technical or other changes	• Decreasing order volumes • No volume rebates	• Costlier production control • Longer set-up times • Larger inventory • Costlier quality control	• Larger finished goods inventory to maintain supply capability • More errors in order processing	• Decreasing "fix it right first time" quota
Disposal cycle	• Clearing up of data	• Costlier planning of product withdrawal	• Disposal of tools and other operating resources	• Costlier planning of product withdrawal	• Spare parts inventory 5-10 years after product withdrawal

Figure 2.9 Potential sources of complexity costs (Rathnow, 1993, p. 24)

A study in the automobile industry conducted by McKinsey found that up to 20% of total costs are incurred because of complexity, i.e. component and product variety (Rommel, Brück, Diederichs, Kempis, & Kluge, 1993, pp. 23-25). Figure 2.10 depicts in which functional areas these complexity costs are generated. Not surprisingly, the majority of costs stem from R&D and production. Child et al. (1991, p. 73) reported that complexity's costs even range from 10 to 40 percent of total costs. The empirical work by Hichert (1986a, p. 143) concluded that roughly 20% of total costs of the electrical appliances manufacturer he investigated were due to product variety.

Complexity costs are characterized by the fact that they are not easily reversed. When a company decides to reduce product variety, this does not automatically cancel all corresponding complexity costs. As the majority of these costs are fixed costs, their elimination is a long-term task. This phenomenon was called *cost remanence*[15] by Hichert (1986b) and is shown in Figure 2.11. The costs associated with increasing complexity stem, for instance, from more flexible (and more expensive) machines, or costlier IT systems. As these costs are fixed and because variety reduction usually means fewer revenues, profits decrease as a result of the cost remanence.

As products mature, variants typically proliferate in order to allow the product to attain the highest possible market share. When sales slow down and the product moves towards the end of its life cycle, firms usually do not reduce product variety to the same extent as the revenues decrease (see Figure 2.12). This is mainly due to the fact that "most companies have not developed a policy for handling aging products" (Kotler & Keller, 2006, p. 329). The increasing ratio of product variants to sales during the maturity and decline phase depicted in Figure 2.12 leads to a sales distribution steadily

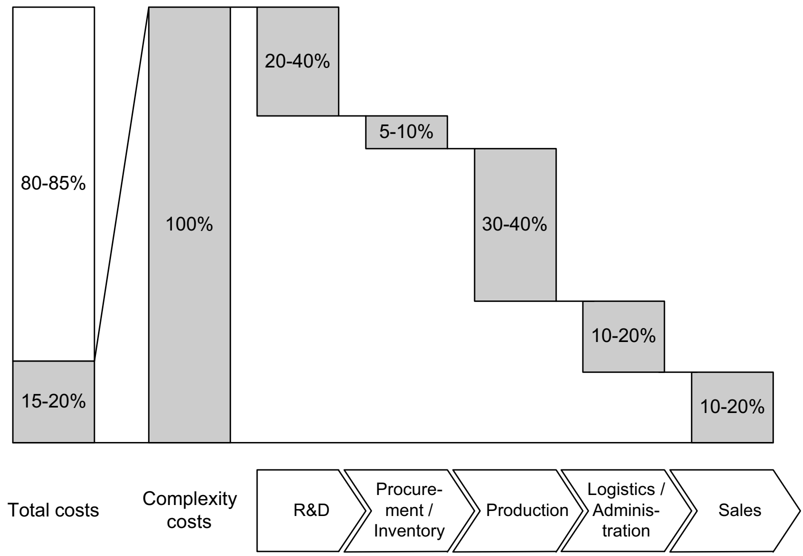


Figure 2.10 Complexity cost structure of an automobile manufacturer (Rommel et al., 1993, p. 24)

[15] The term remanence is taken from physics, where it denotes the remaining magnetization of a medium after an external magnetic field is removed (hysteresis phenomenon).

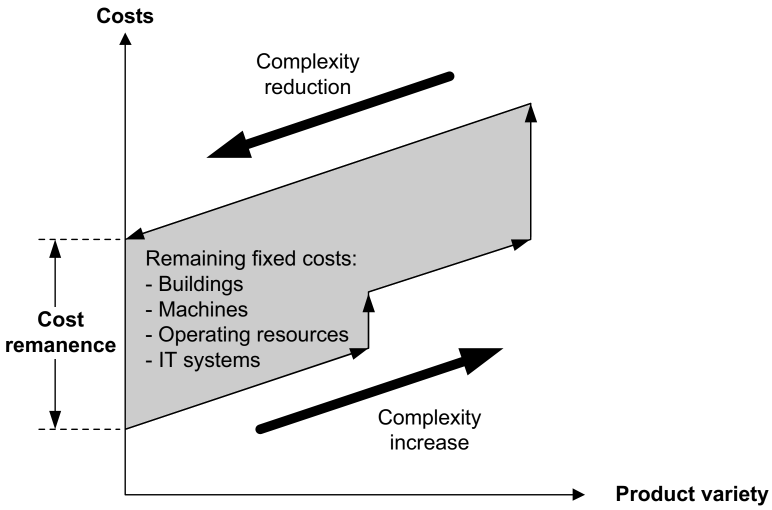


Figure 2.11 Remanence of complexity costs; derived from Hichert (1986b, pp. 673-674), figure according to Kaiser (1995, p. 31) and Rathnow (1993, p. 26)

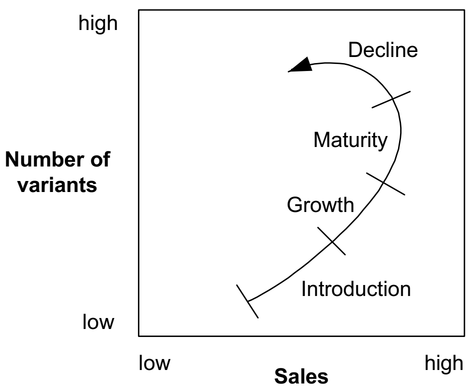


Figure 2.12 Number of variants and sales in the course of a product's life cycle; source: Hichert (1987, p. 227); life cycle phases according to Kotler and Keller (2006, p. 322)

shifting away from the high-sales variants towards less common (more “exotic”) variants. This development is illustrated in Figure 2.13 on page 32.

The situation is exacerbated by the inability of today’s commonly employed cost accounting systems of correctly allocating overhead costs to each and every product variant. Indirect costs (which are mostly considered “fixed”) are assigned to products according to labor hours, machine hours, and / or material dollars. While this might have been a fairly accurate estimate several decades ago, overhead is becoming the dominant portion of manufacturing costs.[16] For instance, as a result of more automated machinery, direct labor increasingly is concerned with set-up and supervisory functions rather than the actual work on the product. Therefore, direct labor is no longer a reasonable indicator of overhead consumption of a product (Cooper & Kaplan, 1988a). As a consequence, most companies’ accounting systems assign too low a fraction of overhead to low-sales variants, while the standard variants (due to their high volumes) are burdened with too high a fraction. Schuh and Schwenk (2001, p. 17) argued that the economies of scale expected when introducing new variants are overestimated and the diseconomies of scope underestimated. Cooper and Kaplan (1988a) neatly described the problem commonly encountered in enterprises today as follows:

> Low-volume products create more transactions per unit manufactured than their high-volume counterparts. The per unit share of these costs should, therefore, be higher for the low-volume products. But when volume-related bases are used exclusively to allocate support-department costs, high-volume and low-volume products receive similar transaction-related costs. When only volume-related bases are used for second-stage allocations, high-volume products receive an excessively high fraction of support-department costs and, therefore, subsidize the low-volume products. As the range between low-volume and high-volume products increases, the degree of cross-subsidization rises. Support departments expand to cope with the additional complexity of more products, leading to increased overhead charges.

[16] See Miller and Vollmann (1985, pp. 143-144) for a study on the development of overhead costs in several segments of American industry.

> The reported product cost of all products consequently increases. The high-volume products appear more expensive to produce than previously, even though they are not responsible for the additional costs. The costs triggered by the introduction of new, low-volume products are systematically shifted to high-volume products that may be placing relatively few demands on the plant's support departments. (p. 24)

The cross-subsidization of low-volume variants by the standard variants of a product is summarized in Figure 2.13.

In the 1980s, *activity-based costing* (ABC) was developed as a reaction to the deficits of traditional accounting systems becoming more apparent (Seiler, 1998, p. 242). ABC is a management accounting practice that identifies all of an organization's major operating activities (both production and non-production), traces costs to those activities, and then assigns costs to products or services that use the resources and services supplied by those activities (Needles, Powers, & Crosson, 2002, p. 815). The aim of ABC is to assign costs to products (and their individual variants) fairly and to determine their unit cost. It equips managers with more accurate product cost data and,

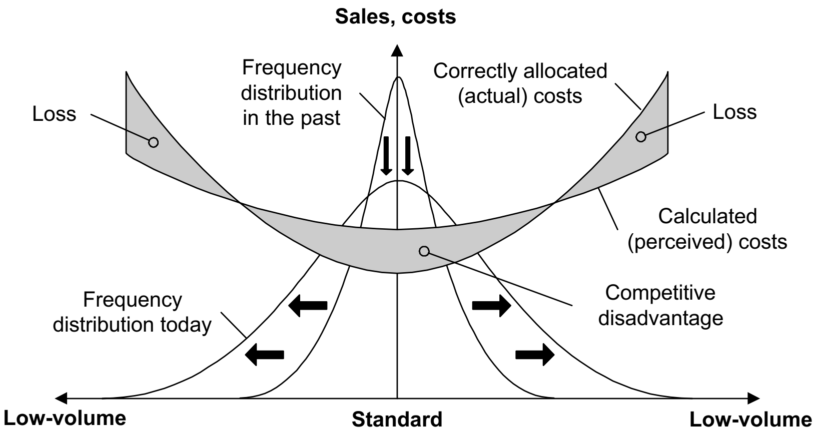


Figure 2.13 Shift from high-volume to low-volume product variants and their cross-subsidization; source: Schuh and Schwenk (2001, p. 18)

therefore, helps to prevent a firm from making misinformed decisions on product strategy. As a more detailed treatment of ABC is beyond the scope of this work, I refer to the abundant literature covering the subject.[17]

2.2 The Complexity of Systems

Everything is connected, but some things are more connected than others. The world is a large matrix of interactions in which most of the entries are very close to zero.

Herbert A. Simon[18]

A *system* can be described as an assembly of elements related in an organized whole (Flood & Carson, 1988, p. 7). Therefore, a number of elements and a corresponding number of relationships connecting the elements constitute a system of any kind, be it an insect, a corporate organization, or an aircraft. *Elements* can be thought of as components, building blocks, piece parts, or ingredients, while *relationships* take the form of interfaces, functional dependencies, communication channels, or interactions of any kind (Patzak, 1982, p. 19). A widely used taxonomy of interactions between elements was introduced by Pimmler and Eppinger (1994, p. 346), who considered four generic interaction types:

- *Spatial.* Adjacency or orientation between two elements;

[17] See Cooper and Kaplan (1988b) for an introduction to ABC and an illustrative example showing the differences from traditional cost accounting systems. Turney (1992) gave an overview of activity-based management, which he sees as guiding the continuous improvement process, while ABC supplies the necessary data. Johnson (1992) – himself a major contributor to the rise of the activity-based concept – placed a warning of overestimating ABC's abilities and emphasized the need to integrate the customer perspective. Further reading is provided by Horváth and Mayer (1989), and Schulte (1991), who stressed the ability of ABC to support variety reduction. An interesting comparison of target costing and activity-based costing is given by Sakurai and Keating (1994), who concluded that ABC is a management accounting system focusing on operating costs and analyzing product profitability, while target costing is a tool to direct the design process and reduce material and component cost.

[18] Simon (1973), p. 23

- *Energy.* Energy transfer between two elements;
- *Information.* Information or signal exchange between two elements;
- *Material.* Material exchange between two elements.

Any characteristic quality or property ascribed to an element is termed an *attribute* of that element (such as color, size, strength, and shape). The way in which the elements are related to each other is called the *structure* of a system. It is important to distinguish a system from its *environment,* which necessarily entails the definition of a system boundary. The elements of a *closed system* do not engage in relationships with anything outside the system, while *open systems* share spatial relationships or exchange material, information, or energy with their environment across the boundary (Flood & Carson, 1988, pp. 7-11).

Now, what makes a system complex? In everyday language, “complex” means composed of many interconnected parts, compound, composite, and characterized by a very complicated or involved arrangement of parts, units, etc. (Lexico Publishing Group, 2007). Originally, the word stems from the Latin word complexus, the past participle of complecti, which means to embrace, include, and unite. In a 1962 work on system complexity, Simon wrote:

> Roughly, by a complex system I mean one made up of a large number of parts that interact in a nonsimple way. In such systems, the whole is more than the sum of the parts, not in an ultimate, metaphysical sense, but in the important pragmatic sense that, given the properties of the parts and the laws of their interaction, it is not a trivial matter to infer the properties of the whole. (p. 468)

Patzak (1982, pp. 22-24) viewed complexity as an attribute of a system and differentiates between two aspects of complexity: connectivity and variety. Connectivity describes the number and diversity of relationships, while variety is determined by the number and diversity of elements. Patzak’s definition of complexity is illustrated in Figure 2.14, from which it can clearly be seen that according to this definition, complexity consists of four dimensions: (1) the number of relationships between the

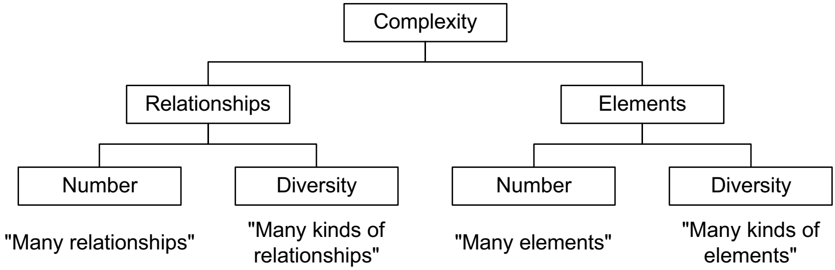
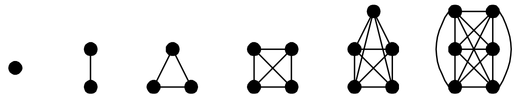


Figure 2.14 Complexity of a system; derived from Patzak (1982, p. 23)

elements of a system, (2) the number of different kinds of relationships between the elements of a system, (3) the number of elements constituting a system, and (4) the number of different kinds of elements constituting a system.

The amount of information needed to describe a system increases with its complexity. A simple three-element system consists of a maximum of three relationships. Assuming the elements are fixed and the relationships are bidirectional and not varying in fashion and intensity and can take on only two states – "present" and "absent" – (this assumption itself is already a strongly simplifying constraint on the system), the system can take on eight different states (or structures). By merely increasing the number of elements to six, the same calculation leads to a total number of 32,768 possible structures (see Figure 2.15). If the elements were traffic lights and could either emit red, yellow, or green light, it is obvious that the system's complexity would increase even more. This basic example shows that describing (let alone handling) a system with a large number of varying elements and relationships is indeed a complex task.

A classification of systems according to a well-defined scheme has been suggested by various authors. Checkland (1981, pp. 110-121) introduced a systems typology based on five distinct types: natural systems (such as subatomic systems, living systems such as animals or plants, or galactic systems), designed physical systems (man-made systems, e.g. hammers, tram cars, or space rockets), designed abstract systems (mathematics, poems, philosophies), human activity systems (including all human acts



Number of elements	1	2	3	4	5	6	n
Number of possible relationships	0	1	3	6	10	15	$\frac{n}{2}(n-1)$
Number of possible different system structures	1	2	8	64	1,024	32,768	$2^{\frac{n}{2}(n-1)}$

Figure 2.15 Elements, relationships, and system structures as a measure of complexity; based on Flood and Carson (1988, p. 23), and Göpfert (1998, p. 42)

of design, for instance, wielding a hammer), and transcendental systems (including all systems beyond knowledge). While Checkland's typology is intentionally very general in nature and does not investigate systems with respect to their complexity, Tushman and Rosenkopf (1992) presented an approach that focuses on products as systems and ranks them on a complexity scale from simple to complex:

- *Non-assembled products* are the simplest form, characterized by no separable components. Their technological essence stems from a manufacturing process or raw material. Examples: aluminum, cement, petroleum, paper, and springs.

- *Simple assembled products* are made up of distinct subsystems that are combined or fit together. Examples: stoves, cans, skis, and guns.

- *Closed systems* consist of a set of subsystems that is enclosed by a clear boundary. Examples: watch, bicycle, VCR, automobile, and airplane.

- *Open systems* are constituted by a number of subsystems (often closed systems) that are dispersed and not enclosed. The subsystems can be viewed as networked components working together over a distance. Examples: telephone system, rail-road, and power systems.[19]

The four product types introduced by Tushman and Rosenkopf (1992) do not have clear boundaries as they blend into each other: one type cannot be sharply separated from the other. Tushman and Rosenkopf's typology can be summarized as shown in Figure 2.16.

A further very important classification of systems is the concept of modular and integral system architectures. It is strongly rooted in Simon's (1962) theory of hierarchic systems and near-decomposability.

- A *nearly decomposable system* is composed of subsystems among which the interactions are weak but not negligible, i.e. intra-component linkages are generally stronger than inter-component linkages.

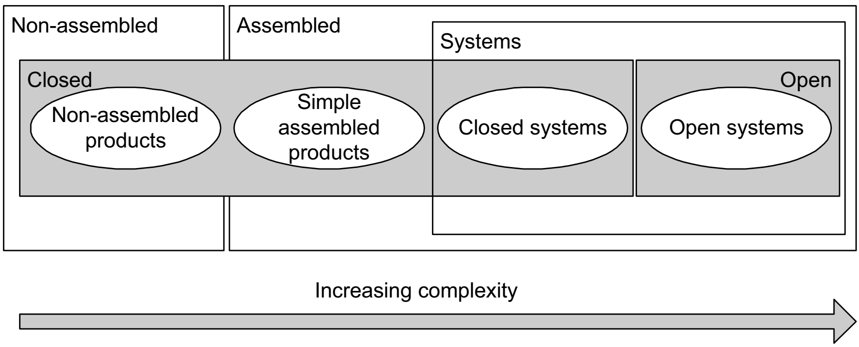
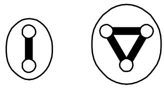
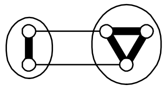
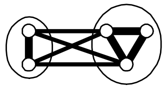
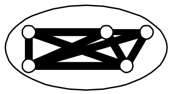
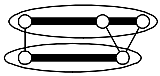


Figure 2.16 Complexity-based typology of systems; based on Tushman and Rosenkopf (1992)

[19] Open systems as presented here are similar to the concept of complex products and systems (CoPS) as introduced by Hobday, Rush, and Tidd (2000). They define CoPS as "high cost, technology-intensive, customised, capital goods, systems, networks, control units, software packages, constructs and services" (pp. 793-794). As an analytical category, they are a subset of capital goods.

- A *hierarchic system* consists of interrelated subsystems, each of which is hierarchic in structure until some lowest level of elementary subsystem is reached.[20] Hierarchic systems have the property of near-decomposability.

System architecture type	Strength of external and internal relationships	Example
System disintegrates into subsystems	$R_{external} \approx 0$	
Modular system architecture	$R_{external} << R_{internal}$	
Integral system architecture	$R_{external} < R_{internal}$	
Subsystems merge into one single unit	$R_{external} \approx R_{internal}$	
New system structure	$R_{external} > R_{internal}$	

Legend:			
	$R_{external}$	Strenght of external relationships (among subsystems)	Strong relationship
			Intermediate relationship
	$R_{internal}$	Strenght of internal relationships (within subsystems)	Weak relationship

Figure 2.17 System architecture classification based on strength of external and internal relationships; source: Göpfert (1998, p. 32)

[20] The definition of an elementary subsystem is a somewhat arbitrary task. It is common practice to adapt the definition according to the problem in question. When studying traffic flow, it suffices to treat a car as the elementary subsystem, while considering some detail of the car's acceleration process requires the elementary subsystem to be set at a lower level, e.g. the car engine.

As many complex systems have a nearly decomposable, hierarchic structure, they are more easily understood and described. Göpfert (1998, p. 30) defined a *modular system architecture* as being characterized by the property of near-decomposability, i.e. consisting of relatively autonomous subsystems. A *module,* therefore, can be defined as a special subsystem whose internal relationships are much stronger than the relationships with other subsystems (e.g. to other modules). On the other end of the scale, in *integral system architectures,* the relationships among subsystems are more pronounced. As a result, subsystems are more dependent on each other and less easily distinguished; they lose their autonomy. Figure 2.17 depicts a typology of system architectures based on the relative strength of external and internal relationships. Applying the theory of systems architecture to products leads to the concept of product architecture. This will be the subject of the next section.

2.3 The Importance of Product Architecture

2.3.1 Definition and Implications of Product Architecture

Product architecture is defined by Ulrich (1995, p. 419) as "the scheme by which the function of a product is allocated to physical components." It is constituted by three distinct aspects:

- The *structure of functionality* determines the arrangement of the overall function, sub-functions, and functional elements, which are commonly expressed as terms consisting of a verb and a noun, like in "reduce speed," or "increase pressure."[21] The structure of functionality describes what the product does. It is characterized

[21] I use the term structure of functionality because a product's functionality is displayed in a structured way. Furthermore, by referring to the term "functionality," the link to external complexity (see Section 0) and to the research question (see Section 1.2) becomes evident. The structure of functionality has been variously called a function structure, a function diagram, a functional description, and a schematic description (Ulrich, 1995, p. 420). In design theory, the term "function structure" is favored by both Pahl and Beitz (2003, p. 31) and Hubka and Eder (1988, pp. 72-77, and p. 257). Functional elements are sometimes referred to as functional requirements or functives (Ulrich, 1995, p. 420). The term elemental function is sometimes also used in literature.

by the fact that the higher its level of detail, the more assumptions about how the product physically works are embodied.

- The *structure of physical components* identifies the individual components[22] – the physical elements of the product – and their organization into subassemblies.[23] The collection of components implements the functions of the product. The specification of the interfaces among interacting components is often part of the structure of physical components, as is information about product variety (e.g. the variants of one particular component). While the structure of functionality is concerned with *what* the product does, the structure of physical components describes *how* it is done.

- The mapping from the structure of functionality to the structure of physical components is determined by functional elements implemented by physical components. The mapping can be one-to-one, many-to-one, or one-to-many (Ulrich, 1995, p. 421), depending on the type of product architecture.

It is readily seen that the structure of functionality defines the requirements of what the product must be able to do, i.e. the needs and expectations of customers or, in terms of Section 2.1, the external complexity. The structure of physical components, which describes how the required functionality is translated into a physical product, may be viewed as a representation of the cost effects that, for instance, product variety, optional product features, and non-standardized interfaces exert on the enterprise's value chain. Thus, the structure of physical components is an important indicator of internal complexity. Figure 2.18 depicts a schematic product architecture, showing, on the left, the hierarchy of the structure of functionality from the overall function all the

[22] I define a component similarly to Ulrich (1995, p. 421) as a separable physical unit or, more generally, any distinct region of the product. The definition of a component strongly depends on the problem at hand. In a simple product, components might be individual piece parts, while in a complex context a component might be composed of a large number of individual piece parts (e.g. defining the jet engine as one component of an aircraft). In any case, I consider a component as the physical unit designating the lowest hierarchic level of a product's structure of physical components.

[23] The subassemblies resulting from the grouping of components into major building blocks are often referred to as chunks. See Ulrich and Eppinger (1995, pp. 131-132).

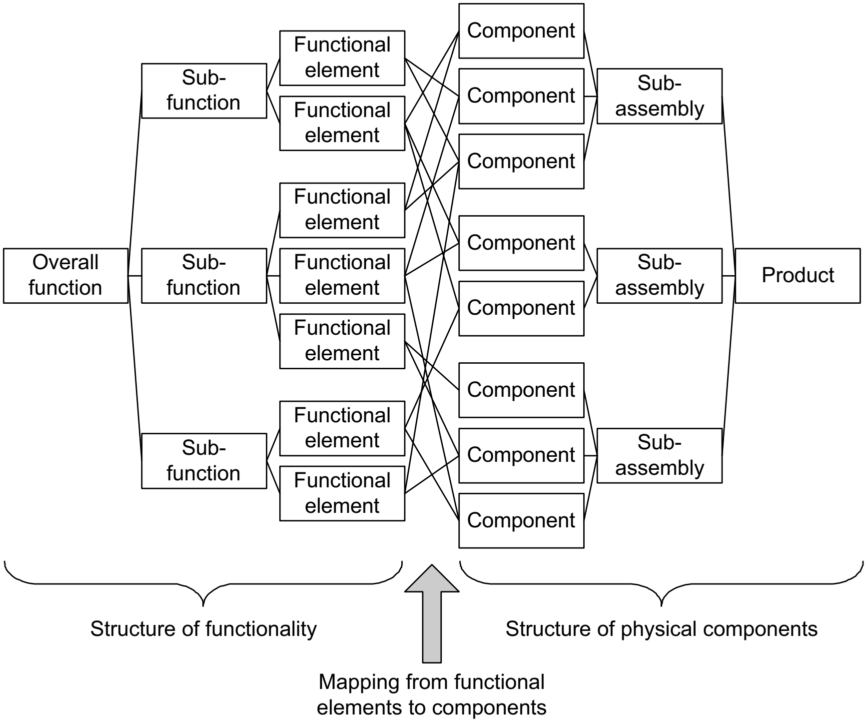


Figure 2.18 Schematic product architecture[24]

way to the functional elements. On the right hand side of Figure 2.18, the structure of physical components with the components and subassemblies can be seen. The mapping from functional elements to components is indicated by the lines connecting the structures of functionality and physical components, respectively. The number of hierarchic levels of both the structure of functionality and the structure of physical components (three levels in Figure 2.18) can vary depending on the product under investigation and on the desired level of detail.

[24] The figure is based on Göpfert and Steinbrecher (2000, p. 25) and was supplemented with additional aspects.

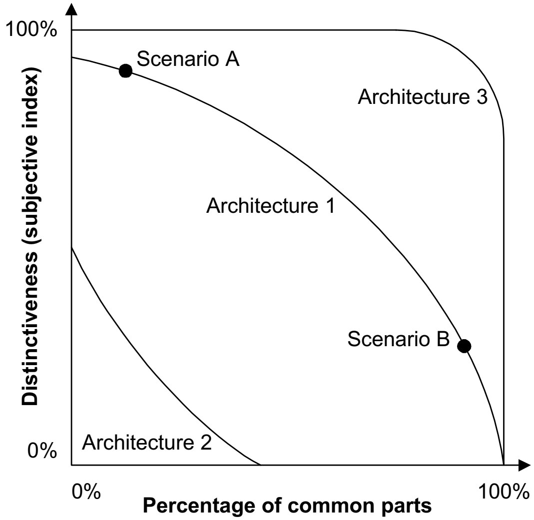


Figure 2.19 Trade-off between distinctiveness and commonality; source: Robertson and Ulrich (1998, p. 22)

The importance of product architecture in a complexity management context stems from the widely accepted fact that "a manufacturing system's ability to create variety resides … with the architecture of the product" (Ulrich, 1995, p. 428). Child et al. (1991, p. 79) reported that as much as 80 percent of costs, 50 percent of quality, 50 percent of time, and about 80 percent of business complexity can be influenced through product and process design, both of which are directly related to product architecture. Product architecture decisions have profound implications on several issues of fundamental importance to the entire enterprise, ranging from product performance, product change, product variety, component standardization, manufacturability, and product development management (Ulrich & Eppinger, 1995, pp. 133-138).

Robertson and Ulrich (1998) argued that product architecture dictates the way in which the product balances commonality (i.e. reducing costs through economies of scale) and distinctiveness (which enhances the competitive edge). Figure 2.19 shows the trade-off of a sample product architecture (architecture 1 in Figure 2.19) allowing, at one point, very distinctive products that share very few common components (scenario A) and, at another point (scenario B), less distinctive products sharing many components. The nature of the trade-off between commonality and distinctiveness can be influenced by altering the product architecture. Architecture 2 in Figure 2.19 decreases rapidly in distinctiveness for only a slight effort to increase shared components. Such a product architecture must, of course, be avoided at any rate. The ideal case is represented by architecture 3 in Figure 2.19, which is characterized by a high degree of commonality without much sacrifice in distinctiveness.

2.3.2 Modular and Integral Product Architectures

The architecture of a product may be classified in many different ways, the most important characteristic probably being its modularity (Ulrich & Eppinger, 1995, p. 132). As modularizing products is an important concept that is widely employed in industry, a full section of the next chapter is dedicated to modularization. Therefore, only a short introduction is given here. Modular system architectures were introduced in Section 2.2, where it was seen that they are composed of a set of relative autonomous subsystems. Similarly, a *modular product architecture* comprises a number of *modules,* which can be described as relatively autonomous subassemblies. The opposite of a modular product architecture is an *integral product architecture.* A product is not either modular or integral but can vary in its degree of modularity.[25] The extreme case of a completely modular architecture includes a one-to-one mapping from functional elements to the physical components, and specifies *decoupled interfaces* between components. An integral architecture includes a complex (non one-to-one) mapping from functional elements to the components and *coupled interfaces*[26] between components (Ulrich, 1995, p. 422).

[25] Therefore, modularity is a gradual (and not a discrete) property of products.

[26] A coupled interface between two components results in the need to change both components if a change is made to only one component. A decoupled interface eliminates this need, and both components can be changed without affecting the other. See Ulrich (1995, pp. 423-424) for more details and an illustrative example.

Göpfert (1998, pp. 107-111) expanded the concept of modular / integral architectures and investigated product architectures with regard to the two dimensions of components' functional independence and components' physical independence (see Figure 2.20). Beside the modular and integral product architectures on the diagonal, two further types are introduced. *Functional-modular product architectures* are defined by functionally independent components which are connected through physical interfaces that are difficult to separate. *Physical-modular product architectures,* on the other hand, consist of physically independent components (i.e. easily separable) that share strong functional dependencies. Such products can readily be disassembled into their components but only provide their functionality when their constituent components are connected.

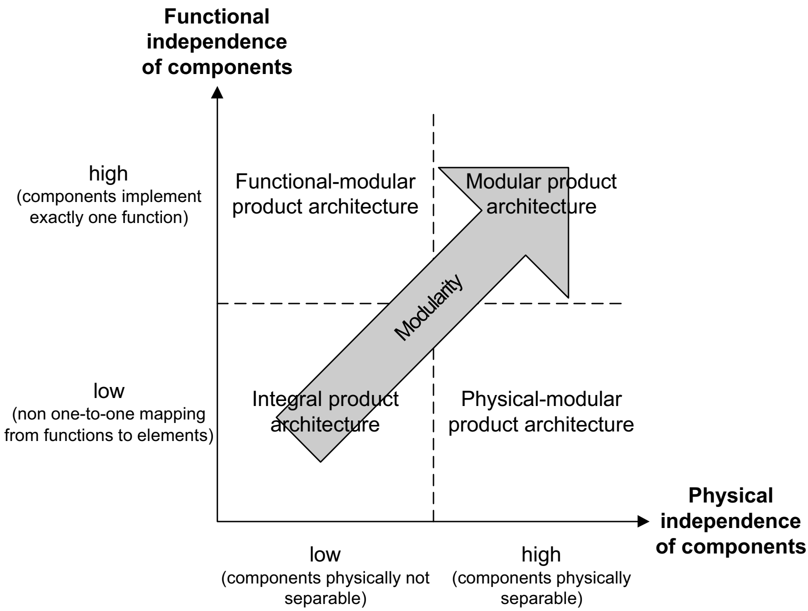


Figure 2.20 Classification of product architectures based on Göpfert (1998, p. 107)

Example: Cummins in India

Cummins, a producer of diesel engines and power generators, boasted a share of 60 percent of the high-horsepower end of the Indian market by 2000. It was a marginal player, though, in the large and fast growing low-horsepower (under 100-kilowatt) end. Each segment of the low-power end needed different features: farmers, for instance, asked for engines protected against dirt, while noise was a more important issue for hospitals. Cummins realized that it had to somehow offer an engine at a low price that would meet the needs of all these customers.

The solution was to develop a series of smaller, lower-powered, modularized engines and to combine them with add-ons called "gensets" (generation sets) that could be customized for different segments. The hospital version was supplemented with a noise abatement hood that was omitted for the farm kit, which came with dust and dirt guards not included in the hospital version. Customers liked the gensets because the product came tailor-made. Modularizing the product also helped solve operational problems. Ordinarily, customizing products means smaller manufacturing runs, which translates into increased average unit cost of production. But because the company was able to increase production runs of the common subsystems and components, overall costs were kept low.

Cummins' product strategy was based on the idea of combining manufacturing cost cuttings with listening to the needs of India's low-horsepower segment. Designing the product architecture in a modular way provided the key to the company's success. Cummins won 40 percent of the market, and genset sales accounted for 25 percent of the company's total power generation sales in India. Despite the much lower unit prices of the low end, its net profitability was comparable to that of the high end. In 2002, exports began to other parts of Asia and were later extended to Africa, Latin America, and the Middle East. Source: Brown and Hagel III (2005, pp. 40-42).

2.4 Concluding Remarks

Several concepts regarding product complexity and product architecture have been introduced so far in this chapter. Table A.1 in Appendix A summarizes these definitions and provides a number of additional terms that will be referred to in the remainder of this work. This chapter has shown that management decisions concerning product complexity and product architecture are an important influencing factor on a product's profitability. This work presents a model that attempts to give support in this field. First, the next chapter presents existing concepts and concludes that there is a lack of models investigating a product's complexity quantitatively and on a product architecture basis, while not neglecting strategic and market aspects.

3 Literature Review: Existing Concepts

It has been the motivation of many concepts and models in theory and industry practice alike to create and implement successful solutions to the challenging situation described in the previous chapter that most manufacturing companies find themselves in. This chapter presents a selection of existing concepts that have been established in literature and have found application in an industry setting. They are all, in a general sense, concerned with managing product complexity, and most of them consider product architecture in one way or another. Although the set of concepts chosen here reflects the current state of knowledge in the field, it does not mean to be complete as this would be beyond the scope of this work. Section 3.1 develops the criteria by which each concept is assessed. Sections 3.2 and 3.3 give a short overview of each concept, while the last section of this chapter summarizes the evaluation.

3.1 Assessment Criteria

The concepts presented in the following sections are assessed with a set of five criteria that seem important in the context of this work:

- *Company and product strategy.* To what extent does the concept include aspects of company and product strategy? I argue that when dealing with product complexity, it is utterly necessary to take into account the broader strategic setting of the product under investigation. Changes made to a product's variety or its architecture normally affect its perception with customers, and they commonly alter its production costs and entail design changes. Therefore, knowledge of strategic aspects is one important key to effectively managing complexity.

- *Market aspects.* How well are customer requirements integrated into the model? As we have seen in the previous chapter, market needs are the starting point for introducing complexity to a product. Thus, market aspects must be part of a concept concerned with complexity management.

- *Product architecture.* As this work focuses on product architecture as a means to manage product complexity, the concepts presented in this chapter are assessed in respect of to what extent they consider product architecture.

- *Quantification of complexity.* As will be shown in Chapter 4, it is a major feature of the model presented by this work to include a quantification of product complexity. In this way, decisions about product architecture are made on a quantitative and more objective basis. This, in turn, enhances the model's credibility and facilitates agreement among the managers in charge of architectural decisions. As a consequence, I evaluate the existing concepts presented in this chapter with regard to their quantitative nature.[1]

- *Applicability in practice.* How easily is the concept implemented in industry practice? Any model that intends to be relevant not only for the scientific community, but also for practitioners must balance scientific rigor with ease of application. The costs of applying the model must in any case be offset by the benefit it provides. Otherwise, it remains a theoretical construct irrelevant for management practice.

3.2 Managing Complexity on a Conceptual Level

This section introduces three concepts that – from a complexity management point of view – provide conceptual frameworks to cope with a company's complexity situation. These concepts provide a broader picture, mostly involving strategic and market aspects. They are not, however, concerned with the nuts and bolts of implementing complexity management. This is the domain of the concepts presented in Section 3.3.

---

[1] It must be said, though, that caution always has to be used when applying quantitative models. This prevents one from blindly following the advice given by such a model. In a management science context, qualitative aspects strongly prevail and, therefore, should be given their fair share of attention to balance quantitative with qualitative aspects.

3.2.1 Mass Customization

The term *mass customization* was coined by Davis (1987), who argued that mass production and customization are not necessarily opposites and can very well be combined. Davis explained that – similar to the dualism of light being composed of particles as well as made of waves – mass customization has the same role in business: accepting contradictions without trying to resolve them. According to Davis (1987, p. 169), “mass customization of markets means that the same large number of customers can be reached as in mass markets of the industrial economy, and simultaneously they can be treated individually as in the customized markets of pre-industrial economies.” Tseng and Jiao (2001, p. 685) define mass customization in a similar way as “producing goods and services to meet individual customer’s needs with near mass production efficiency.”

Mass customization represents a *hybrid competitive strategy* that overcomes the traditional hypothesis of Porter (1980) that an enterprise must either embrace a strategy of *cost leadership,* pursue a *differentiation* strategy, or follow a *focus* strategy to prosper. The postulate of cost leadership and differentiation being incompatible is a widely held opinion (Piller, 2003, p. 211). In contrast to that, mass customization presents a strategy that allows customizing a company’s products at a cost level approaching that of a mass producer.

In his work, Pine introduced five distinct methods to implement mass customization (see Pine, 1993a, pp. 171-212, and Pine, 1993b, pp. 7-13), each of which focuses on different stages of the value chain:

- *Customize services around standardized products and services.* A standardized product can be tailored by people in marketing and delivery before it reaches customers. For example, car rental companies add customized services such as express service and club memberships for frequent customers to its standard commodity service. The competitive advantage gained from this method is not very sustainable, which makes it necessary to consider other methods, such as the ones presented in the following.

- *Create customizable products and services.* This method involves producing goods that customers can easily adapt to individual needs in a "self-service" manner. It changes the focus of development and marketing, while production and delivery remain almost undisturbed. Office furniture that can be adjusted and computer applications that allow users to create their own system environment provide examples of this widely employed method.

- *Provide point-of-delivery customization.* As customers know best what they want, this method performs the final customizing step at the point of sale. Men's suits and eyeglasses are individualized to a customer's specific preferences right at the shop. If a firm moves a step further and shifts the entire production process to the point of delivery, this affects the entire organization – with all the difficulties associated with such a step. Therefore, the method discussed here is more appropriate for products having one inherently individual characteristic on an otherwise relatively standard commodity. In this way, the standard part can be manufactured centrally, while the customized characteristic can be produced at the point of sale.

- *Provide quick response throughout the value chain.* Reducing the time needed along a firm's entire value chain is known as time-based competition. Speeding up new product development and reducing set-up time in manufacturing significantly decreases complexity costs. Shortening the order-to-delivery cycle in marketing also lowers complexity costs by reducing final goods inventory.

- *Modularize components to customize end products and services.* The most effective method of mass customizing products is by creating modular components that can be configured to a large number of product variants. Economies of scale are achieved through the components, while economies of scope and customization are gained by reusing the components to create a large stream of product variants.[2] Subsection 3.3.5 provides further details of the concept of modularization.

---

[2] See Feitzinger and Lee (1997) for an insightful case study of how mass customization was achieved at Hewlett-Packard by introducing modular product and process design.

While the above methods are based on a firm's value chain, Gilmore and Pine (1997) defined four approaches to mass customization that are characterized by either changing or not changing the product and its representation (see Figure 3.1).

- *Collaborative customization* is most often associated with mass customization. Here, companies work closely together with customers to determine their individual needs. Both the product and its representation are customized. Finished products are only produced in response to actual customer needs, which saves costs by keeping finished goods inventory low.

- *Adaptive customization* offers a standard product that can be customized by the customers themselves, without any direct interaction with the company.[3]

- *Cosmetic customization* presents a standard product differently to different customers, i.e. a standard offering is packaged specially for each customer. This type is most appropriate when the standard product satisfies almost every customer and only the product's form needs to be adapted.

- *Transparent customization* provides a customized product without letting the customer know that the product has been customized. This type is best employed in businesses where customers do not want to be bothered with direct collaboration, such as delivering goods and services that are not part of the customer's core business.

Most authors writing about the concept of mass customization tend to define the state of mass production as the starting point for companies to work themselves towards mass customization and present ways to customize their products (while still attempting to maintain the cost level of mass produced goods). It is difficult to find contributions that describe the path to mass customization from the other extreme, i.e. showing ways to cut complexity costs of strongly customized products (e.g. by com-

[3] This type of mass customization is similar to the "create customizable products and services" method presented above.

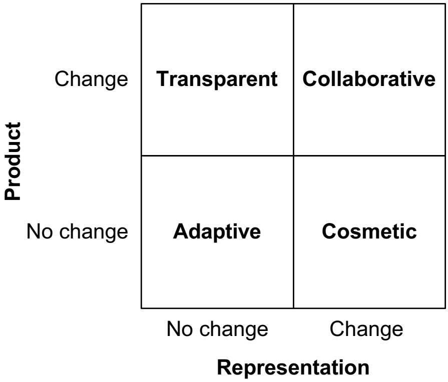


Figure 3.1 Four approaches to mass customization (Gilmore & Pine, 1997, p. 95)

ponent standardization). The focus of most research on mass customization lies on moderately increasing complexity to customize products while keeping a watchful eye on costs.

From a complexity management point of view, mass customization provides a conceptual framework to balance internal and external complexity and to optimize the trade-off between the two. The focus lies on strategic aspects and on the implementation of how to mass customize products, while product architecture and quantification of complexity are not considered at all. Mass customization does of course take into account market aspects as it attempts to customize products, but methods to gauge customer requirements are not presented. Table 3.3 at the end of this chapter shows a summarizing evaluation of mass customization with regard to the assessment criteria presented in Section 3.1.

Example: National Bicycle Industrial Company (NBIC)

In 1992, the National Bicycle Industrial Company (NBIC), a subsidiary of Matsushita, was Japan's second largest manufacturer of bicycles, producing roughly 700,000 units per year. It was a time when bicycle assemblers such as Giant, Trek, Cannondale and many others exerted fierce price pressure on manufacturers as demand for standard bicycles was sluggish. The price for high-end sporting bicycles, though, kept increasing because the market required customized solutions.

Among NBIC's three product lines, the high-end Panasonic brand accounted for approximately 20% of total production. The bulk of NBIC's revenues and profits, however, depended on the mass market. NBIC therefore embarked on a strategy quite unusual in industry: it pursued mass production and mass customization in parallel. While the mass-production factory produced 90% of the bicycles, the mass-custom factory operated under a special system named the Panasonic Ordering System (POS). Under POS, custom-made bicycles were able to be delivered within two weeks – at a price premium of 20 to 30% over the "standard" Panasonic bicycles produced in the mass-production factory. Customers can select options, colors, patterns and models from an estimated 8 million possible variations.

The mass-custom factory targets a smaller segment of the market by strongly differentiating its products. Meanwhile, NBIC managed to keep costs at an acceptable level through several measures, some of which are mentioned here. NBIC employed automated robots for painting that were previously only used in the mass-production facility; much of the software required to operate the CAM systems are shared across the mass-production and the mass-custom factories; the mass-custom factory is directly linked to customers, which ensures that a custom-made bicycle is produced only after the arrival of the customer's order, effectively canceling out the need for a finished goods inventory. By implementing the POS at the mass-custom factory, NBIC successfully attempted to increase its share of the high-end market, yielding an increased profit margin and catching its major competitors by surprise. Sources: Kotha (1995) and Kotha (1996).

3.2.2 Lean Management

We've become convinced that the principles of lean production can be applied equally in every industry across the globe and that the conversion to lean production will have a profound effect on human society – it will truly change the world.

James P. Womack, Daniel T. Jones, and Daniel Roos[4]

A five-year investigation from 1985 to 1990 called the International Motor Vehicle Program (IMVP) revealed that Japanese car factories boasted double the productivity as compared to their Western counterparts (Womack, Jones, & Roos, 1990, pp. 4-7, p. 13). This considerable difference in performance was credited to *lean production* – "lean" meaning increased productivity thanks to reduced engineering hours to develop a new product, less workforce and manufacturing space needed in the factory, increased quality (such as fewer defects), and less than half the inventory of traditional manufacturers. Lean thinking strives to eliminate waste – or *muda* in Japanese – at any rate and at any point in a product's progress from development to production to delivery. In short, a lean value chain translates to reduced complexity. Womack and Jones (2003) provided a guideline to achieve this objective by defining five *lean principles* a company must follow to become a lean enterprise:[5]

---

[4] Womack, Jones, & Roos (1990, pp. 7-8)

[5] Based on Toyota's success, Liker (2004) introduced fourteen management principles that are in some ways similar to the five lean principles presented here. The fourteen principles are organized into the following groups: philosophy (long-term thinking), process (eliminate waste), people and partners (respect, challenge, and grow them), and problem solving (continuous improvement and learning). Liker's (2004) principles two and three, for instance, are comparable to Womack and Jones' (2003) principles three ("flow") and four ("pull"). For further literature on Toyota and the foundations for its success, see Hino (2006).

- *Specify value.* The starting point for becoming lean is, in any case, the customer. The way to do this is to precisely define the capabilities offered by a specific product at a specific price at a specific time.

- *Identify the value stream.* When analyzing the value stream for each product, a considerable degree of *muda* can normally be exposed. These are all steps in the process that do not create any value. It is important to note that the value stream includes suppliers, also requiring the rethinking of firm-to-firm relations.

- *Flow.* Once the wasteful steps in a product's value stream have been scrapped, the value-creating steps must be made to flow. This involves discarding the conventional idea that activities must be grouped by type (e.g. in batches, functions, and departments) to be performed more efficiently and managed more easily.

- *Pull.* A lean enterprise can design, manufacture, and deliver exactly what the customer wants just when the customer wants it. Therefore, the customer pulls the product from the producer rather than companies pushing their products on the market according to some uncertain sales forecast.[6]

- *Perfection.* Offering a product means continuously improving customer value and the flow through the value stream, reducing mistakes and the time and cost needed.

Lean production is seen by its proponents as the latest paradigm in the world of production, succeeding craftsmanship and mass production (see Figure 3.2). Womack and Jones (1994) pointed out that lean production is merely one milestone on the journey to becoming a lean enterprise. The previously described rethinking of the value stream is no doubt one essential ingredient. However, the needs of three distinct groups must also be understood and satisfied: (1) employees (e.g. job security), (2) functional areas (they are a company's learning organizations accumulating knowl-

[6] One half of all books printed in the U.S. each year are shredded because they did not find a customer! (See Womack and Jones, 2003, p. 25.)

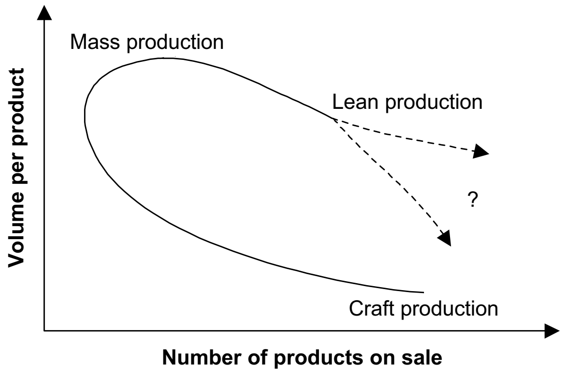


Figure 3.2 The progression of product variety and production volume depending on the prevailing production paradigm; source: Womack, Jones, and Roos (1990, p. 126)

edge and, therefore, need a secure place), and (3) other companies involved in the value stream.[7]

Firms implementing a lean philosophy are inherently involved with managing complexity. The process of eliminating waste while providing the demanded level of customer value attempts to establish a new and optimized balance between internal and external complexity. When assessing lean management with regard to the criteria introduced in Section 3.1, it is evident that strategic and market aspects are taken into account to a considerable degree as these are the first issues to think about when implementing lean production. Product architecture is not considered by the lean management concept, nor is a quantification of product complexity. Table 3.3 at the end of this chapter shows a summarizing evaluation of lean management with regard to the assessment criteria of Section 3.1.

---

[7] See Womack and Jones (1994, pp. 94-96) for details on the three needs.

3.2.3 The Concept of Optimum Variety

In his work on managing product variety, Rathnow (1993) introduced a concept to optimize complexity based on the following three steps (see Figure 3.3):

- *Optimize product offering.* In order to find ways to increase customer benefit, the appropriate level of product variety is determined in this first step. Customer requirements are gathered and, based on this market analysis, the product portfolio breadth (i.e. the product variety offered on the market) is defined.

- *Optimize structure.* This step considers how the level of variety determined in the previous step can be handled by the enterprise. The following aspects are sought to be optimized with regard to their complexity: variety of inputs (raw material, suppliers, modules etc.), fabrication technologies, organizational complexity (processes, organizational interfaces, rules), human resources (know-how, competences, cultural diversity), and variety of outputs (product portfolio, product variants, functionality, quality).

- *Overall optimization.* The last step combines the previous two steps to achieve an optimized solution, i.e. to determine the optimum variety. The interdependencies between optimizing the product offering and the enterprise-internal structure are considered, and the constraints imposed by the firm’s environment and its internal structure are also taken into account.

The concept of optimum variety integrates the internal and external dimensions of complexity in a very clear way and combines them in a very powerful conceptual framework for managing complexity. While strategic and especially market aspects are covered by the methodology, it lacks information on the actual implementation. Hints on how the optimum variety can be created in an industry setting – be it by optimizing product architecture or any other driving factor of complexity – are not provided. Table 3.3 at the end of this chapter shows a summarizing evaluation of the optimum variety concept with regard to the assessment criteria presented in Section 3.1.

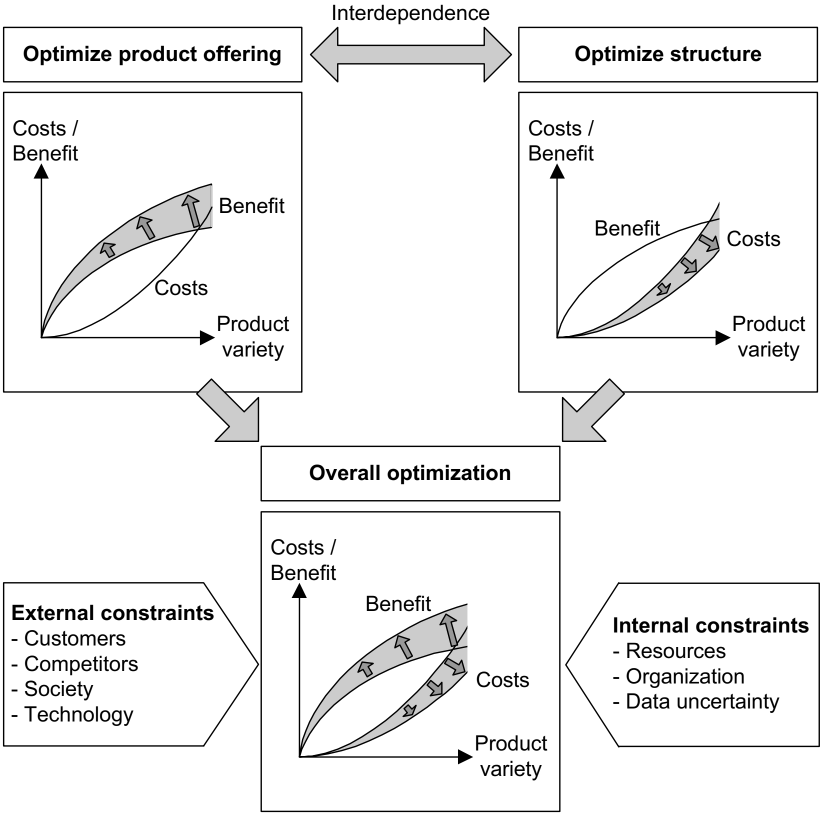


Figure 3.3 Determining the optimum variety[8]

3.3 Tools for Managing Complexity

While the previous section has shown how complexity management can be tackled on a conceptual level, this section presents actual tools that go into the details of optimizing complexity within products.

[8] Figure 3.3 is based on Rathnow (1993, p. 42) and Matern (2000, p. 20).

3.3.1 Quality Function Deployment (QFD)

The roots of quality function deployment (QFD) go back to the late 1960s in Japan, and its basic ideas and issues were first published in the 1970s.[9] It is claimed to have helped Toyota to cut its development time and costs by 40%, and more recently many American firms have adopted QFD, while there has been relatively little application in European firms (Tidd, Bessant, & Pavitt, 2001, p. 169). The aim of QFD is to facilitate the complex task of translating customers' needs and wants into technical product specifications. To this end, it draws on a tool called the quality function deployment matrix, often referred to as *"the house of quality."* Such a matrix is outlined in Figure 3.4, where it can be seen that a typical QFD matrix contains customer information in the horizontal part, while the vertical part reveals technical information (Johnson, 2003, p. 104). Building the house of quality evolves in ten steps:

- *Specify customer requirements.* The first step in building the house of quality consists of answering the question of what customers want. These needs are typically expressed as phrases customers use to describe products and product characteristics, such as "easy to close," and "stays open on a hill" for a car door (Hauser & Clausing, 1988, p. 65). For more complex products, customer requirements can be hierarchically structured into several levels. (See step 1 in Figure 3.4.)

- *Introduce weightings for customer requirements.* Customer requirements usually are not equally important. Therefore, they can be weighted by any scheme most suitable to the application context. (See step 2 in Figure 3.4.)

- *Competitor assessment from a customer perspective.* Comparison between the company's own product and the competitors' enables the strategic positioning of the product and reveals opportunities for improvement. (See step 3 in Figure 3.4.)

---

[9] See Akao (1990) for an overview of QFD's history.

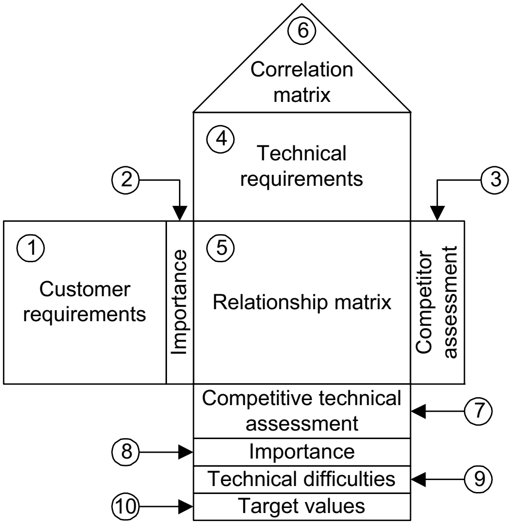


Figure 3.4 Quality function deployment matrix, or "house of quality"; derived from Johnson (2003, p. 104)

- *Specify technical requirements.* Now that the marketing domain has identified what the product has to do, the engineering domain specifies how to do it. The design team identifies those technical product characteristics that are likely to affect one or more of the customer requirements and lists them at the top of the QFD matrix. (See step 4 in Figure 3.4.)

- *Fill the relationship matrix.* This important step links the voice of the customer (i.e. customer requirements) with the language of the engineer (i.e. technical requirements). It forms the basis for extensive discussions between all functional areas involved (R&D, marketing, sales, and production) and makes the interdependencies between customer and technical requirements transparent (Seghezzi, 2003, p. 321). Any system can be employed to rank the strength of the relationships, such as numbers, different symbols, etc. (See step 5 in Figure 3.4.)

- *Establish correlations between technical requirements.* The house of quality's roof matrix helps to specify the various engineering features that have to be considered simultaneously (e.g. increasing speed means increasing engine power or lowering air drag) and facilitates necessary engineering trade-offs. (See step 6 in Figure 3.4.)

- *Competitor assessment from a technical perspective.* As opposed to step 3, where the comparison between the company's own product and the competition is made from a market perspective, the technical requirements are considered here. (See step 7 in Figure 3.4.)

- *Introduce weightings for technical requirements.* Similar to step 2, relative weights are often assigned to the technical requirements. (See step 8 in Figure 3.4.)

- *Indicate technical difficulty.* A row can be added indicating the technical difficulty, showing in engineers' terms how hard it is to achieve the technical requirements. (See step 9 in Figure 3.4.)

- *Specify target values.* Based on the previous comparisons between the company's own and competitors' products, target values for the technical requirements are established. (See step 10 in Figure 3.4.)

The QFD matrix for an automobile's outside mirror is shown in Figure 3.5, outlining the ten-step procedure introduced above. As there is no standard QFD matrix, most design teams would custom-build their houses of quality to their specific needs. For instance, teams may add other columns for histories of customer complaints and the cost of servicing those complaints. Some applications add data from the sales force to the customer requirements list to represent strategic marketing decisions. When cost cutting is a goal, the design team can set priorities for improving components by comparing weighted characteristics to actual component cost (Hauser & Clausing, 1988, pp. 67-68).

The "hows" of the house of quality (the technical requirements) can become the "whats" of another house, one mainly concerned with detailed product design. The wind noise column in Figure 3.5 can be taken and made the row in a parts deployment house, while parts characteristics – such as surface properties of the mirror housing – become the columns. This procedure can be advanced further to a third and a fourth

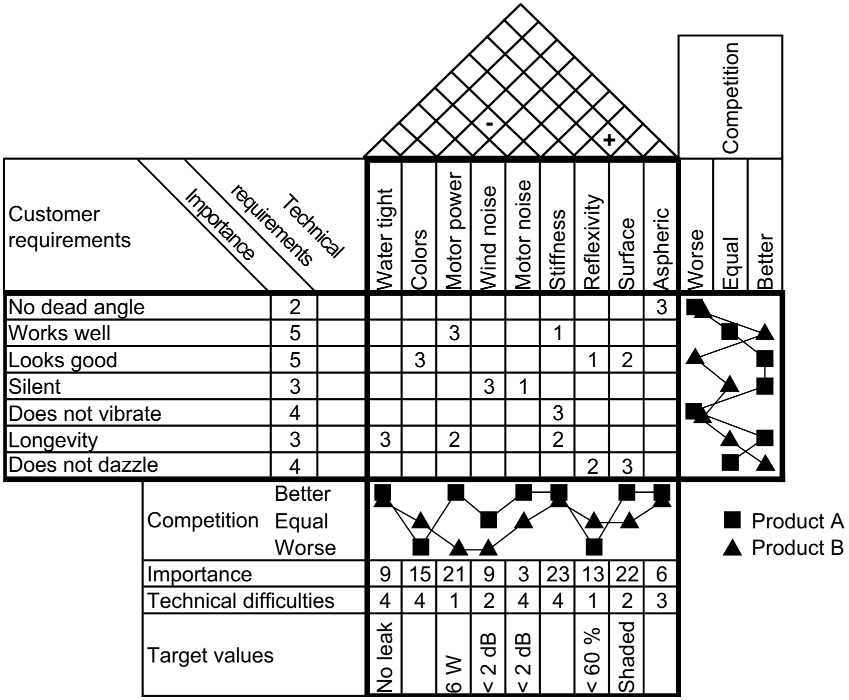


Figure 3.5 House of quality for an automobile outside mirror (Seghezzi, 2003, p. 322)

house, commonly referred to as the process planning and production planning houses. The columns of each house become the rows of the next. This sequence of houses has been called *cascade of QFD charts* (Govers, 1996, p. 577) and is shown in Figure 3.6.

The strength of QFD lies in its requirement to involve an interdisciplinary team of people forced to establish a consensus opinion on how customers' requirements can best be presented (Burn, 1990, pp. 80-81). However, going through the QFD procedure, which eventually leads to the house of quality, is a demanding task (Seghezzi, 2003, p. 323). Tidd et al. (2001, p. 169) pointed out the practical problems of implementing QFD and reiterated the fact that the compilation of a lot of marketing and technical data is required. A study by Griffin (1992) on American industry found that only about 20% of the projects investigated have resulted in any quantifiable benefit.

Yet, Griffin reported that QFD has the potential to improve the development climate in the long run and acknowledged that it can provide significant intangible benefits, such as reducing cross-functional barriers and facilitating changes in corporate culture.

QFD provides a well structured tool to collect and quantitatively assess customer requirements. However, strategic aspects and product architecture are not a focus of QFD, and quantification of product complexity is not considered at all. Table 3.3 at the end of this chapter shows a summarizing evaluation of QFD with regard to the assessment criteria presented in Section 3.1.

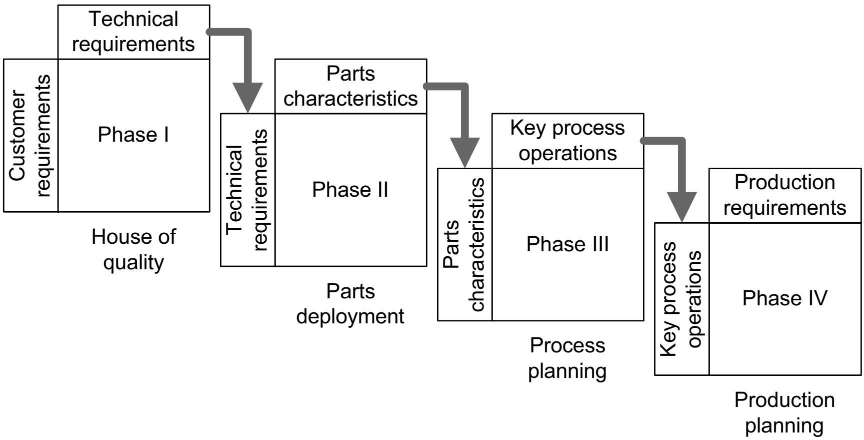


Figure 3.6 Cascade of QFD charts; derived from Govers (1996, p. 577), and Hauser and Clausing (1988, p. 73)

3.3.2 Target Costing

It's got to sell for x. Let's work backwards to make sure we can achieve it.

Ford S. Worthy[10]

The above quotation describes the aim of target costing in a nutshell: while the traditional approach of cost-based pricing is increasingly considered a relic of the past, price-based costing (or target costing) is emerging as a key strategic tool (Shank & Fisher, 1999, p. 73). As can be seen from Figure 3.7, the procedure outlined by target costing starts with the determination of the *target price,* which is the price that the company believes the market will accept. The *allowable cost* is then calculated by subtracting the *target profit,* a value that reflects the company's strategic plans and financial projections. The allowable cost normally is far below what realistically can be attained. Therefore, the standard cost based on current technologies and practices is calculated, i.e. the standard cost achievable without innovation. Finally, management establishes a *target cost* that lies between the allowable and the standard cost[11] (Hiromoto, 1988, p. 24).

The target costing process described above represents the "pure" form of target costing and is called *market into company,* as target costs are exclusively based on market prices. Apart from the market-into-company approach, Seidenschwarz (1991, pp. 199-200) reported from his study of literature and industry practice of target costing that four additional approaches to determine target costs are used: *out of company* (target costs based on the company's existing design know-how and production

[10] Worthy (1991, p. 74)

[11] According to Seidenschwarz (1991, p. 200), the most common method is to define the target cost as the average of standard and allowable cost. Seidenschwarz points out, however, that target costs must be determined by considering company strategy and competitive intensity. When pursuing a cost leadership strategy, target costs can even be defined as identical to the allowable costs.

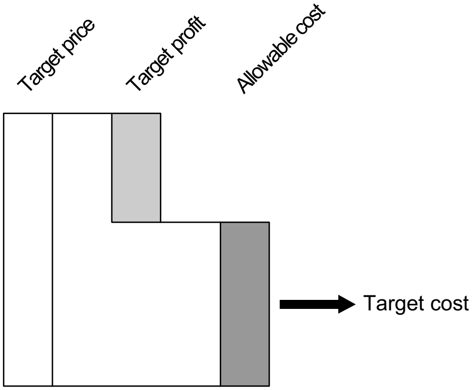


Figure 3.7 Determining the target cost

abilities); *into and out of company* (a combination of the two former types); *out of competitor* (target costs estimated from the competitors' costs); *out of standard costs* (target costs based on standard costs, less a certain deduction). Seiler (1998, p. 366) stated that the into and out of company approach is most commonly used in practice. Often, these companies are at risk to believe they are determining costs in a market-oriented way, while in fact they give priority to internal costs.

Once consensus on the target cost of a product has been achieved, the next step in the target costing process deploys the target cost to the product's individual components. Tanaka (1989) introduced such a methodology that bases the cost deployment on the product's functions. Functions are either defined as "hard" (mechanical functions) or "soft" (convenience and value functions) and are evaluated by their degree of relative importance. In this way, a percentage is assigned to each function, the sum of all functions being 100%. The designers then create trial products that satisfy the target cost as far as possible. In a next step, the functions are deployed to the components, which, again, leads to a percentage assigned to each component reflecting its relative importance (column to the far right in Table 3.1). For each column in Table 3.1, the shaded cells add up to 100% as they represent the splitting of every function to the

Table 3.1 Deploying functions to components; based on Tanaka (1989, pp. 62-63)

		Functions				
		Marking	Ink maintenance	Ink guidance		Total
Importance (%) / Components		16.2	13.6	12.5		100%
C1	Ink	35 / 5.7	40 / 5.4	33 / 4.1		17.3
C2	Nib	35 / 5.7	60 / 8.2	33 / 4.1		18.3
C3	Pen ring	10 / 1.6		10 / 1.3		10.9
C9	Cap					3.9

components. Multiplying the shaded cell with the respective function importance leads to the value in the adjacent cell.[12]

A component's *value index* can now readily be calculated as the ratio of its relative importance and its percentage of product cost. According to Horváth and Seidenschwarz (1992, p. 147), the value index reveals whether a function is implemented in a too expensive way (value index < 1) or too cheaply (value index > 1). Ideally, a component's cost should reflect its contribution to the overall product's functionality. This means that major components can cost more, while less important ones should be designed at a low cost. In this ideal case, the value index is equal to unity. When all components have been evaluated with respect to their relative functional importance and their relative cost contribution, they can be assembled in a diagram referred to as

12 For instance, the function "marking" is fulfilled to a degree of 35% by the ink. As "marking" contributes 16.2% to the product's overall functionality, a relative importance of 5.7% (= 16.2% × 35%) is assigned to the ink. Adding all numbers in the white cells of the ink row (not all of which are shown in Table 3.1) equals 17.3%. This is the relative contribution of the ink to the product's functionality.

the *value control chart* (see Figure 3.8). Because the chart's diagonal represents the ideal case (value index = 1), a component's vertical distance from the diagonal visualizes the *target gap,* i.e. the need to reduce costs. In practical applications it suffices to ensure that components are located within the *optimal value zone.* If components are located above that zone (to the northwest), cost reductions should be made to bring the value index within the zone. For components to the southeast of the zone, cost increases may become necessary to ensure that the product performs its functions satisfactorily. Kaiser (1995, pp. 135-136) pointed out that the value control chart contains relative (and not absolute) values. Therefore, when the cost of component C9 in Figure 3.8 is reduced to bring it into the optimal value zone, all the other components wander upwards. Components to the southeast of the optimal value zone can hence jump into the zone without any changes (e.g. cost increases) made to those components.

Target costing is a highly market-oriented approach to control costs in an early stage of product development. The process of evaluating a product's functions from a customer perspective all the way to deriving the value control chart requires a cross-functional and interdisciplinary team. Guidelines as to how to reduce component costs are not provided, however. Kaiser (1995, p. 133) even argued that target costing does

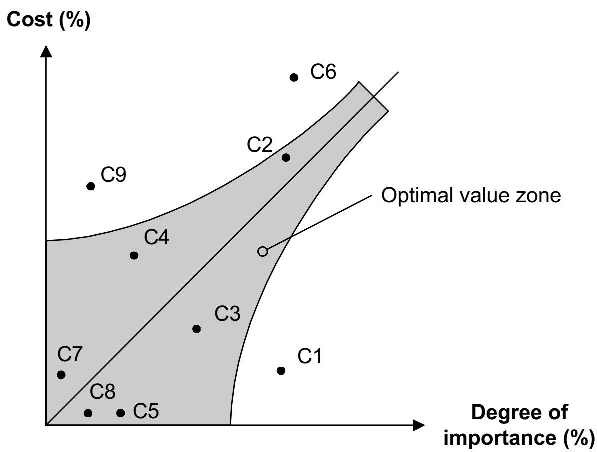


Figure 3.8 Value control chart; based on Tanaka (1989, p. 68)

not give any hints as to how it can be implemented in industry practice. Nevertheless, due to its undisputable strengths, target costing is a widely employed and well established method to handle the complexity in product development. Table 3.3 at the end of this chapter shows a summarizing evaluation of target costing with regard to the assessment criteria presented in Section 3.1.

3.3.3 Design for Variety

The design for variety (DFV) concept presented by Martin and Ishii (1996) provides a means to estimate the costs incurred by introducing variety into a product line. As described in Chapter 2 of this work, these costs are commonly indirect and are often not thoroughly understood because they are difficult to quantify. DFV attempts to capture these indirect variety costs by defining three indices:

- The *commonality index* is a measure of the percentage of parts that are reused for other product models and accounts for the utilization of standardized parts.[13]
- The *differentiation point index* considers the points along the value chain where variety is introduced. It is based on the generally agreed premise that (ceteris paribus) the later variety occurs, the better.[14]
- The *setup cost index* relates the estimated setup costs to the overall product costs (material, labor, and overhead).

As a next step, Martin and Ishii (1997) proposed the *process sequence graph,* which shows the flow of the product through the manufacturing and assembly lines and visualizes its differentiation points. A quantitative algorithm then shifts those

---

[13] The commonality index used in DFV is based on the work of Collier (1981), who introduced the degree of commonality index, an analytical measure to determine the effect of the degree of commonality on total cost, inventory cost, delivery performance, etc.

[14] The differentiation point considered by Martin and Ishii (1996) is somewhat similar to the order penetration point (OPP) introduced in Chapter 2, but not entirely the same. While there are several differentiation points within the value chain (every time variety is added to the product), there is only one OPP (the point where a specific customer order enters the configuration of a product).

components causing variety as far back in the manufacturing and assembly process as possible. The differentiation points in the process sequence graph are called nodes. The algorithm performs the optimization by minimizing the number of nodes in the process sequence graph of a product.

The methodology provided by DFV is primarily concerned with quantifying the costs incurred by product variety and deriving strategies on how to reduce those costs by optimizing the manufacturing and assembly sequence. It gives valuable support in describing complexity with key figures. While the focus is driven by technical issues, market and strategic aspects are excluded by the methodology. Table 3.3 at the end of this chapter shows a summarizing evaluation of design for variety with regard to the assessment criteria presented in Section 3.1.

3.3.4 Design for Configuration

Design for configuration (DFC) is a methodology that supports designers in producing and managing information and knowledge needed to configure products (Pulkkinen, Lehtonen, & Riitahuhta, 1999, p. 1497). A fixed set of variants can be derived from a *configurable product,* which can be formed from a fixed set of modules, components, and add-ons with a given variety. Riitahuhta (2001, p. 2) named the creation of a particular variant the *configuration task.* The objective of DFC boils down to offering a relatively broad product portfolio while limiting costs due to the absence of customer-specific designs. Thus, it is able to combine several virtues of mass production and customization (Bongulielmi, 2003, pp. 51-52), making it a viable tool for implementing mass customization. According to Pulkkinen et al. (1999, p. 1496), a configurable product is characterized by the following properties:

- Each product variant can be clearly specified as a combination of pre-designed components and / or modules.
- There is a pre-designed product architecture which meets a given range of customer requirements.

- The sales process does not entail the design of new components. It only requires the systematic configuration of product variants.

- As all variants are based on the same common architecture, they are considered a product family.

From a complexity management point of view, DFC is an approach mainly concerned with designing a product's architecture to allow for a cost-effective configuring of product variants. The customer requirements the product family is supposed to cover must be defined before the design process. Therefore, they should be well understood, rendering the consideration of market aspects a fairly important task in the DFC process. The strategic background is not investigated by DFC, nor does it provide a framework to quantify product complexity. Table 3.3 at the end of this chapter shows a summarizing evaluation of the design for configuration concept with regard to the assessment criteria presented in Section 3.1.

3.3.5 Product Modularization

There has been growing interest in the concept of modularizing products as a means to tackle product complexity. Section 2.3 gave a very brief introduction to modular and integral product architectures and defined a modular product as consisting of a number of relatively independent units (the modules) sharing decoupled interfaces. As these interfaces are clearly defined and highly standardized, the independently designed modules still function as an integrated whole (Balwin & Clark, 1997, p. 86). Product modularization is supposed to speed up the development process, enhance the ability to adapt to changes in the environment and reduce the cost of making changes because it increases a company's flexibility by minimizing the interdependencies between the modules of a product (Thomke & Reinertsen, 1998, pp. 24-27). Furthermore, Pine II (1993a, pp. 196-212) credited modularity with enabling the customer to choose from a large variety of products while letting the producer profit from economies of scale (shared components) and economies of scope (using modules in different products).

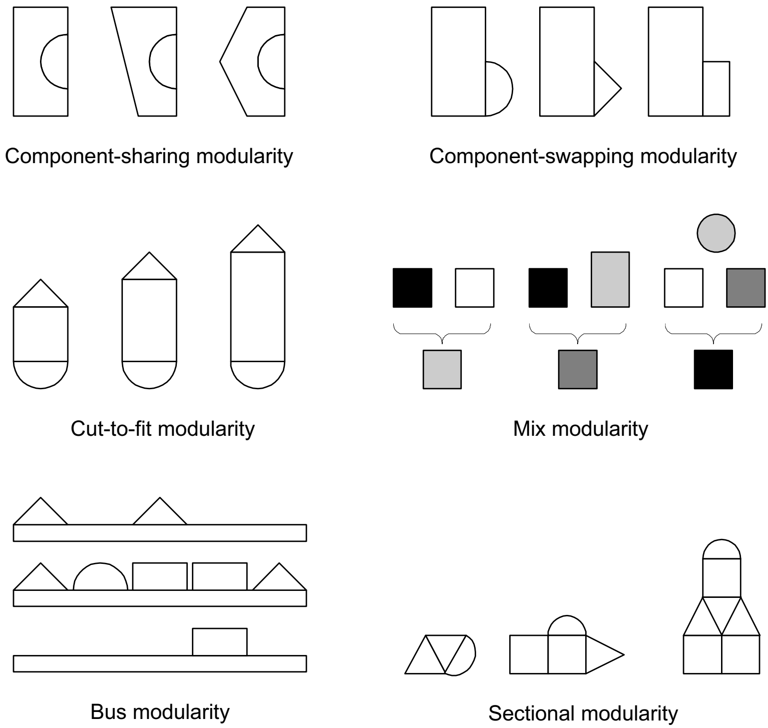


Figure 3.9 Types of modularity; source: Pine II (1993a, p. 201)

A widely employed typology of modularity was introduced by Pine II (1993a, pp. 200-211), who based his classification on the work of Ulrich and Tung (1991, pp. 77-78). The six types of modularity can be described as follows (see Figure 3.9):

- In *component-sharing modularity,* the same component is used across multiple products to provide economies of scope. It is often associated with the idea of component standardization.

- *Component-swapping modularity* is the complementary case to component-sharing modularity. Here, different components are combined with the same basic product to create a number of product variants belonging to the same product family. Component-swapping modularity is often associated with product variety as perceived by the customer.[15]

- *Cut-to-fit modularity* is the use of standard components with one or more continually variable components. Mostly, the variation is expressed as physical dimensions that can be modified (e.g. length, power).[16]

- *Mix modularity* can use any of the above three types, with the distinction that the resulting product is something different than the constituent components that are mixed together. Therefore, it can only be applied to products consisting of a mixture of various substances, such as colors or fertilizers. For instance, an endless stream of distinct colors can be produced by mixing only a limited number of basic colors.[17]

- *Bus modularity* relies on a standard structure with two or more interfaces that can attach any selection of components from a set of component types. While bus modularity allows variation in the number and location of the components, component-swapping, component-sharing, and cut-to-fit modularity only allow variation in the type of component used in an otherwise identical product architecture.

- *Sectional modularity* provides the largest degree of variety and customization. It allows connecting components in any arbitrary way, as long as each component is connected to another through standard interfaces. In this type, the product's scope is not predefined and can be changed to the specific needs of the situation. The

---

[15] Note that the distinction between component-sharing and component-swapping modularity is a matter of degree. One is inclined to ask whether the basic elements of a product are the components shared across all product variants (component-sharing modularity), or whether they are the basic product supplemented with additional components providing the variety (component-swapping modularity). The difference between swapping and sharing lies in how the basic product and the components are defined.

[16] Ulrich and Tung (1991, p. 78) used the term "fabricate-to-fit" for this modularity type as their work focused on manufacturers. Pine II (1993a) also considered process and service industry, hence "cut-to-fit."

[17] Mix modularity is not included in the work of Ulrich and Tung (1991) as they focused on manufacturers and excluded process industry.

classic example is Lego building blocks, from which an infinite number of objects can be built.

In his 1995 study, Ulrich defined three types of modular product architecture: slot, bus, and sectional architecture. While the bus and sectional architectures are identical to the two types discussed above, a slot architecture is characterized by interfaces between components that are each of a different type. The various components in the product therefore cannot be interchanged. Though slot modularity is similar to component-sharing modularity, they are not entirely the same: while component-sharing modularity focuses on one component that is shared across products, slot modularity does not limit the number of components considered.

As pointed out in Section 2.3, modularity is a continuous and not a discrete product property. A product is not either modular or not modular but can have various degrees of modularity. An approach to measure the modularity embedded in product architectures was presented by Mikkola (2006), who presented a quantitative calculation of what she called the *modularization function.* The function is based on such architectural characteristics as the number of components, the interfaces between components, the degree of coupling between the components, and the substitutability. In her model, a distinction is made between standard components and those that are new to the firm. Rapp (1999, pp. 44-47) introduced a set of two key figures that both gauge a product's degree of modularity. While the *interdependence index* quantifies the dependencies between components, the *degree of integration* measures to what extent the functionality is concentrated in the individual components.

When a firm modularizes its products, it is able to respond to the external complexity while keeping the internal complexity within reasonable limits. Thanks to widely standardized interfaces, a limited number of standard and customized modules can be combined in many different ways to form a stream of distinct product variants. A broad product portfolio can therefore be maintained that does not cause excessive costs to the enterprise. As the individual modules are highly independent from each other, changes made to one module do not affect other modules, which also saves costs.

When a product consists of components that go through life cycles of different lengths, modularization allows for decoupling the life cycles of these modules. In Baldwin and Clark's (1997, p. 87) opinion, modularity pays off the most when the manufacturing process and the design responsibility is delegated to many separate suppliers – the module makers. In this way, the assembler gains flexibility and cuts costs.

Besides these undisputed advantages of modularization, caution should also be used when developing modular products. Development costs and direct costs per unit (due to over-dimensioning) generally increase (Rapp, 1999, p. 60). Pine II (1993a, pp. 211-212) pointed out that this is true only for a single product (or a close-knit product family). The greater the number of distinct products, the greater the cost and performance advantage of modularity. According to Baldwin and Clark (1997, p. 86), modular products are much more difficult to design than comparable interconnected products because the visible design rules[18] have to be specified in advance to make the modules function as a whole. Other sources of concern regarding modularizing products stem from the easier reverse-engineering of modular designs and the less innovative solutions than if the development process had encompassed a greater scope (Pine, 1993a, p. 212).

Depending on the context of application, the scope of product modularization varies. While marketers concentrate on target markets' needs and on the implications for segment-specific modules, technical personnel focus on the architectural aspects of modular products, such as interface layout and mapping from customer requirements to physical modules. Nevertheless, the primary strength of modularization lies in its ability to provide a solution to growing complexity by rethinking the product architecture. Market and strategic considerations are part of the modularization concept as well, but are clearly not at the focal point. Table 3.3 at the end of this chapter shows a

[18] Baldwin and Clark (1997, p. 86) stated that designers divide information about a modular product into visible design rules and hidden design parameters. Visible design rules are decisions that affect subsequent design decisions. They fall into three categories (architecture, interfaces, and standards) and should be established in an early design stage. The hidden design parameters are decisions that do not affect the design beyond the local module and can, therefore, be chosen late.

summarizing evaluation of product modularization with regard to the assessment criteria presented in Section 3.1.

3.3.6 Modular Function Deployment

Modular function deployment (MFD) is a method presented by Erixon (1998) that supports the development of modular products. It is based on the concept of *module drivers,* which are supposed to describe the main criteria of modularization. The heart of the method, the *module indication matrix* (MIM), examines the function carriers (e.g. components) with respect to their aptness to form a module. MFD consists of the following five steps (see Figure 3.10):

- *Clarify customer requirements.* The first step involves the identification of requirements from a market perspective and draws on the procedure outlined by quality function deployment introduced earlier in this chapter.

- *Select technical solutions.* The above requirements have a strong customer focus and must be translated into a more technical product specification in order to proceed with product design. This is achieved by decomposing customer requirements into functions and sub-functions. The functions-based view, in turn, is translated into technical solutions.

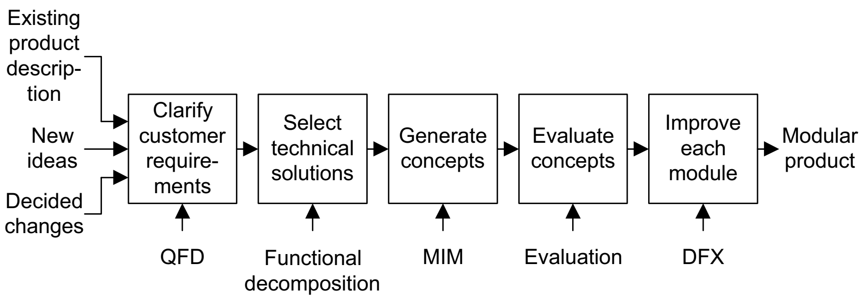


Figure 3.10 Modular function deployment; source: Erixon (1998, p. 66)

- *Generate concepts.* The sub-functions derived in the previous step are now evaluated with regard to twelve module drivers, shown in the first two columns of the module indication matrix (see Table 3.2). Depending on the specific application context, these module drivers can be supplemented with additional ones, such as strategy, financial limitations, etc. The MIM provides advice in two areas. First, an indication is given of which sub-functions should form a module. The higher a sub-function is weighted with respect to the module drivers (i.e. the stronger a sub-function is dependent on the module drivers), the more interesting it is as a module candidate. This is shown in the last two rows of Table 3.2. Second, the MIM points out which sub-functions can be integrated into one single module. By following the markings in the MIM horizontally, it becomes obvious which sub-functions share similar module drivers (in which case integration should be investigated) and which ones are dependent on contradictory module drivers. In the latter case, integration should be avoided.

- *Evaluate concepts.* Step four assesses the remaining concepts based on a set of metrics and rules considering development, assembly, and sales / after sales.[19] The module evaluation chart assists the evaluation process.

- *Improve each module.* The last step's goal is to enhance the technical details of the modules. DFX[20] methods such as design for assembly (DFA) and design for manufacturing (DFM) are employed.

The procedure outlined above facilitates architectural decisions about grouping a product into modules and, by employing QFD as an integral part of the methodology, manages to integrate a thorough understanding of customer needs. In step four (concept evaluation), it also provides some quantitative measures of product complexity.

---

[19] Erixon (1998, pp. 83-103) gives an overview of the metrics and rules that are applied in this step.

[20] DFX stands for "design for X" and includes a growing number of approaches to enhance product design, such as design for variety, design for assembly, etc. The X either denotes a certain phase in the product's value chain (manufacturing, assembly, transport, testing, etc.), or any product characteristic (cost, quality, time, flexibility, variety, etc.).

Table 3.2 Module indication matrix for a vacuum cleaner; source: Erixon (1998, p. 108)

		Fan	Noise absorbent, fan	Electric motor	Damper	Noise absorbent, motor	Chassis	Bag	Filter		Brake + knob
Design and develop-ment	Carry-over	●		●							⊘
	Technology push							●	●		
	Product planning										
Variance	Technical specification	○	○	○					○		
	Styling										●
Manufac-turing	Common units	⊘	⊘	⊘	●	●	●	●	⊘		⊘
	Process / Organization	●		●			●	●			
Quality	Separate testing			●							
Purchase	Black box engineering								●		
After sales	Service / maintenance			⊘					○		
	Upgrading								●		
	Recycling			●			●				
Company specific											
●= 9 ⊘= 3	Weight of driver (vertically summarized)	22	4	43	9	9	27	27	32		15
○= 1	Module candidates	✓		✓			✓	✓	✓		

Modular function deployment does not, however, consider aspects of product and interface variety, nor does it mirror its recommendations with company and product strategy. Table 3.3 at the end of this chapter shows a summarizing evaluation of modular function deployment with regard to the assessment criteria introduced in Section 3.1.

3.3.7 Product Platforms

An increasingly popular method to reduce complexity in products is the *product platform,* which essentially divides the product architecture into a standardized part (the platform) and customized modules. Combining the two allows the creation of a large number of distinct product variants (see Figure 3.11). The underlying rationale is to optimize the trade-off between cost savings (through scale economies) and competitive edge (through differentiation). Boutellier, Dinger, and Lee (1997, p. 61) argued that a platform does not necessarily spread across an entire product, as many authors require it to do. In highly modularized products, it can be advantageous to establish platforms on the level of individual modules, indicated by the platform modules in Figure 3.11.

Robertson and Ulrich (1998, p. 20) defined a product platform in a more general way as "the collection of assets that are shared by a set of products," not confining it to the common physical structure shared across products. These assets fall into one of the following four categories: *components, processes, knowledge,* and *people / relationships.* In Meyer and Lehnerd's (1997, p. 39) view, "a product platform is a *set of subsystems and interfaces that form a common structure* from which a stream of derivative products can be efficiently developed and produced." In a research context

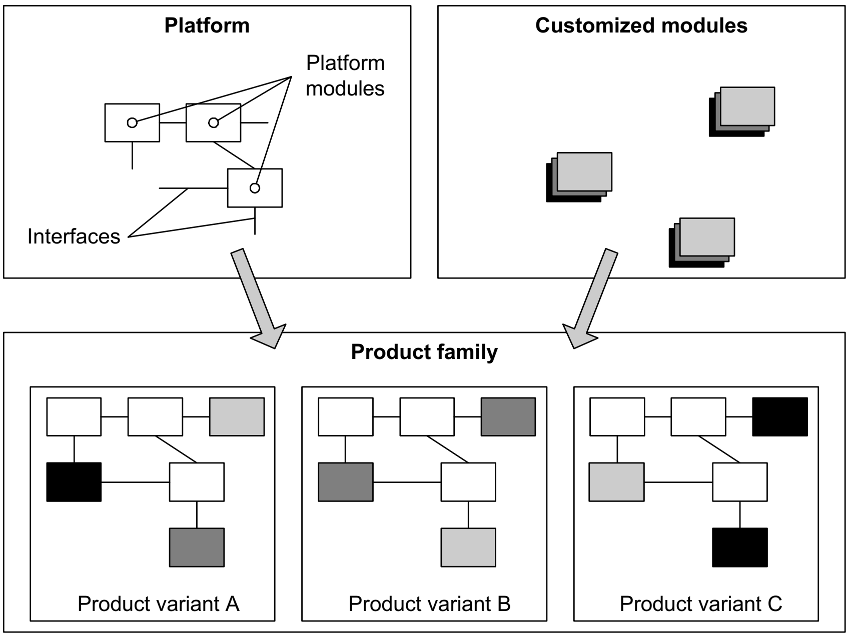


Figure 3.11 Product family derived from a product platform

investigating product architecture, the latter definition (which has the decidedly narrower scope) proves to be more useful.

Schuh and Schwenk (2001, p. 86) viewed the product platform as a special case of product modularization. Hofer (2001, p. 37) emphasized the point that the focus of modularization is decomposing a product into modules, while establishing a platform means structuring the product's architecture according to a certain hierarchy. A second differentiation is made here: product platforms must be kept apart from the effort to standardize parts across products. *Component sharing* as presented by Fisher, Ramdas, and Ulrich (1999) – no doubt a very powerful and successful strategy widely adhered to in industry – leads to a set of shared components, but such a collection of components is generally not considered a product platform (Robertson & Ulrich, 1998, p. 20). According to Rapp (1999, p. 73), component sharing is mostly performed on a product level (at best across similar products), but the scope of product platforms includes entire product families, not merely individual products.

A dizzying number of authors have published their views on how to implement product platforms. Two procedures that stand out of the crowd and are well established in literature are presented here. Robertson and Ulrich (1998, pp. 23-29) proposed a process that is based on three information management tools: the product plan, the differentiation plan, and the commonality plan.

- The *product plan* reflects the company's product strategy and usually comes from the overall product plan. It identifies the collection of products encompassed by the platform and specifies the distinct market offerings over time. While the product plan indicates major models, it does not show every variant and option.
- In the *differentiation plan,* all target values of the differentiating attributes[21] for each product in the plan are specified. It should represent the product's differentia-

[21] According to Robertson and Ulrich's (1998, p. 24) definition, differentiating attributes are the dimensions of the product that are meaningful to customers, such as "styling of instrument panel," or "color and textures" for an automobile.

tion for maximum appeal to customers in the target segment. For example, the differentiating attribute "engine power" for an automobile would have target values of 150 kW for the sports segment and 100 kW for the family segment.

- The *commonality plan* lists the number of components shared across the products in the plan and accounts for all development and manufacturing costs associated with each product.

As all three plans interact, it is essential that the results of all plans influence each other. Therefore, the plans are iteratively refined so that the optimum combination of best possible market presence and least possible complexity costs is attained.

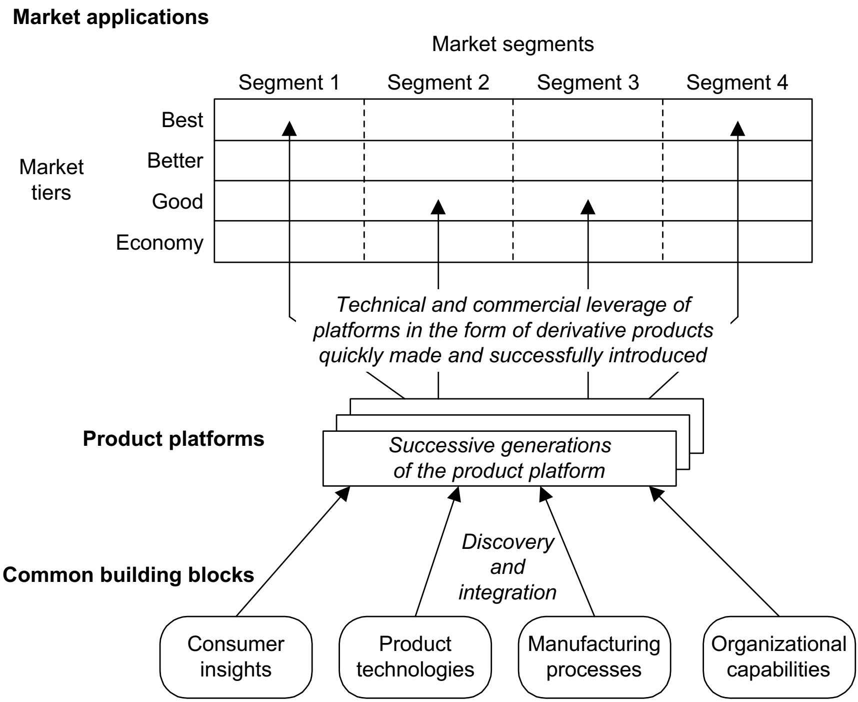


Figure 3.12 The power tower; source: Meyer and Lehnerd (1997, p. 38)

Meyer and Lehnerd (1997, pp. 37-48) introduced the "*power tower,*" an integrative model for managing product and process innovation (see Figure 3.12). It forces management to consider the following three elements:

- In a first step, the *market applications* are visualized by a matrix of customer segments and product price and / or performance that defines what customer groups the derivative products go to.

- Every company must, second, determine the architecture of its *product platforms* most suitable for its particular business. Meyer and Lehnerd (1997, p. 41) emphasized that product platforms should provide leverage, i.e. they must be capable of accommodating new technologies and variations, enabling firms to create derivative products at a low cost.[22]

- The third step determines the *common technical and organizational building blocks* forming the basis of product platforms. These building blocks are categorized into four areas: insights into customer needs, product technologies, manufacturing processes, and organizational capabilities. These capabilities must be leveraged across the product platforms of different product lines to achieve more successful product development. Meyer and Utterback (1993) reiterated this point and elaborated on the importance of closely coupling a company's core capabilities with its product platforms.

To underscore the great potential of product platforms, Meyer and Lehnerd (1997, pp. 13-15) reported that Black & Decker, a producer of consumer power tools, was able to maintain a staggering rate of new product introduction of one per week on average after dedicating three years to planning its platform. Thanks to massive cost savings, a gross margin of 50 percent over its cost of goods sold was achieved, even

[22] When creating derivative products, the costs of the platform elements carried forward are essentially sunk costs. Only the marginal costs of creating variations accrue to the derivatives (Meyer & Lehnerd, 1997, p. 41).

though end-user price reductions amounted to over 50 percent in some instances. Not surprisingly, demand soared and many competitors exited the market.

Besides these strengths of product platforms, the following problematic areas can be identified:[23] (1) due to only approximate market estimates, determining the number of platforms and derivative variants is a very difficult, if not impossible task; (2) usually, platforms take long to develop, which translates to higher market risk; (3) platform projects bind a considerable number of resources. These issues must be closely monitored to avoid jeopardizing platform projects.

The two concepts for implementing product platforms presented above integrate product strategy and market aspects to a considerable degree (by the product and differentiation plans, and the market application step in the power tower). When establishing a platform, the analysis of and decisions about the product architecture are of fundamental importance. The wide-spread usage of the concept points out its applicability in industry. Procedures to quantify a product's complexity are not provided, however. Table 3.3 at the end of this chapter shows a summarizing evaluation of the product platform concept with regard to the assessment criteria presented in Section 3.1.

Example: Fiat and Ford of Europe

Fiat and Ford of Europe agreed in late 2005 to tie up on developing and manufacturing their next generation small cars – the Fiat 500 and the Ford Ka – on the same product platform. Both models will be based on a shortened version of the platform for the Fiat Panda and are designed to share major components, such as Fiat engines and transmissions. The shells and interiors, which are those parts that are most visible to customers, will be different. Ford will pay Fiat for the platform per finished car as both models will be produced at Fiat's Tychy plant in Poland. It is a win-win situation for both car

[23] The problematic areas as presented here are based on Boutellier et al. (1997, pp. 60-61).

manufacturers because Ford gets inexpensive access to Fiat's small-car technology, while Fiat can fill its factories.

By teaming up on development (Fiat will manage the development of the power train and chassis), each partner can save roughly half of the $1.5 billion cost of developing a new model. Also, Ford of Europe can shift part of its manufacturing to low-cost Eastern Europe, where wages are 20% lower than in Spain, where the Ka has been produced since 1997. Contrary to a full-scale merger of two companies, the tieup focuses on the engineering and production for two car models based on the same product platform. The move spares Fiat and Ford of Europe the initial expenses usually associated with mergers and supports them in concentrating on cutting costs. Source: Edmondson (2005).

3.3.8 Variant Mode and Effects Analysis

The methodology introduced by Caesar (1991), variant mode and effects analysis (VMEA), provides an approach for designers to reduce product variety that is not perceived by customers. It is based on the following four steps (see Figure 3.13):

- *Variety analysis.* In the first step, the current product is investigated with regard to its variety. Normally, the variant tree (see Schuh, 1989, pp. 45-54) is employed as a supporting tool to visualize how the product variants evolve over the assembly process. Figure 3.14 depicts such a variant tree for an automotive exhaust system.

- *Priority setting.* Based on the variety analysis, the components and subassemblies with the largest potential to reduce variety are indicated.

- *Variety-oriented product design.* New design concepts are generated that manage to provide the necessary variety while using as many standardized components as possible.

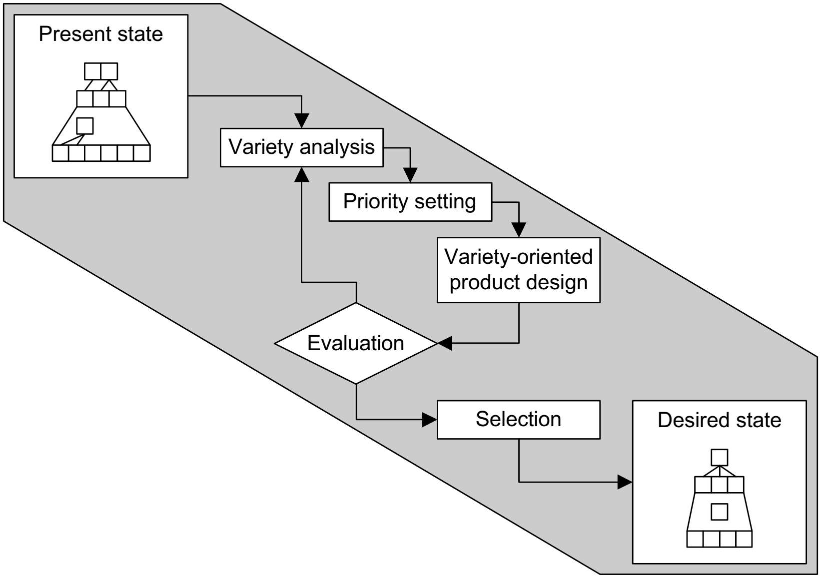


Figure 3.13 Procedure outlined by VMEA; source: Caesar (1991, p. 36)

- *Evaluation.* The design concepts are evaluated by means of a set of key figures reflecting both design and cost issues.[24]

The above process is run through iteratively, i.e. the design concepts are refined and evaluated several times. At the end of the iteration, the most suitable concept is selected for implementation.

The VMEA concept clearly concentrates on reducing product variety during the design process. It intentionally does not integrate market and strategic aspects since the variety desired by customers is considered a given by the model (Caesar, 1991, p. 33). By analyzing the effects of component variety with the variant tree, VMEA investigates product architecture to a considerable degree. Furthermore, the set of design

[24] Caesar (1991, pp. 74-80 and pp. 164-174) introduced a broad set of quantitative figures that consider product variety.

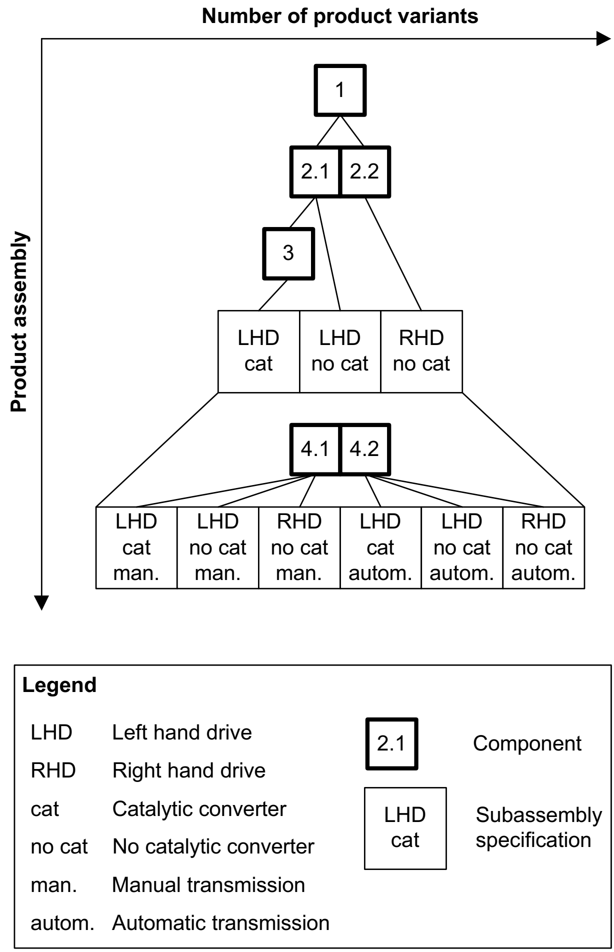


Figure 3.14 Variant tree for an automotive exhaust system[25]; source: Schuh (1989, p. 47)

[25] When reading the variant tree from top to bottom (along the direction of assembly), the pre-silencer (1) is connected to one of two exhaust bend variants (2.1 and 2.2). Exhaust bend 2.1 represents the left hand drive (LHD) variant, while 2.2 is used for right hand drive (RHD) systems. The catalytic converter (3) is an optional component that does not occur in all product variants (only LHD with cat) and never in RHD systems. The cat increases the number of variants at that stage from two to three. The heat insulation subassembly has two variants (4.1 and 4.2) employed in all variants, depending on whether the transmission is manual or automatic. This last component doubles variety (six product variants at the end of the assembly process).

and cost indices mentioned above provides a thorough framework to quantify product complexity. Table 3.3 at the end of this chapter shows a summarizing evaluation of VMEA with regard to the assessment criteria presented in Section 3.1.

3.3.9 Variety Reduction Program

The variety reduction program (VRP) advanced by Suzue and Kohdate (1990)[26] presents a concept to decrease complexity costs by reducing the number and variety of parts and processes. To this end, costs are divided into the following three categories (Suzue & Kohdate, 1990, pp. 28-36):

- *Variety costs.* The sheer variety of parts and processes causes this type of cost and includes, for instance, expenses for retooling, new equipment and other investments when introducing a new type of part. The larger the variety and the smaller the lot sizes, the higher the variety costs.
- *Function costs.* Factors such as product specifications, designed functions, and product construction method all contribute to the function cost and cause it to fluctuate. As they are based on market needs, function costs depend heavily on how well customer requirements are translated into product structure.
- *Control costs.* The costs incurred by people in the design team, the production facilities, and the materials department are defined as the control costs. These expenses are all due to activities for planning and controlling parts and processes.

In order to quantify the individual cost drivers of the above complexity cost types, Suzue and Kodate (1990, pp. 39-44) defined the parts index, production process index, and control point index. They serve to indicate how well a company's plant is performing in developing and manufacturing products. The *parts index* depends on the number of part types that go into a particular product; the *production process index*

[26] The English translation, published in 1990, is based on two Japanese editions, VRP *buhin hangenka keikaku* (1984), and VRP *giho ni yoru seihin costo daun suishin manyuaru* (1985). Source: Suzue and Kohdate (1990, p. iv).

indicates how many lines and how many processes are used by a particular part or product; the *control point index* is based on the number of control points.[27]

Cutting variety, function, and control costs is performed by means of five techniques that all influence parts and processes (Suzue & Kohdate, 1990, pp. 44-47, 57-70):

- The *fixed vs. variable* technique distinguishes between fixed parts as standard, commonly used parts, and variable parts that address changing market needs.

- The *combination* technique attempts to provide the variety needed with less component variety. The building block system and product modularization are examples to implement this technique.

- In the *multifunctionality and integration* technique, it is sought to reduce the number of parts and processes by integrating the required functions into a smaller number of parts.

- Product attributes are split into distinct value ranges by the *range* technique. For a component's dimension (e.g. length, diameter, etc.), dimension ranges are created that are applicable in as many models as possible.

- The *trend* technique organizes product attribute values and investigates their distribution along several dimensions. The goal is to eliminate unnecessary product variants, which reduces parts, production processes, equipment, and retooling operations.

The variety reduction program presents a promising concept to assess complexity costs and identify ways to reduce the number and variety of parts and processes. Quantification of complexity is addressed to a considerable degree, and product architecture

---

[27] A control point is defined as a substantial change in the control process relating to the flow of drawings, materials, parts, etc. (Suzue & Kohdate, 1990, p. 40). For example, drawings handed over from the design department to the production technology team, which determines the necessary tools, is counted as a control point.

issues are involved in part, too. However, the underlying market demands causing product variety and how the firm decides to strategically cope with external complexity are both excluded from the method. Table 3.3 in the next section shows a summarizing evaluation of the variety reduction program with regard to the assessment criteria presented in Section 3.1.

3.4 Assessment Summary

Table 3.3 summarizes the assessment of the concepts presented in the previous sections with respect to the five criteria introduced in Section 3.1.

Table 3.3 Assessment summary

	Company / product strategy	Market aspects	Product architecture	Quantification of complexity	Applicability in practice
Mass customization	●	◕	○	○	●
Lean management	◕	◕	○	○	●
Optimum variety	◕	◕	○	○	◑
Quality function deployment	◔	●	◔	○	●
Target costing	◑	●	◔	○	●
Design for variety	○	○	◑	◕	◕
Design for configuration	○	◕	●	○	◕
Modularization	◔	◑	●	○	●
Modular function deployment	○	●	◕	◑	◕
Product platforms	◕	◕	●	○	●
Variant mode and effects analysis	○	○	◕	◕	◑
Variety reduction program	○	○	◑	◕	●

Legend

○ Not fulfilled

◔ Fulfilled to a low degree

◑ Partly fulfilled

◕ Fulfilled to a considerable degree

● Completely fulfilled

4 Complexity Management Model

4.1 Overview

The previous chapter has shown that currently there is no concept or tool in the field of complexity management that satisfies all five criteria of Section 3.1 (strategic aspects, market aspects, product architecture, quantification of complexity, and applicability in an industry setting). This chapter presents a model that attempts to combine all of these issues. It is obvious, therefore, that the five criteria will be drawn upon at a later stage of this work to assess the complexity management model presented here. The model's focus is on products and their architectures and, thus, is always applied to a particular product (or close-knit product families), but never to an entire product portfolio, an enterprise, or some other organizational unit.

To give a brief overview, the model consists of three major building blocks that are all run through consecutively when applied to a product (see Figure 4.1):

- *Strategy and product life cycle assessment.* The purpose of the first step is to analyze the general surrounding and the setting of the product. The company's strategy is taken into account, as well as the product's positioning within its life cycle. The analysis performed here deliberately involves only qualitative data. Section 4.2 provides more details on this first step.

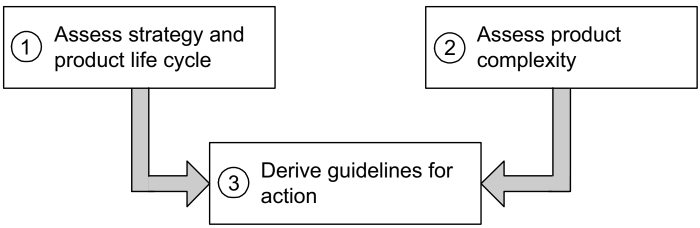


Figure 4.1 Three-step procedure of the complexity management model

- *Product complexity assessment.* Here, the analysis goes into the nuts and bolts of a product. Quantitative data on product functionality and physical complexity are gathered. The result of this second step is the *complexity matrix,* the model's starting point for optimizing the product's architecture. In Section 4.3, the procedure of computing all inputs for the complexity matrix is explained by using an insightful example.

- *Derivation of guidelines for action.* The last step integrates the findings of the previous two steps to provide managers with guidelines for action to reduce complexity within their products. Components and modules that are the source of major concern are highlighted, and support for optimizing product architecture is given (see Section 4.4).

Even though steps one and two are normally performed consecutively, they could basically be tackled in parallel, as shown in Figure 4.1. The second step does not require any information from the first, and vice versa – they are independent. The third step, however, is based on the previous two steps.

4.2 Strategy and Product Life Cycle Assessment

The objective of the complexity management model presented in this work is to optimize product architecture. Altering and improving product architecture involves changing, merging, or splitting components and entire modules, redesigning and standardizing interfaces, and in some cases rethinking the fundamental concept of a product. This optimization process affects virtually all of a firm's functional areas, from product development to manufacturing to sales, and must also take into account the strategic direction the company decided to pursue. The guidelines for action derived in Section 4.4 depend on, for instance, whether a product is fully standardized and caters to a broad market, or whether it is an entirely customized solution uniquely manufactured for one single customer. Therefore, the first step of the complexity management model considers two aspects intended to provide a broadly based picture of a product's environment:

- In the *strategy* assessment part, the firm's strategic positioning is considered (see Subsection 4.2.1).
- The *product life cycle* part determines the product's current phase within its life cycle (see Subsection 4.2.2).

I selected the above two steps as I believe they describe two very important dimensions for analyzing a product's surrounding when optimizing its architecture. Furthermore, a wide range of literature is available on the two subjects,[1] which makes it more convenient to choose a ready-to-use concept appropriate for the situation at hand. It can be argued, of course, that strategic and product life cycle aspects do not fully capture all constraints imposed on a product architecture optimization. Giving way to such concerns would, however, overload the model and obscure the essentials.

4.2.1 Strategic Considerations

The following treatment of strategic issues begins with describing the extremes of standardization and customization (4.2.1.1). They are drawn upon by the subsequent framework of generic strategies (4.2.1.2). Hybrid competitive strategies add an additional dimension to the generic strategies and also employ the concept of standardization and customization (4.2.1.3). An additional aspect is brought in by the fact that different strategies apply depending on the industry (4.2.1.4). Finally, the strategic considerations are condensed into a short summary (4.2.1.5).

4.2.1.1 Standardization versus Customization

A firm can choose from an infinitely wide variety of possibilities to structure its product lines, the two extremes of which are of special interest here. In the first case, all products offered differ from one another, while in the second the exact same product is sold over and over again. These two extreme cases are called *customization* and *stan-*

[1] Reference to the literature will be made in the respective Subsections 4.2.1 and 4.2.2.

dardization, respectively. They are hardly ever found in industry in such a pure form, as they merely outline an idealized typology. Still, knowing the two extremes of a continuous scale allows assessing what is in between. The two strategic dimensions of customization and standardization will accompany us throughout the remainder of this work. Thus, a short introduction follows.

An enterprise offering customized products caters to a very narrow customer base and tailors its every offering to the very needs of one particular customer. Normally, very close ties between the producer and its customers are established and, often, customers participate in designing the product and express preferences on how it should be manufactured. A customizer's competitive edge is therefore primarily based on product attractiveness; the product is differentiated from competitors' by satisfying each and every customer requirement. The know-how necessary to maintain such a market position is an invaluable asset, but requires continuous investments. Further costs are incurred by the large product variety, by increasing complexity throughout the value chain, and by highly qualified personnel, to name a few. This cost disadvantage can only be balanced by a higher price. The more customized a product, the more customers are willing to pay a higher price because the product closely reflects their requirements. As customer benefit increases, the price elasticity of demand decreases, which enables the producer to harvest the consumer surplus (Mayer, 1993, pp. 58-62). The negative cost effects mentioned above are balanced somewhat by the economies of scope that can be realized due to synergies becoming apparent when producing several products simultaneously. If those products have something in common (e.g. fabrication tools, R&D resources, etc.), the shared activities and assets can be "spread" across the products. This, in turn, leads to lower costs than if they were produced separately (e.g. by different companies). Table 4.1 summarizes the key characteristics of the customization strategy.

Contrary to customization, an enterprise following a standardization strategy sells homogeneous mass products. Close relationships between customer and producer no longer exist and are replaced by anonymity. Therefore, a certain (large) number of product units are manufactured based on market research estimates, i.e. products are not made to order but kept in stock. As mass-produced, standardized goods cannot consider individual customer preferences, their product attributes are chosen based on an average taken from a large number of customers believed to best reflect what customers want.[2] Since individual customers' preferences diverge from this average preference, the benefit provided by the product – and thus the price at which it is sold – is much lower than in the customization case. The competitive edge of a mass producer is always price-based. By producing the same standardized product in large quantities, costs can be saved thanks to the following two effects:

- *Economies of scale* are achieved due to generally larger facilities (factories, call centers, inventory, etc.), which spreads a considerable fraction of fixed costs to a large number of product units (Mayer, 1993, pp. 94-96).
- The *experience curve* effect states that costs drop by 20 to 30 percent every time the cumulative volume doubles (Seiler, 2000b, p. 273). The main reasons are increased labor efficiency (thanks to learning), specialization and redesign of labor tasks, product and process improvements, and rationalization, such as introducing more up-to-date technology (Hax & Majluf, 1982, pp. 53-54).[3]

Table 4.1 summarizes the key characteristics of the standardization strategy and highlights the most pronounced differences to the customization strategy.

[2] Conjoint analysis is a very powerful tool to assess preferences of a wide customer base. It provides the necessary data to cluster individual customers' "ideal points" and to create specific product offerings thereupon. (A customer's ideal point is defined by his / her preferred combination of attribute values from the range of attributes and values offered.) See Subsection 2.1.2 for more information on conjoint analysis.

[3] The experience curve effect was first discovered in 1925 at Wright-Patterson airbase in Ohio, where employees of the U.S. airforce discovered that labor costs per unit weight of the airplane to be assembled decreased digressively with time (Wright, 1936, as cited in Mayer, 1993, p. 90). In the early 1960s, The Boston Consulting Group analyzed the phenomenon systematically for several industries and coined the term "experience curve" (Seiler, 2000b, pp. 273-274).

Table 4.1 Characteristics of the customization and standardization strategies; source: Mayer (1993, p. 137)

Characteristic	**Customization**	**Standardization**
Scope of offering	Specifications of individual customers	Average preference of a large number of customers
Number of customers per offering	One, or very few	Many
Contact to customer	Close; customer integrated in designing and producing product	Not or hardly established (anonymous consumers)
Product fabrication	After order	Before order; in stock
Source of information on customer requirements	Directly from customer	Market research
Similarity of products within line	No product the same; tailored solution; batch size one	All products the same; homogeneous mass product
Product variety	Very large	Only one product variant
Product attractiveness	Inherently high	Inherently low
Customer retention	High	Low
Costs	High	Low
Risk of substitution	Low	High
Competitive effect	Decoupled from competition due to product attractiveness and know-how advantage; opportunity to avoid price-based competition	Risk of price-based competition (especially for firms with low market share); market leader protected by cost advantage
Market entry barrier	Product attractiveness and know-how advantage	Cost advantage of market leader
Price range	Rather high	Rather low

Example: Volkswagen

The German automobile manufacturer Volkswagen (VW) introduced in December 2006 a new line called VW Individual. It stands for special solutions customized to the wishes of individual customers, such as an interface with the iPod, the leather color matching with her favorite nail polish, a refrigerator, etc. The basic VW models of the mass market (Golf, Polo, etc.) and the upper segment (e.g. Touarag) can all be complemented with a customized touch and receive the VW Individual label.

The new line is a novel concept among mass market auto producers and attempts to offer a highly customized alternative next to the more standardized portfolio. VW founded an independent competence center to ensure that the VW Individual line is perceived as something distinct from the "normal" models. Thanks to the new line, VW gives itself a new positioning in the continuum between customization and standardization, serving both cost-sensitive customers and individuals that desire a personal touch to their car. Source: "Begeisterung für das Besondere" (2007).

4.2.1.2 Porter's (1980) Framework of Generic Strategies

Probably the most widely known strategy framework (and the best established in industry and academic literature[4]) are the *three generic strategies* introduced by Porter (1980). Piller (2003, p. 211) reasoned that its profound influence on researchers and practitioners alike stems from its clarity and unequivocal conception, but also from its apparent weaknesses. The three generic strategies, depicted in Figure 4.2, are strongly based on the above discussion of customization and standardization. Porter, too, acknowledged the existence of the two extremes but added a further aspect to his model: whether the strategic target is industrywide or focused on a particular segment. This leads to the following generic competitive strategies (cf. Porter, 1980, pp. 34-41):

- In the *differentiation* strategy, the firm creates a product offering that is perceived industrywide as being unique. The differentiation can take many forms: design,

[4] A study by Miller and Dess (1993, pp. 553-554) found that between 1986 and 1990, half of all manuscripts published in the Strategic Management Journal referenced Porter's (1980) work.

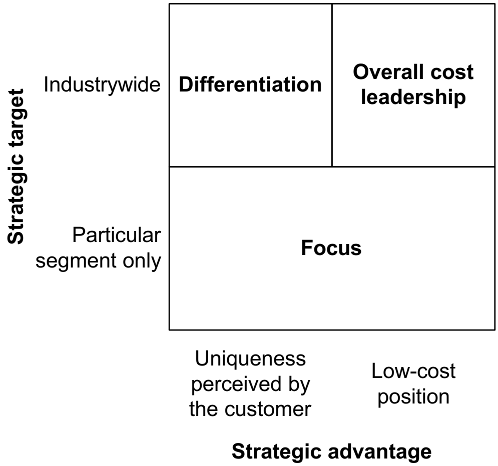


Figure 4.2 Three generic competitive strategies; source: Porter (1980, p. 39)

brand, technology, features, customer service, dealer network, and many more. The firm should attempt to differentiate itself along several dimensions.[5] Costs are not allowed to be ignored, of course, but they are not the primary strategic target. If differentiation is achieved, above-average returns can be yielded due to the defensible position it creates. Brand loyalty and lower sensitivity to price, decreased buyer power due to a lack of comparable alternatives, margins that avoid the urge for a low-cost position, and the resulting entry barriers are all factors that make differentiation a viable competitive strategy.

- The *overall cost leadership* strategy dictates that the firm construct efficient and appropriately scaled facilities, pursue cost reduction based on the experience curve, and tightly control direct costs and overhead. Even though lower cost relative to competitors is the major strategic target, a watchful eye must be placed on quality,

[5] It can be argued whether a firm should exploit several aspects of differentiation, as Porter (1980, p. 37) argued. In certain cases it is more appropriate to concentrate the firm's resources on one or two dimensions of differentiation.

service, and customer satisfaction. Achieving overall cost leadership yields above-average returns due to the lower costs, while competitors have competed away their profits.

- A company pursuing the *focus* strategy caters to a particular segment only (e.g. one particular buyer group, geographic market, etc.). It bases its above-average returns on serving a particular target very well, i.e. more efficiently than competitors competing more broadly. A focus strategy either achieves differentiation by better meeting the needs and wants of the particular target it focuses on, or manages to maintain lower costs in serving this target, or both. The differentiation or lower cost position is not achieved for the entire market, but only for the narrow market target.

Porter (1980, pp. 41-44) argued that firms failing to establish their strategies in one of the above three directions are in a very difficult strategic situation – they are *"stuck in the middle."* Such firms lack the market share, capital investment, and know-how necessary for a low-cost, differentiation, or focus position and, in turn, suffer of a lower profitability. The concept of being "stuck in the middle" implies that the three generic strategies are alternative approaches, i.e. their combination is not considered a viable option. In other words, cost leadership and differentiation are regarded as being incompatible.

4.2.1.3 Hybrid Competitive Strategies

In subsequent publications, the postulate that cost leadership and differentiation mutually exclude each other was repeatedly questioned. Murray (1988, pp. 395-397) argued that the preconditions for a viable cost leadership strategy stem primarily from industry's structural characteristics, while customer preferences principally form the preconditions for product differentiation. Because industry structure and customer preferences are two independent factors, the possibility of combining the two strategies is not necessarily precluded.[6] Extending the logic, Murray (1988, p. 396) concluded that "by combining both generic strategies successfully, a firm should be able to outcompete rivals that pursue only one strategy." Hill (1988) advanced the thesis that pursuing both strategies simultaneously results in a sustainable competitive advantage.

Based on these weaknesses of the generic strategies, a growing number of *hybrid competitive strategies* are emerging. They are all based on the premise that cost leadership and differentiation can be pursued simultaneously to form a powerful combination. The matrix in Figure 4.3 illustrates this growing awareness of the profound advantages of combined strategies: by combining a high relative differentiation position and a low relative cost position, a leadership position among competitors can be achieved. The two generic strategies (high relative differentiation / high relative cost position, and low relative differentiation / low relative cost position) are only ascribed an average competitive position.

One prominent member of the hybrid strategy family is *mass customization,* where attempts are made to produce customized solutions for a relatively large market at a cost level approaching that of a mass producer.[7] The matrix shown in Figure 4.4 is defined by the two dimensions of differentiation and standardization. Mass customization can be located somewhere between pure differentiation (customization) and pure cost leadership (standardization), its exact location depending on the strategy the company decides to pursue.

When studying hybrid competitive strategies, the question soon arises whether a firm should tackle differentiation and cost leadership at the same time or whether it is more promising to concentrate on one after the other. Hybrid strategies can be classified into *simultaneous hybrid strategies,* where differentiation and low-cost position

---

[6] Similarly, Piller (2003, p. 213) reasoned that cost leadership is based on special structural preconditions of the producer (such as efficient fabrication systems), while differentiation is driven by market aspects.

[7] See Subsection 3.2.1 for more details on mass customization.

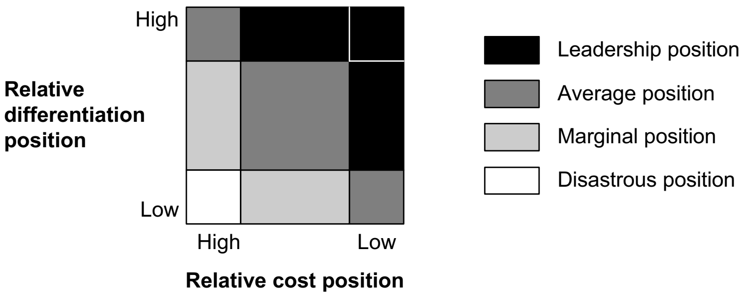


Figure 4.3 Matrix of competitive strategies; source: Hall (1980, p. 80)

are attained in parallel. The rationale here is to ensure that all strategic decisions are consistent with both a high relative differentiation position as well as a low relative cost position. Contrary to that, *sequential hybrid strategies* first concentrate on one of the two, then the other. They assume that it is a too demanding task to handle differentiation and cost leadership in parallel and therefore recommend a sequential procedure.[8] Fleck (1995, pp 59-152) and Piller (2003, pp. 219-222) gave an in-depth overview of simultaneous and sequential hybrid strategies.[9]

When the two competitive advantage dimensions of relative differentiation and relative cost position (shown in Figure 4.3) are supplemented with Porter's (1980) dimension of market scope (the "strategic target" axis in Figure 4.2), a cube of competitive strategies can easily be drawn (see Figure 4.5). Basically, a firm's strategic positioning can be placed anywhere in the three-dimensional space defined by the cube.

8 White (1986, p. 230) introduced the two concepts of either combining differentiation and cost leadership simultaneously or giving sequential attention to both strategies.

9 An example of a sequential hybrid strategy is the outpacing strategy introduced by Gilbert and Strebel (1987). Besides mass customization, Piller (2003, p. 220) identified – among others – the dynamic product differentiation concept (Kaluza, 1989) and the hypothesis of simultaneity (Corsten & Will, 1993) as simultaneous hybrid strategies. While discriminating between sequential and simultaneous strategies primarily focuses on time as a differentiating attribute, Fleck (1995, pp. 71-80) introduced a third type named multilocal hybrid strategies. These strategies emphasize the spatial decoupling of differentiation and low-cost position.

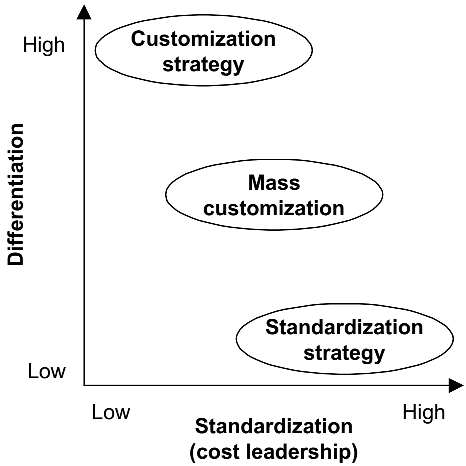


Figure 4.4 Complexity management strategies; adapted from Schuh and Schwenk (2001, p. 62)

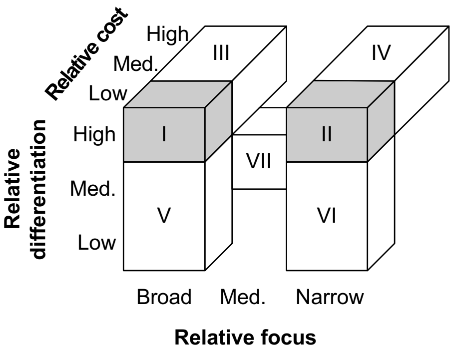
Miller and Dess (1993) identified seven possible strategies that are shown in Figure 4.5. Note that four of these are identical to Porter's framework: differentiation (III), cost leadership (V), differentiation focus (IV), and cost focus (VI).[10] Strategy VII represents the situation of being "stuck in the middle" mentioned above. The two hybrid strategies I and II describe what Porter did not regard as a successful (and sustainable) positioning. Miller and Dess (1993) conducted an empirical study based on 715 strategic business units (SBUs) taken from PIMS data[11] and assessed the success of the above seven strategies. The performance variables were chosen as follows: return on investment (ROI), cashflow on investment, real sales growth, market share gained, and ROI's instability. The striking result was that the hybrid group I proved to be by far

[10] Strictly speaking, Porter's framework as depicted in Figure 4.2 exhibits only three generic strategies. The differentiation focus and cost focus strategies as defined in Figure 4.5 are both part of the focus strategy in Figure 4.2.

[11] The objective of the PIMS (profit impact of market strategies) program is to investigate the factors that are decisive for sustained success of SBUs. PIMS data are based on more than 3000 SBUs, collected over several years. Eight factors were identified that are strongly related to and very important for a business' profitability (Seiler, 2000b, pp. 301-302).

the most successful of all (ROI: 37.8%). Moreover, group VII ("stuck in the middle") was slightly more successful than the niche differentiators (IV). The table in the lower portion of Figure 4.5 summarizes the study's findings with regard to the above set of performance variables. The authors pointed out that their research had indicated that "not only are hybrids feasible, but also that they are extremely profitable" (Miller & Dess, 1993, p. 579).

In a publication following his 1980 work, Porter (1985, pp. 19-20) recognized that in certain cases, combining the differentiation and cost leadership strategies is a viable option. He mentioned three circumstances under which companies can successfully pursue a combined strategy: (1) if competitors are "stuck in the middle," (2) if the company enjoys considerable economies of scale (due to a large market share) or if it can exploit interrelationships between industries that others cannot, and (3) if it pio-



I Hybrid, broad
II Hybrid focus
III Differentiation, broad
IV Differentiation focus
V Cost, broad
VI Cost focus
VII Stuck in the middle

Performance variables	**I**	**II**	**III**	**IV**	**V**	**VI**	**VII**
Return on investment	37.8	31.6	32.9	30.2	17.0	23.7	17.8
Cashflow on investment	5.2	4.1	4.5	4.0	2.3	3.2	2.4
Real sales growth	11.5	21.3	13.5	13.5	16.4	17.5	12.2
Market share gained	4.1	7.2	5.3	5.5	6.1	6.3	4.4
ROI's instability	2.7	3.0	2.6	2.2	2.7	2.6	2.0
N	78	45	160	100	141	86	105

Figure 4.5 Three-dimensional extension of Porter's framework; source: Miller and Dess (1993, p. 565)

neers a major innovation. All of these conditions, however, are considered temporary by Porter. In a later publication in which operational effectiveness and strategic positioning are compared, Porter (1996, p. 62) stated that a company can outperform its rivals by delivering greater value to customers, or creating comparable value at a lower cost, or pursuing both. In doing so, companies are bounded by a productivity frontier that is defined by the best possible trade-off between relative differentiation and relative cost position (see Figure 4.6).[12] The frontier itself is constantly moving outward as new technologies are developed and new inputs become available. Figure 4.6 visualizes the productivity frontier and clearly indicates that low relative cost position and high relative differentiation can be combined to a certain degree.

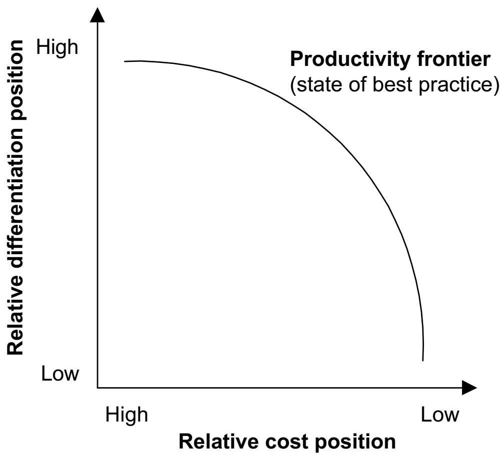


Figure 4.6 Productivity frontier defining the best possible trade-off between relative cost position and buyer value delivered[13]

[12] According to Porter (1996, p. 62), the best possible trade-off can be determined by investigating existing best practices.

[13] The graphic in Figure 4.6 is derived from Porter (1996, p. 62). In the original version, the ordinate is labeled “nonprice buyer value delivered.”

4.2.1.4 Different Strategies for Different Industries

Before proceeding to a brief summary of the strategic considerations, a last framework to assess a company's positioning on the standardization and customization scale is presented here. The model introduced by Lampel and Mintzberg (1996) does not draw on Porter's generic strategies and solely focuses on providing details on the continuum between the extremes of standardization and customization. A set of five strategies was defined: pure standardization, segmented standardization, customized standardization, tailored customization, and pure customization (see Figure 4.7).[14] Next, the authors identified seven different industry types, also listed in Figure 4.7. These industries were then classified along the standardization and customization continuum with regard to their process strategies, product strategies, and transaction strategies, the result of which is shown in Figure 4.7. Mass industries and thin industries are fully standardized and customized, respectively, in their processes, products, and transactions. The remaining five industries show different values for their process, product, and transaction strategies along the standardization and customization scale. Agent industries, for instance, such as health care or auditing, involve a rather generic or standardized transaction, governed by standard contracts and professional or technical codes of conduct.[15] Yet the actual activities tend to be craftlike in nature, tailoring professional skills to each customer's specific needs. Processes and services (the "product") are therefore best described as tailored customization. In health care, for example, interventions are based on a standardized procedure adapted to a particular patient's condition. The strategic situation of every industry in Figure 4.7 can be interpreted in a similar way as just presented for agent industries. For further details on the industry and strategy classification by Lampel and Mintzberg (1996), refer to their respective work.

---

[14] Note that these strategies differ in the location of their order penetration point (OPP). (See footnote 3 in Chapter 2 for an explanation of the OPP.) The closer to pure customization, the earlier the OPP occurs in the value chain. Figure B.1 in Appendix B depicts the five strategies introduced by Lampel and Mintzberg (1996) with respect to the value chain and the OPP.

[15] As an example, Lampel and Mintzberg (1996, p. 28) mentioned that we do not generally bargain over price with our medical doctors.

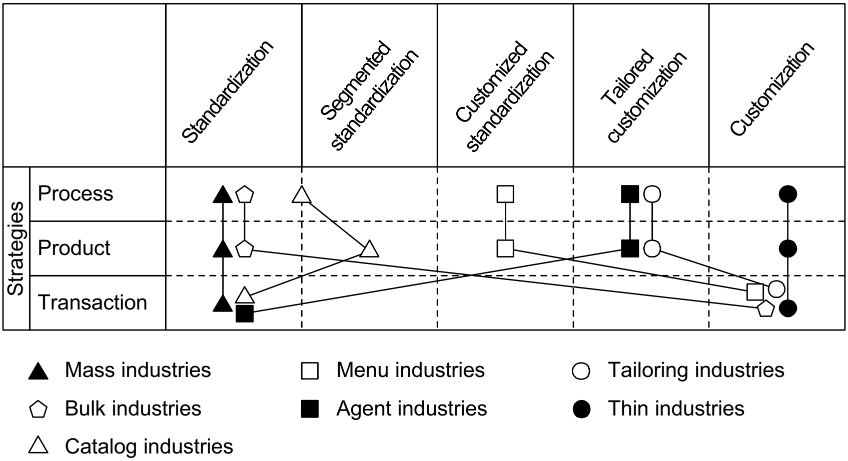


Figure 4.7 Strategic positioning of industries[16]

4.2.1.5 Strategic Considerations: Intermediate Summary

So far, several frameworks have been introduced that all provide a means to assess a company's strategic positioning. First, the terms of standardization and customization have been presented as the foundation for understanding the subsequent concepts. Porter's (1980) generic strategies then added a dimension considering whether the firm is active in the broad market or a niche. Hybrid competitive strategies even allow for a combination of the two seemingly contradictory extremes of standardization and customization (or, in Porter's terms, cost leadership and differentiation) and posit their superiority to non-hybrid strategies in many situations. Finally, a classification has been described that categorizes different industries according to the degree of customization with regard to the industries' products, processes, and transactions.

[16] Own illustration of Lampel and Mintzberg's (1996) classification of industries.

The complexity management model presented in this chapter does not prescribe, in a strict sense, which concepts to employ and which ones not when assessing company strategy. The above list's objective is merely to give the model's user an idea of the breadth of available strategic frameworks. Nevertheless, I believe that in the context of product architecture, the decision to customize or standardize (or to combine both) is of fundamental importance for and has a profound influence on the optimization of a product's architecture. Therefore, the model's application always involves considering the firm's choice of strategic positioning, which is most suitably done with the concepts (or a selection of them) presented above. The goal, in any case, is to provide a strategic foundation for the more quantitatively oriented assessment of product complexity (see Section 4.3). Moreover, knowing and being able to classify a firm's strategy supports the derivation of norm strategies in Section 4.4 and assists the decision-making process when optimizing product architecture. In summary, the concepts presented in this subsection provide guidelines (but not a strict procedure) as to how a company's strategy can be assessed in order to tune the recommendations for managerial action in one or another direction.

4.2.2 Product Life Cycle Considerations

Apart from strategy, the life cycle of the product which the complexity management model is applied to also exerts an influence on the implications the model draws. Depending on the phase of the life cycle, the basic norm strategies that will be derived in Section 4.4 must be adjusted to the specific product under investigation. For example, architectural optimizations will take a different form for products that are still in an early, introductory phase than for products that will be canceled from the portfolio some time soon.

The most widely employed framework for describing product life cycle (PLC) is the four-phase concept depicted in Figure 4.8, with the corresponding changes in sales and profits also shown. The four phases of a PLC can be sketched as follows (see Kotler & Keller, 2006, p. 322; Seiler, 2000b, pp. 144-148):

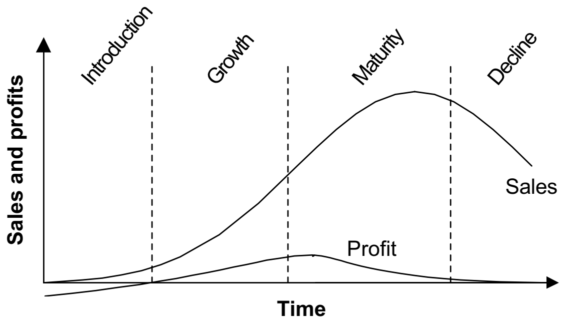


Figure 4.8 Sales and profit during the product life cycle; source: Kotler and Keller (2006, p. 322)

- *Introduction.* This is a period of slow sales growth and nonexistent profits. The product's success strongly depends on customer acceptance and, therefore, expenses of product introduction (such as promotional expenditures) are at their highest ratio to sales. As costs are high, prices tend to be high, too.

- *Growth.* In the growth phase, sales climb rapidly, attracting new competitors. Prices remain where they are or fall only slightly, while promotion costs are spread over a larger volume and unit manufacturing costs fall faster than the price declines. Thus, profits increase at an accelerated speed. In order to cope with increased competition, companies usually enhance product design and extend functionality. They also add new models and variants and attempt to enter new market segments.

- *Maturity.* A slowdown in sales growth due to saturated markets marks the beginning of the maturity phase. Profits remain stable or decline because of intensified competition. The maturity stage typically lasts longer than the two previous ones, which translates to the situation of most products being in the maturity stage of the life cycle. While advertising is increased and R&D budgets are raised to develop product improvements and line extensions, all expenditures must be planned cautiously due to the inevitable divestments in the decline phase. Product managers try to stimulate sales by modifying the elements of the marketing program.[17] For instance, the extended product[18] becomes a more and more important marketing instrument, and companies change packaging, create additional features, and offer product-related services.

- *Decline.* Sales decrease and profits erode as a consequence of a shift in customer requirements, technological advances rendering the current product disadvantageous or even obsolete, or aggressive competition. Costs are cut rigorously, and prices are reduced further. Generally, firms either "milk" their investment to recover cash quickly or divest the business altogether. In some cases, though, increasing or at least maintaining the firm's level of investment in the product proves to be the better strategy – for instance, when the firm strives to dominate the market or strengthen its competitive position by forcing competitors to exit the market. The most appropriate strategy depends on the industry's attractiveness and on the company's competitive strength.

Table 4.2 gives a summary of the objectives and strategies during the four phases of a product's life cycle. In practice, the bell-shaped PLC curve shown in Figure 4.8 hardly ever occurs in the idealized form presented here. Rather, the cycle takes many different forms and durations, and varies from one application to the other (Seiler, 2000b, pp. 148-151). Kotler and Keller (2006, pp. 322-324) introduced three common alternate PLC patterns: the growth-slump-maturity pattern, the cycle-recycle pattern, and the scalloped pattern. They also distinguished three special categories of PLC:

---

[17] One traditional conception of marketing activities is in terms of the marketing mix, which is defined as the set of marketing tools a firm uses to pursue its marketing objectives. McCarthy (1996, as cited in Kotler & Keller, 2006, p. 19) classified these tools into four elements called the four Ps of marketing: product, price, place, and promotion.

[18] Seiler (2000a, pp. 200-201) differentiated between the core product, the formal product (packaging, quality, styling, product characteristics, etc.), and the extended product (installation, service, warranty, free delivery, etc.). This classification is similar to Kotler and Keller's (2006, pp. 372-373), who introduced a hierarchy of five levels: core benefit, basic product, expected product, augmented product, and potential product. Kotler and Keller's typology, in turn, is based on Levitt (1980).

Table 4.2 Summary of PLC objectives and strategies; based on Kotler and Keller (2006, p. 332)

Objectives and strategies	**Introduction**	**Growth**	**Maturity**	**Decline**
Marketing objectives	Create product awareness and trial	Maximize market share	Maximize profit while defending market share	Reduce expenditure and milk the brand
Product	Offer a basic product	Offer product extensions, service, warranty	Diversify brands and models	Phase out weak items
Price	Charge cost-plus	Price to penetrate market	Price to match or beat competitors'	Cut price
Distribution	Build selective distribution	Build intensive distribution	Build more intensive distribution	Go selective; phase out unprofitable outlets
Advertising	Build product awareness among early adopters and dealers	Build awareness and interest in the mass market	Stress brand differences and benefits	Reduce to level needed to retain hard-core loyals
Sales promotion	Use heavy sales promotion to entice trial	Reduce to take advantage of heavy consumer demand	Increase to encourage brand switching	Reduce to minimal level

style, fashion, and fad.[19] Critics of the PLC concept charge that it is often difficult to determine what stage the product is in. For example, a product might appear to be mature when actually it has only reached a temporary flattening of sales before another upsurge. They argue that the PLC concept is a result of marketing strategy and not an inevitable and given course that products must follow.[20] Dhalla and Yuspeh (1976, p. 105), who published an article that strongly challenged the PLC concept, posited that "clearly, the PLC is a dependent variable which is determined by marketing actions; it

[19] See Kotler and Keller (2006, p. 323) for depictions of the three alternate patterns and the three special categories, respectively.

[20] See Kotler and Keller (2006, p. 331) for a brief critique of the PLC concept.

is not an independent variable to which companies should adapt their marketing programs. Marketing management itself can alter the shape and duration of a brand's life cycle."

Despite the share of critics the PLC has, it is used here by the complexity management model as a framework to assess a product's situation and to provide an additional viewpoint apart from strategic considerations. Caution and flexibility must be used, though, when applying the PLC concept, and one must bear in mind that there is no single product that exactly follows the idealized pattern shown in Figure 4.8.

A study by Heina (1999) on managing product variety introduced a PLC-based concept that is also of interest here. Apart from the market phase – which is normally considered by PLC concepts – the product development and disposal phases are added (see Figure 4.9).[21] Depending on the phase the product is in, different strategies concerning product variety should be embraced. In the product development phase, product variants are generated to fulfill anticipated market requirements but, at the same time, preventive measures (often on the product architecture level) must be taken to avoid excessive variety generation in later stages of the product's life cycle. During the market phase, further variants are generated (to cope with market demand) while the preventive actions on the architectural level are maintained (to avoid excessive variety). The need for eliminating low-performing variants rises as variety proliferates,[22] and efficiently handling the growing portfolio becomes an important success factor.[23]

[21] Note that normally, the PLC is considered identical to the product's market cycle. Similar to Heina (1999), Pahl and Beitz (2003, p. 122) also introduced a product development phase prior to market introduction. Heina (1999) based his three phases on Fritz and Oelsnitz (1996, p. 126, as cited in Heina, 1999, p. 110), who divided the market cycle into five phases: introduction, growth, maturity, saturation, and decline.

[22] Eliminating product variants during the market phase can conflict with ongoing obligations and contracts. Manufacturers are often forced by customers to retain certain variants in their portfolio and maintain a broad range of spare parts for several years.

[23] As an example of handling product variety, Sekolec (2005, p. 30) mentioned the introduction of computer aided selling (CAS) and product configurators.

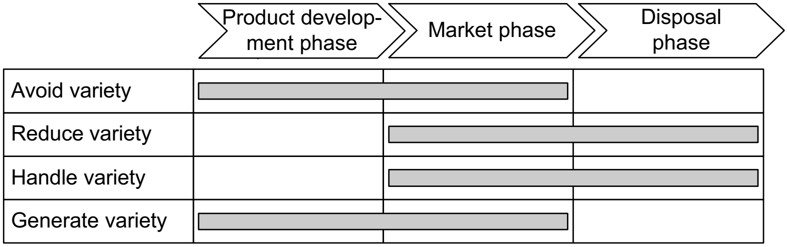


Figure 4.9 Product variety management decisions during a product's life cycle; source: Heina (1999, p. 42)

The disposal phase mainly focuses on the ongoing variety handling and on successively phasing out product variants.

This subsection has introduced the notion of product life cycle and outlined two concepts: the four-phase PLC and the concept of product variety management during a product's life cycle. I find these two concepts most useful in applying life cycle considerations to product architecture optimizations. As was the case in the preceding subsection on strategic considerations, the complexity management model does not prescribe the use of any particular concept. What it underscores is the need to take into account the product's life cycle positioning when applying the model. To do this, the concepts presented above provide a helpful guideline. Knowing a product's life cycle phase (and the corresponding implications) supports the derivation of guidelines for action in Section 4.4 and assists the decision-making process when optimizing product architecture.

4.2.3 Summary of Strategy and Product Life Cycle Assessment

The previous two subsections presented the first step of the complexity management model – strategy and product life cycle assessment. Figure 4.10 summarizes the key elements of the model's first step in a brief and schematic way. Step 1a in Figure 4.10 depicts the strategy assessment, while step 1b covers the product life cycle assessment. Note that steps 1a and 1b do not necessarily have to be gone through in this order, i.e.

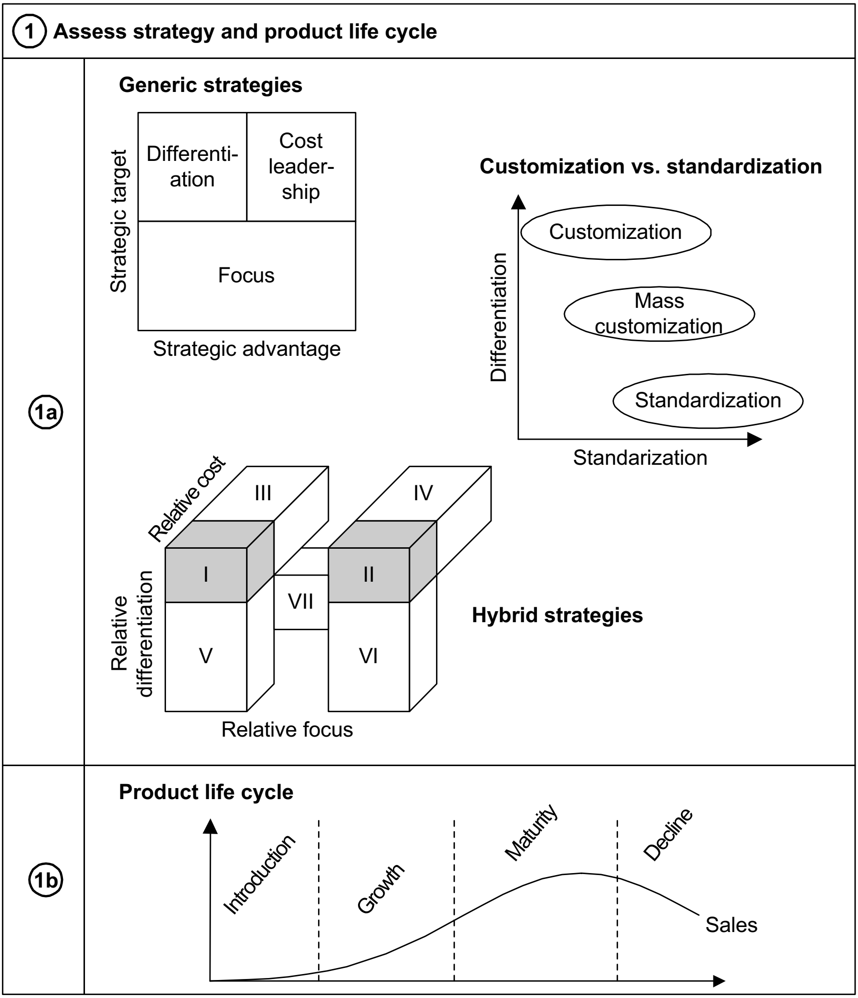


Figure 4.10 Summarizing depiction of strategy and product life cycle assessment[24]

---

[24] The graphics are based on the following sources: "generic strategies," Porter (1980, p. 39); "customization vs. standardization," Schuh and Schwenk (2001, p. 62); "hybrid strategies," Miller and Dess (1993, p. 565); "product life cycle," Kotler and Keller (2006, p. 322).

the product's life cycle assessment can precede the strategic considerations if this is found to be of advantage.

4.3 Product Complexity Assessment

4.3.1 Introduction

The second major step of the complexity management model leads us into the details of the product the model is applied to. If we recall the message of Figure 2.1, a product is subject to a myriad of external factors (external complexity) or, in a narrower sense, a long list of market requirements. To live up to these customer expectations, a product causes a certain amount of complexity within the company's value chain (internal complexity). These two fundamental aspects of complexity management provide the two dimensions that lie at the heart of the product complexity assessment step. However, an appropriate simplification had to be found since describing external and internal complexity is itself an inherently complex, if not impossible, task:

- External complexity is narrowed down to the *functionality* of the product's components, which reflects customer requirements (see Subsection 4.3.2).

- Internal complexity is represented by the degree of *physical complexity* of the product's components, a term that will be explained in detail in Subsection 4.3.3.

Functionality and physical complexity were chosen because they both provide a very simplified (and thus easy to handle) description of external and internal complexity, while still capturing the essentials. An endless stream of different possibilities exists, of course, to tackle the two dimensions. Here, the above two entities were selected for two main reasons. First, the derivation of functionality is based on the procedure outlined by target costing, a well-known tool widely employed in industry.[25] Second,

[25] See Subsection 3.3.2 for details on target costing.

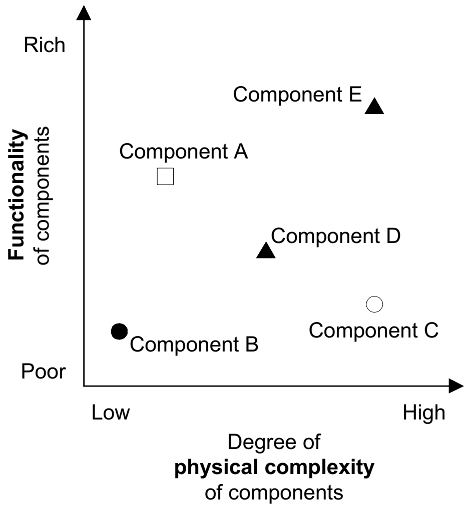


Figure 4.11 Complexity matrix for one product consisting of several components

the definition of physical complexity has its roots in systems theory, giving the model a thorough theoretical background.

If the functionality and physical complexity dimensions are combined to form a matrix, we arrive at what I call the *complexity matrix* (see Figure 4.11). The model is based on evaluating the *components*[26] that constitute the product under investigation. Their contribution to the product's functionality and their degree of physical complexity are calculated quantitatively. Once that computation has been performed, all components can readily be assigned their specific location within the complexity matrix. For instance, component A in Figure 4.11 provides a relatively high share of functionality while not causing much complexity to the product. Therefore, it translates exter-

[26] The term "component" is used in a very broad sense here. It designates any distinct region of the product and can either be one single piece part or be composed of several parts. Typically, components are easily distinguished from each other, especially in the case of modules. See Table A.1 in Appendix A for a definition of a component and related terms.

nal complexity in an effective way into the physical product – it is perceived as being important by customers without being the source of excessive internal complexity. Component C is the other way round: it is physically very complex without adding much customer value. Components B, D, and E are more balanced, i.e. their level of physical complexity roughly matches their contribution to functionality. The two axes in Figure 4.11 are labeled “functionality of components” and “degree of physical complexity of components,” respectively. For the sake of brevity, the axes will be called “functionality” and “physical complexity” in the remainder of this work (written in bold letters in Figure 4.11).

The following two subsections will give a detailed description of calculating the values for the functionality and physical complexity axes. An illustrative example will accompany us during the journey of developing the complexity matrix. This ensures that all explanations and calculations are well understood. The product I chose as an example is a ballpoint pen because this device does show a certain degree of complexity (consisting of several distinct components and several interfaces between the components), but is simple enough so that the basic arguments are not obscured. A sketch of the pen can be viewed in Figure C.1 in Appendix C.

4.3.2 Quantifying Functionality

The vertical axis of the complexity matrix weights the contribution of all components to the overall product’s functionality. A percentage is assigned to each component, reflecting its relative contribution. The sum of all component percentages, of course, adds up to 100 percent. But the starting point for the functionality evaluation is the product’s structure of functionality[27], which summarizes what the product must do from a customer perspective. Deriving all relevant functions and molding them into a hierarchical structure is a tedious task and requires several workshops and interviews

[27] See Chapter 2 for a detailed explanation of the structure of functionality, and Appendix A for the definition of the term.

with employees of many different functional areas, such as R&D, production, marketing, and sales. It would be best to involve customers to achieve a structure of functionality that really integrates customers' requirements. In most companies, this is either considered to be too costly or to consume too much time, or the customers do not see the benefit of participating in such an exercise.

The left portion of Figure 4.12 depicts the structure of functionality for the ballpoint pen, showing the hierarchy of functions and the percentages assigned to each function. "Provide good writing feel," for instance, accounts for 12.1 percent of the pen's overall functionality and is further split into two more detailed functions, contributing 5.9 and 6.2 percent, respectively. In such a way, every function's share of product functionality is determined. Next, the functions (and their percentages) must be deployed to the physical components by which they are carried out. This is exemplified in the right portion of Figure 4.12 for two components, showing how the function percentages are deployed to the front and rear housings. "Provide convenient size" is carried out by both housings to equal parts, giving both components 1.95 percent (3.9 / 2). "Enable easy attachment" is solely fulfilled by the rear housing, which adds up to 5.45 percent for that component (1.95 + 3.5). Apart from convenient size, the front housing is also responsible for preventing the fingers from being stained, but not exclusively (only to a degree of 40 percent): that function is also fulfilled by other components, indicated by the dashed line attached to the "prevent staining fingers" box. As the front housing carries out several more functions, the percentage shown (12.5%) is larger than the above discussion would suggest ($1.95 + 0.4 \times 5.8 = 4.27$).

Assigning weights to each function and working oneself down the hierarchy of the structure of functionality all the way to the functional elements represents only one possibility to assign functionality percentages to the components. Alternatively, the number of functional elements a component fulfills can be counted. In this case, every functional element is assumed to be of the same importance. This eventually also leads

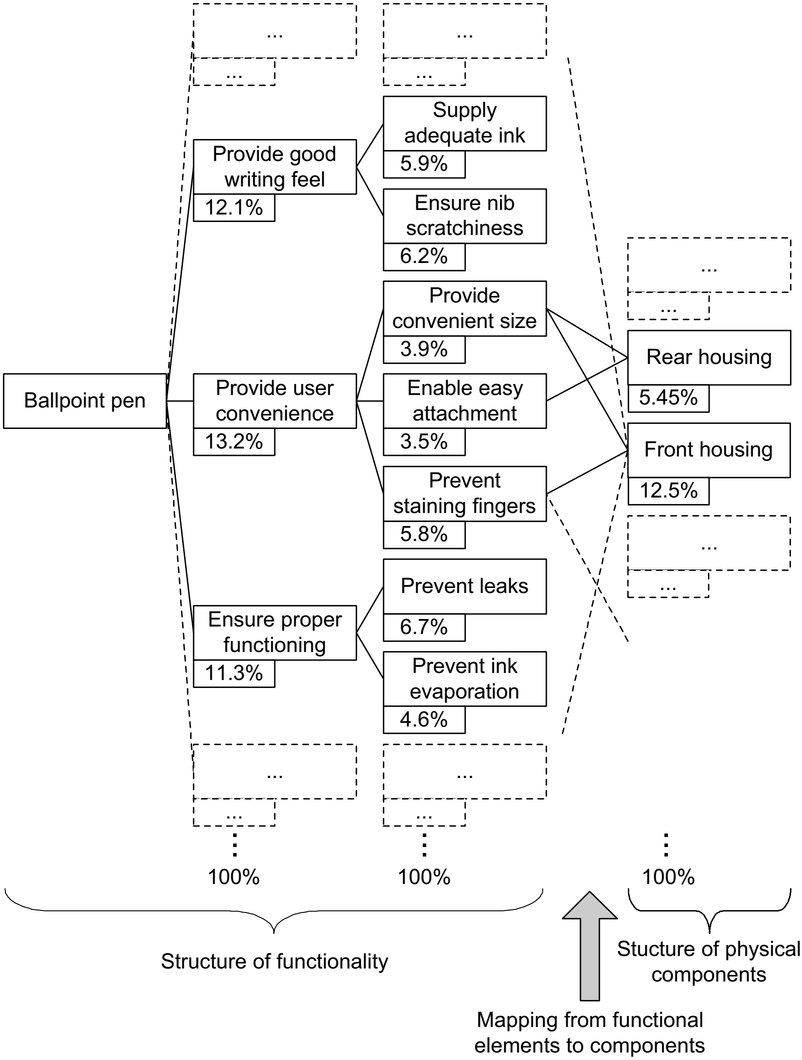


Figure 4.12 Excerpt of the ballpoint pen's structure of functionality and physical components with percentages assigned to functions and components[28]

[28] The functions and percentages are based on Tanaka (1989, pp. 62-65).

to a functionality percentage assigned to each component.[29] This procedure is especially suitable if the process of weighting functions is considered too tedious (i.e. saving time is more important than a very accurate calculation) or if the product has a very large structure of functionality and / or consists of many components.

The percentages for each component derived in the way described above provide the input for the functionality axis of the complexity matrix. Note that the procedure outlined here is very similar to the one introduced by Tanaka (1989) for target costing (see Subsection 3.3.2). The two main differences are as follows:

- *Displaying the functions.* Tanaka (1989) divided the functions into soft and hard functions and grouped certain soft functions together. Here, the functions are structured in a more hierarchic manner, displayed in several functional layers – just as it was introduced in Section 2.3. With Tanaka's method it is fairly difficult to assign percentages to each function because so many functions have to be handled at the same time. Giving a hierarchy to functions facilitates the distribution of percentages as only a few functions have to be considered simultaneously: the functions on the same hierarchical level sharing the same superior function.[30]

- *The computation's purpose.* The goal of target costing is to identify those components of a prototype that must be redesigned because their manufacturing cost is too high. The model presented here is concerned with restoring the balance between external and internal complexity within a product.

---

[29] For example, a component with ten functional elements receives double the percentage compared to a component responsible for only five functional elements.

[30] One disadvantage of displaying functions hierarchically becomes evident when a functional element (at the lowest level of hierarchy) cannot be fully assigned to one single superior function (termed sub-function in Figure 2.18). One option is to simply break the strict hierarchy and assign the functional element to two sub-functions. Another (more difficult) possibility is to divide the functional element into two separate functions. A non-hierarchic way of displaying functions does not pose such problems.

4.3.3 Quantifying Physical Complexity

4.3.3.1 Introduction

The horizontal axis of the complexity matrix weights all components of a product with respect to their contribution to the product's physical complexity. In other words, the farther right a component is located in the complexity matrix, the more complexity it causes within the product. Now what do I mean by physical complexity? To start with, go back to Figure 2.14, where the four dimensions of any system's complexity were defined: the number of elements ("many elements"); the diversity of elements ("many kinds of elements"); the number of relationships ("many relationships"); the diversity of relationships ("many kinds of relationships"). These are the dimensions – or *complexity drivers* – the model evaluates for each and every component of a product, only slightly adapted to the situation of manufactured products:

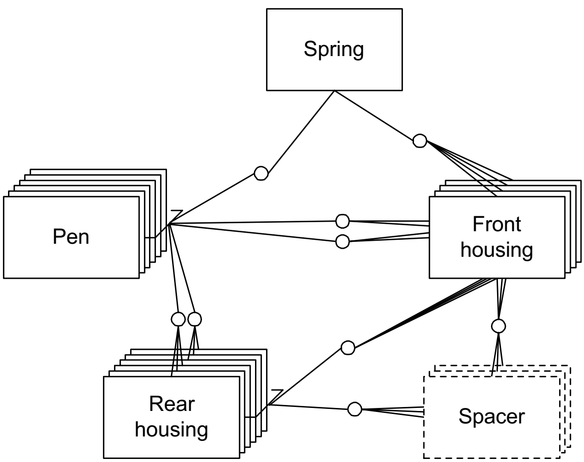


Figure 4.13 Visualization of physical product structure[31]

[31] The graphic is based on Rapp's (1999, p. 38) visualization of product structures.

- The *number of parts* of a component;
- The diversity of the component, i.e. the *number of variants;*
- The *number of interfaces* with other components and the environment;
- The diversity of interfaces, i.e. the *number of interface variants.*

If we take a closer look at the ballpoint pen, its product architecture can be visualized as shown in Figure 4.13. It can easily be seen that it consists of five distinct components: spring, pen, front housing, rear housing, and spacer. The components are symbolized by rectangles, while the interfaces between the components are small circles. Components that occur in several variations are represented by several rectangles stacked behind each other. For instance, the front housing has four distinct variants, while the spring does not show any variety (it is a standardized component). The same is true for interfaces: one circle means the interface never changes, and two circles designate two interface variants. In the latter case, depending on the component variants the interface connects, one or the other interface variant of the two is used.

4.3.3.2 Component Variety and Number of Parts

A necessary prerequisite to evaluating all of the ballpoint pen's components regarding their contribution to physical complexity, a clear understanding of the product's variety is needed. The *attribute-value matrix* is a very simple but effective tool to summarize all possible variations of a product. In the case of the ballpoint pen, the customer can choose from three attributes: color, surface, and length, which can take two (surface and length) or three (color) values (see Table 4.3). As all values can be freely combined, the ballpoint pen has a total of twelve variants ($3 \times 2 \times 2$).[32] Note that the

[32] If, for example, pens with rough surfaces never occurred in the long version (70 mm), the number of variants would be reduced to nine. Three out of the twelve combinations would in that case not exist: red / rough / 70 mm, blue / rough / 70 mm, and green / rough / 70 mm.

Table 4.3 Attributes and corresponding values for the ballpoint pen[33]

Attributes	**Values**		
Color	Red	Blue	Green
Surface	Smooth	Rough	
Length	50 mm	70 mm	

attribute-value matrix in Table 4.3 is structured in the same way as Table 2.1, where coffee-makers were considered and the term *full profile* was introduced. A green ballpoint pen measuring 70 mm in length with a smooth surface defines one of the ballpoint pen's twelve full profiles (indicated by the line in Table 4.3).

A further concept is introduced here to be better able to analyze a product's variety. Going back to Figure 4.13, one can see that the spacer has a dashed outline. This is because it does not occur in all variations of the ballpoint pen. As the spacer has three variants (shown by three dashed rectangles), it is not only an optional, but a variable optional component. The other components in Figure 4.13 are used in all versions of the product. This fundamental difference is shown in Figure 4.14, where a classification of components is given according to their occurrence and variability. A *standardized component* is used in all variants of a product in the same standardized form. Variety is introduced to the product by *variable components.* They occur in all variants of the product but in changing forms. A car always has an engine, but in different versions, e.g. different power levels, different fuel types (hybrid / diesel / gas), etc. If a component is not used in all product variants but can be added on customer demand, it is called an *optional component,* such as a car's sunroof. While an optional component is only used in one single version (if it is used), *variable optional components* occur in

[33] Note that the values of every attribute can be combined with any values of the other attributes. One possible combination (a full profile) is indicated by the line in Table 4.3.

several variants (if they occur). If an optional sunroof were available in two different materials (say metal and glass), it would be a variable optional component. Every component can now readily be classified according to this scheme by the following identifiers: S (standardized component), V (variable component), O (optional component), and VO (variable optional component). These IDs are shown in the second column of Table 4.4 for the ballpoint pen components.

Table 4.4 also lists the number of piece parts each component consists of. While the front housing, the spring, and the spacer are piece parts themselves, the rear housing is an assembly of five parts (clip, push button, housing, thrust tube, and spring) and the pen is composed of four parts (ink cartridge, ink, ball, and metal point). The last column of Table 4.4 refers to the attributes the components depend on, the attributes corresponding to those given in Table 4.3: color, surface, and length. For example, the rear housing is determined by color (three values: red, blue, and green) and surface (two values: smooth and rough). Therefore, the rear housing has six (3×2) variants, shown in the second but last column in Table 4.4. The number of variants for the other components can be calculated in the same way. Note that the spring does not depend

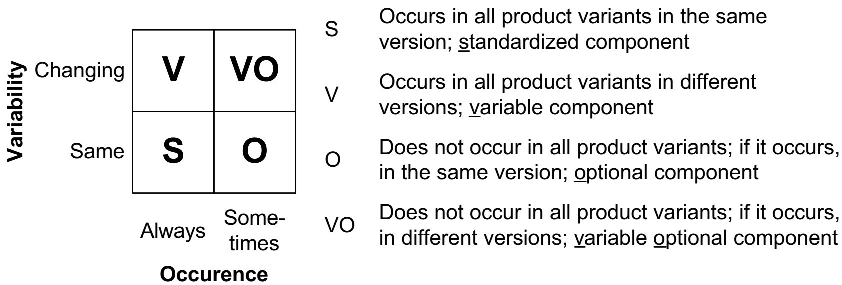


Figure 4.14 Component classification[34]

[34] The classification is based on Schuh (1989).

Table 4.4 List of components with corresponding characteristics

Component	ID	Number of parts	Number of variants	Depends on attribute
Front housing	V	1	4	Surface; length
Rear housing	V	5	6	Color; surface
Pen	V	4	6	Color; length
Spring	S	1	1	–
Spacer	VO	1	4	Color

on any attribute because it is a standardized component, thus exhibiting only one variant. Table C.1 in Appendix C gives a full listing of all component variants.

Note that optional components are counted as two variants (either present or absent). By either using it or leaving the component away, the number of product variants double. If a car manufacturer decides to offer all its variants of a particular product line with an optional sunroof, product variety (at the end-product level) doubles within that line. Based on an analogous reasoning, variable optional components are also counted as having one more variant than the physical component actually has. The spacer in Table 4.4 therefore has four variants although there are only three different types of spacers (red, blue, and green).

In some cases, the values of certain components for the two complexity drivers “number of parts” and “number of variants” are very large compared to the other components. These values would exert a disproportionately heavy influence on the complexity matrix if they were to be taken as the basis for the physical complexity calculations. Therefore, the logarithm with base ten is used in these cases to moderate the large differences. A component with many variants is still detected by the model as a highly variable component and can be distinguished from a component with few variants. But the advantage is that those components with large values do not get an unduly high physical complexity coordinate. Reference is made here to Appendix D.3, where the logarithm procedure is covered in more detail. Several other possibilities exist to circumnavigate the problem of large differences among complexity driver values. The product complexity assessment step of the complexity management model is designed as an open platform and can be adapted freely to the needs of its users.

4.3.3.3 Interface Variety and Number of Interfaces

So far, two of the four complexity drivers have been computed for all components: number of parts (third column of Table 4.4) and number of variants (fourth column of Table 4.4). The remaining two complexity drivers are concerned with the number and variety of the interfaces between the components. The tool I find most effective in analyzing interfaces is the *design structure matrix* (DSM). A DSM is a square matrix with identical row and column titles and displays the relationships between the elements of a system in a compact, visual, and analytically advantageous format (Browning, 2001, p. 292). Various applications exist for DSMs,[35] which Browning (2001, pp. 292-293) classified into component-based DSM, people-based DSM, activity-based DSM, and parameter-based DSM. Although the underlying rationale is the same in all four types, the component-based type is the DSM used in a product architecture context.

The DSM for the ballpoint pen example is shown in Figure 4.15, revealing which components share interfaces. Two numbers in a cell indicate an interface, while an empty cell means the two components are not connected. For instance, the front housing is connected to all of the other four components, while the spring has only two interfaces (with the front housing and the pen). The DSM can easily be derived from the physical product structure visualized in Figure 4.13, where the interfaces are displayed graphically. For complex products with many components and interfaces, the graphical

[35] Browning (2001, p. 293) reported that the DSM has found applications in the building construction, semiconductor, automotive, photographic, aerospace, telecom, small-scale manufacturing, factory equipment, and electronics industries. The correlation matrix (the "roof") of the QFD matrix (see Figures 3.4 and 3.5) exhibits the same principles as the DSM. Pimmler and Eppinger (1994) used the DSM to identify interactions between components and cluster them into major chunks. Also by means of the DSM, Baldwin and Clark (1999) analyzed the influence of design parameters and product attributes on each other and on the production process.

	Front housing	Rear housing	Pen	Spring	Spacer	**Number of interfaces**	**Number of interface variants**	**Average number of interface variants**
Front housing		1 1	1 2	1 1	1 1	**4**	**5**	**1.25**
Rear housing	1 1		1 2		1 1	**3**	**4**	**1.333**
Pen	1 2	1 2		1 1		**3**	**5**	**1.667**
Spring	1 1		1 1			**2**	**2**	**1**
Spacer	1 1	1 1				**2**	**2**	**1**

DSM Complexity calculations

Figure 4.15 Design structure matrix (DSM) and complexity calculations for ballpoint pen

product structure (as shown in Figure 4.13) proves to be cumbersome (even though it gives excellent insights for simple situations). Contrary to that, the DSM manages to cope with very complex and intertwined product architectures.

From the many variations of the DSM, one is especially noteworthy here. While the basic DSM versions simply indicate whether there is a relationship between two elements, Pimmler and Eppinger (1994) classified the relationships into spatial, energy, information, and material.[36] They also assigned weights ranging from –2 to +2, depending on the strength and importance of the relationship. As a result, every cell describing one relationship was actually divided into four sub-cells to accommodate all relationship types with their corresponding weights. I chose a similar way of dis-

[36] See Section 2.2 for more information on Pimmler and Eppinger's (1994) classification.

playing information about interfaces in the DSM version that I use in the complexity management model. Consider a cell in Figure 4.15 with two numbers (designating an interface):

- The number in the cell's upper left corner accounts for the number of interfaces between the two adjacent components. In many applications, the entry is one because commonly there is only one interface between two components.
- The number in the lower right corner indicates the number of interface variants.[37]

Adding the numbers horizontally leads to the number of interfaces and the number of interface variants for every component. These are the two remaining inputs needed to compute every component's contribution to physical complexity. Note that the DSM in Figure 4.15 is symmetrical and only the shaded cells must be filled in to determine the others. The last column of Figure 4.15 divides the number of interface variants by the number of interfaces, which equals the average number of variants per interface. This number instead of the absolute number of interface variants will be used in the complexity calculation below.[38]

Values for the "number of interfaces" and "number of interface variants" complexity drivers can show large differences within one product – just as described for the "number of parts" and "number of variants" complexity drivers (see 4.3.3.2). In these

---

[37] As an illustration of the interface variety numbers in Figure 4.15, look at Figure 4.13. The interface between pen and rear housing has two variants (represented by two circles), while the spring and front housing have a standardized interface (only one circle). The corresponding numbers in Figure 4.15 (in the lower right corners) are also two and one, respectively.

[38] The absolute number of interface variants is strongly dependent on the number of interfaces. A component with four standardized interfaces also has four interface variants because the numbers are added up. The interfaces do not vary, though – they are standardized. By considering the average number of variants per interface, the calculation can be decoupled from the number of interfaces. The front housing in Figure 4.15 has four interfaces and five interface variants, while the pen also has five interface variants but only three interfaces. The variability of the pen's interfaces is therefore larger than the front housing's: the front housing's interfaces have 1.25 variants on average and the pen's 1.667.

cases, the logarithm with base ten is used to moderate the large differences. Consult the last paragraph in 4.3.3.2 and Appendix D.3 for more details.

4.3.3.4 Calculating Physical Complexity

The four drivers of physical complexity have now been derived for all components of the ballpoint pen. They are collected from Table 4.4 and Figure 4.15 and displayed as an overview in Table 4.5. The physical complexity can now be calculated for every component according to Equation 4.1.

$$C_i = \alpha \cdot \frac{N_{e,i}}{N_{e,\max}} + \beta \cdot \frac{V_{e,i}}{V_{e,\max}} + \gamma \cdot \frac{N_{r,i}}{N_{r,\max}} + \delta \cdot \frac{V_{r,avg,i}}{V_{r,avg,\max}} \qquad \text{Equation 4.1}$$

The symbols in Equation 4.1 are defined as follows:

C_i Physical complexity of component *i*

$N_{e,i}$ Number of elements (parts) constituting component *i*

$N_{e,max}$ Maximum occurring number of elements (parts) within a component of the product

$V_{e,i}$ Variety (number of variants) of component *i*

Table 4.5 Complexity drivers for the ballpoint pen

Component	ID	Number of parts	Number of variants	Number of interfaces	Average number of interface variants
Front housing	V	1	4	4	1.25
Rear housing	V	5	6	3	1.333
Pen	V	4	6	3	1.667
Spring	S	1	1	2	1
Spacer	VO	1	4	2	1

$V_{e,max}$ Maximum occurring variety (number of variants) of a component of the product

$N_{r,i}$ Number of relationships (interfaces) of component *i*

$N_{r,max}$ Maximum occurring number of relationships (interfaces) of a component of the product

$V_{r,avg,i}$ Average relationship variety of component *i* (average number of interface variants per interface)

$V_{r,avg,max}$ Maximum occurring average relationship variety of a component of the product (maximum average number of interface variants per interface)

The four coefficients α, β, γ, and δ ensure that all four terms in Equation 4.1 receive the same weighting. Normally, all complexity drivers are considered to exert the same importance on physical complexity – thus the need to introduce weighting coefficients. If for any reason some complexity drivers are more important than others, the coefficients can be tailored to the requirements of that specific application. As every component's contribution to physical complexity is required to be a value between zero and one (i.e. $0 \leq C_i \leq 1$), the sum of α, β, γ, and δ must be one. For full details of the calculation procedure, reference is made here to Appendix D, where the calculation procedure is shown in detail. To give some insight into the results of the computation made in Appendix D for the ballpoint pen, the front housing's physical complexity is shown as an example in Equation 4.2.

$$C_{front\ housing} = 0.33207 \cdot \frac{1}{5} + 0.22770 \cdot \frac{4}{6} + 0.2277 \cdot \frac{4}{4} + 0.21252 \cdot \frac{1.25}{1.67} = 0.60531 \quad \text{Equation 4.2}$$

In two cases, one (or more) of the four complexity drivers is canceled from the calculation in Equation 4.1:

- If all components show the same value for one complexity driver, it does not make sense to integrate the respective values because they do not contribute to any differentiation of the components with respect to their physical complexity. This treatment can virtually only become necessary with the "number of parts" complexity driver: if the components are individual piece parts, the entry for this complexity driver is one for all components.

- If the product under investigation is very complex, determining one (or more) of the complexity drivers can become nearly impossible. The "number of interface variants" complexity driver is especially prone to this problem. In that case, the respective complexity driver must be excluded from the calculation due to a lack of data.

To grasp the essential issues, the above introduction of calculating physical complexity can be summarized as follows. Every component's contribution to physical complexity is derived based on the four complexity drivers. The values represent the components' coordinates on the abscissa of the complexity matrix and must be understood as relative – they do not reflect an absolute level of complexity.[39]

4.3.4 Drawing the Complexity Matrix

The previous two Subsections 4.3.2 and 4.3.3 explained the calculation of the functionality and physical complexity axes in detail. Now that the coordinates of all components are available (see Table 4.6), the complexity matrix can readily be drawn (see Figure 4.16). While the values for physical complexity are calculated as shown above, note that the functionality coordinates are estimated values and do not reflect an evaluation of a real-life ballpoint pen by means of workshops and interviews. For the purpose of the example presented here, however, the numbers shown in Table 4.6 suffice fully.

The complexity matrix shown in Figure 4.16 condenses in a handy, graphical format a large amount of information about a product regarding its functionality provided to customers and its physical complexity. All components the product consists of are

[39] Note that the values of the complexity matrix' ordinate (the functionality axis) are relative, too.

Table 4.6 Functionality and physical complexity coordinates of the ballpoint pen

Component	Functionality	Physical complexity
Front housing	20%	0.61
Rear housing	26%	0.90
Pen	40%	0.88
Spring	12%	0.35
Spacer	2%	0.46

mapped in the matrix with respect to the two fundamental dimensions of external and internal complexity (though in a simplified way). The matrix reveals how well the ballpoint pen's components translate the functions they must fulfill into their physical representation. The spacer, for instance, is not important from a customer's view but

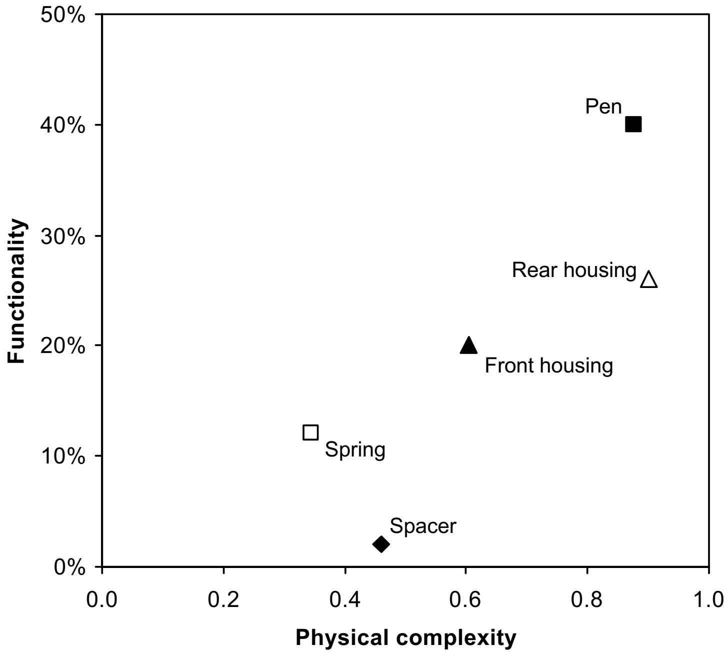


Figure 4.16 Complexity matrix for the ballpoint pen example

does cause a certain degree of complexity within the product. As opposed to that, the front housing's contributions to functionality and physical complexity are more balanced. The next section will introduce ways to interpret and use the information provided by the complexity matrix while also taking into account strategic and product life cycle considerations.

4.3.5 Summary of Product Complexity Assessment

The previous subsections presented the second step of the complexity management model – product complexity assessment. It is the model's centerpiece and is depicted in Figure 4.17, summarizing the key elements of the model's second step in a brief and

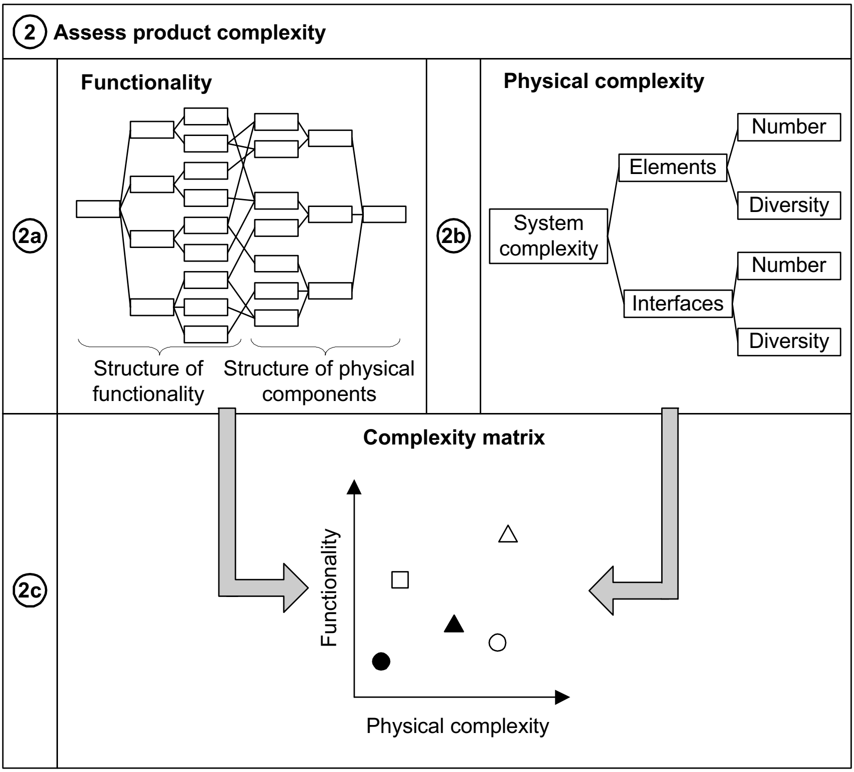


Figure 4.17 Summarizing depiction of product complexity assessment

schematic way. Note that there is no prescribed sequence of going through steps 2a and 2b, i.e. quantifying physical complexity (2b) can precede the functionality step (2a). The combination of functionality and physical complexity within the complexity matrix is the most important novelty for theory and practice presented by the model.

4.4 Deriving Guidelines for Action

4.4.1 Introduction

The previous section introduced how the complexity matrix is derived from functionality and physical complexity inputs. This section combines the information about the product that is graphically condensed in the complexity matrix with the strategic and product life cycle considerations from Section 4.2. The result is a set of *basic norm strategies* that are based on the components' location within the complexity matrix. These basic norm strategies are then adjusted to also take into account *strategic considerations* and *product life cycle considerations.* Figure 4.18 illustrates the procedure.

The basic norm strategies are applied component-wise and depend on the component's location within the complexity matrix, i.e. its relation of functionality (contribution to customer satisfaction) and physical complexity (contribution to complexity costs). Supplemented with strategic and life cycle influences, the basic norm strategies provide guidelines for action as to how product architecture can be optimized. As a guideline for deriving norm strategies, the complexity matrix is divided into four quadrants (see Figure 4.19):

- *Lucky strike.* Components in this quadrant are very important to customers as they add a high level of functionality to the product. At the same time, they cause a comparatively low degree of complexity in the product. These components represent the ideal case (a great deal of customer satisfaction for marginal complexity) and, therefore, their quadrant is termed the lucky strike. Empirical research shows that components rarely occur in this quadrant (see Chapter 5).

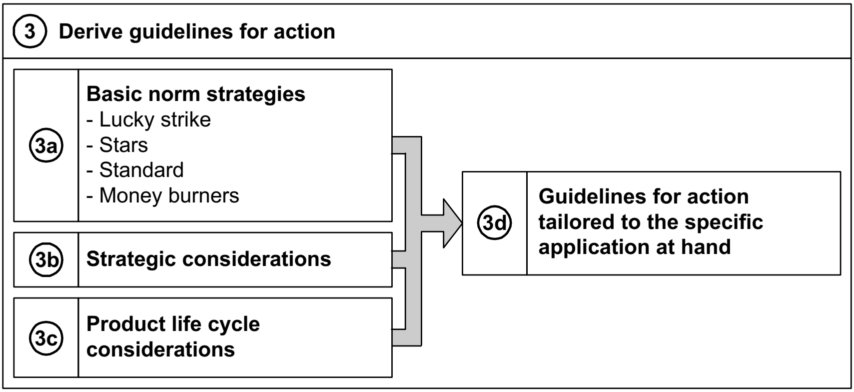


Figure 4.18 Steps involved in deriving guidelines for action

- *Stars*. Components in the stars quadrant are – just like the lucky strike components – very important from a customer perspective. However, they are the source of considerable complexity. Because they provide a high degree of functionality, these components are "allowed" to cause their share of complexity. The empirical results presented in the next chapter show that virtually all high-functionality com-

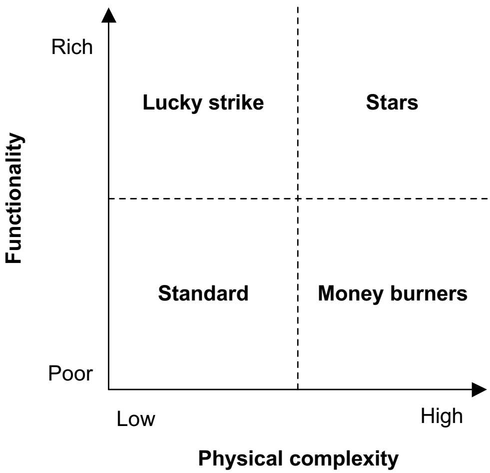


Figure 4.19 Definition of four quadrants within the complexity matrix

ponents are located in this quadrant. As they are mainly responsible for making the product attractive and successful on the market, they are called the “stars.”

- *Standard.* The lower left quadrant houses components neither contributing much to functionality nor to physical complexity. These components often do not receive much attention from customers and normally represent standardized modules and commodity components (screws, bolts, etc.) – therefore the name “standard.” While one might be inclined to perceive them as being unimportant, they should actually be viewed as the product’s backbone – silently holding everything together. The flesh on the bones is provided by the stars (or the lucky strikes), giving a face to the product.

- *Money burners.* Components in the lower right quadrant do not provide much functionality but still are the cause of a high degree of physical complexity. Obviously, this mismatch must be the source of major concern and, therefore, these components must receive by far the most attention when optimizing product architecture. As they incur complexity costs that are not reflected by a corresponding customer benefit (thus the name “money burners”), ways must necessarily be found to shift these “bad guys” out of the lower right quadrant.

Since providing a certain degree of functionality necessitates – under normal circumstances – the drawback of increased complexity, components on the complexity matrix’ diagonal[40] can be regarded as those components that successfully balance external with internal complexity. For example, a module that fulfills many important functions is expected to be more complex than a standardized commodity component and, therefore, is “allowed” to be located farther to the right in the complexity matrix. As long as components are in the vicinity of the diagonal, external and internal complexity are in a balanced state. The lucky strikes and the money burners deviate from the diagonal in a positive and a negative sense, respectively, the latter case re-

[40] With the complexity matrix’ diagonal I designate the diagonal from the lower left corner to the upper right corner.

the diagonal in a positive and a negative sense, respectively, the latter case receiving special attention.

4.4.2 Basic Norm Strategies

As was seen in the previous subsection, of the four quadrants, the money burners must receive the largest share of attention, and a most thorough effort must go into resolving the problems pointed out by the complexity matrix. Nevertheless, norm strategies are presented here for all quadrants.

4.4.2.1 "Lucky Strike" Quadrant

Because components in this quadrant already are very near to ideal, optimizing their situation is restricted to finding the few ways to further decrease the contribution to physical complexity and increase their functional appeal. However, these objectives are difficult to achieve and have only a limited effect because it is challenging enough to bring components into this quadrant, let alone to enhance their situation within the quadrant. It must be kept in mind that the marginal optimization costs are at a high level for an already very well designed component. Figure 4.20 illustrates the directions of action that are – in principle – possible.

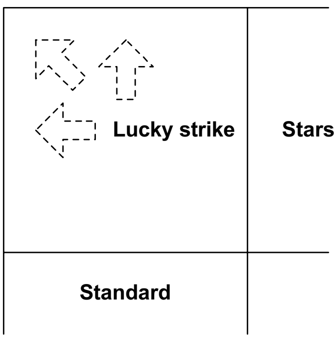


Figure 4.20 Directions of action for the "lucky strike" quadrant

The empirical results of Chapter 5 show that components are rarely ever located in the upper left quadrant, which underscores the difficulty of developing components that are highly functional and do not entail a correspondingly high level of internal complexity. Table 4.7 and Figure 4.27 at the end of Section 4.4 summarize the most promising strategies to optimize product architecture with respect to components located in the "lucky strike" quadrant.

4.4.2.2 "Stars" Quadrant

Although "stars" components are located in the vicinity of the matrix diagonal, there is much room for improvement. High importance from a customer perspective is traded with a considerable degree of complexity added to the product. The main objective, therefore, is to decrease physical complexity while maintaining components' high level of functionality, i.e. pushing them as far to the left in the matrix as possible, in some fortunate cases all the way into the lucky strike quadrant. Sometimes, the functionality coordinate of star components can also be increased, shifting these components to the left and upwards. Figure 4.21 illustrates the directions of action in the stars quadrant.

Now, what concrete form do measures take to achieve the strategy of shifting components leftward (and, if feasible, upwards) in the stars quadrant? The first step is to recall the definition of physical complexity used in the model. A component is defined as physically complex when it consists of many parts, occurs in many varieties, and shares many and strongly varying interfaces with other components. Any effort to decrease a component's contribution to physical complexity must take into account this definition.

Generally, shifting a star component to the left can be achieved by rethinking the component's setup, such as standardizing the interfaces and subcomponents, using

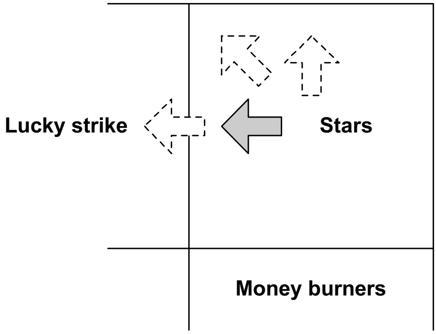


Figure 4.21 Directions of action for the “stars” quadrant

fewer piece parts, deciding whether to drop certain low-sales variants[41] and reducing the interdependencies between components by employing *decoupled interfaces.*[42] All these measures positively affect the four complexity drivers and help to shift a component to the left. It should also be considered whether components can be merged, which in some cases allows for combining functionality in one single component, while at the same time physical complexity (of the merged component) is not (or only slightly) increased. A further possibility to be investigated is splitting a component (e.g. into distinct modules), which can reduce complexity at the level of the newly formed components (albeit at the cost of usually decreased functionality of the individual components).

[41] As was mentioned in Section 4.2, dropping certain product variants can conflict with obligations and contracts with customers (see footnote 22). Canceling low-sales variants should therefore be treated with caution.

[42] Decoupled interfaces allow for changing one component while eliminating the need for also altering the other component. Contrary to that, two components share coupled interfaces if a change made to one component necessitates changing the other component, too (Ulrich, 1995, p. 423). See boxed example below.

Table 4.7 and Figure 4.27 at the end of Section 4.4 summarize the most promising strategies to optimize product architecture with respect to components located in the "stars" quadrant.

4.4.2.3 "Standard" Quadrant

Similar to "stars" components, "standard" components are located close to the matrix diagonal, which means that external and internal complexity are well-balanced. Nevertheless, ways must be sought to optimize the situation here, too. When keeping in mind that standard quadrant components' main feature is their low physical complexity, it becomes clear that this strength must be reinforced even more. Efforts towards that end should primarily be dedicated to further shifting components to the left, while increasing functionality often is not a promising option because standardized and commodity components are usually defined by low perceived customer benefit. Figure 4.22 illustrates the directions of action in the standard quadrant.

The same possibilities for shifting components to the left and upward apply for the standard quadrant as for the stars quadrant discussed above. Redesigning and standardizing interfaces as well as reconsidering all component variants (and, possibly, dropping some) must be looked at in detail. Questions of whether fewer parts could be used and whether decoupled interfaces are a feasible alternative (effectively raising the

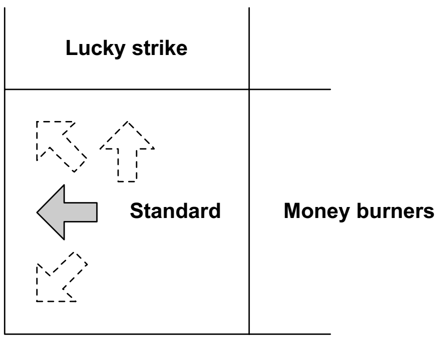


Figure 4.22 Directions of action for the "standard" quadrant

product's degree of modularity) have to be addressed. The difference to the stars quadrant is the focus of the optimization effort. While reducing physical complexity in the stars quadrant must necessarily maintain (or increase) the level of functionality (because stars are the primary source of customer benefit), altering standard components is not hindered by that constraint. The focus is on creating the least complex components possible, no matter whether functionality increases or decreases slightly. Therefore, merging or splitting components is a much more promising strategy in the standard quadrant than for stars. Since components in the standard quadrant are not usually perceived by customers (or to a much lesser extent), changes here are made in a much less sensitive environment. After all, the main purpose of "standard" components is to carry out product functions barely noticed by customers.

Table 4.7 and Figure 4.27 at the end of Section 4.4 summarize the most promising strategies to optimize product architecture with respect to components located in the "standard" quadrant.

4.4.2.4 "Money Burners" Quadrant

Components in the lower right quadrant are the major source of concern as they cause a high level of physical complexity without a corresponding contribution to functionality. Due to this mismatch, these components must receive undivided attention and a fierce effort to alleviate the highly problematic situation. One of the prime benefits of the complexity management model stems from its ability to point out the product's "hot spots" that need optimization because they incur costs while not creating customer benefit.

In terms of the complexity matrix, it is absolutely imperative to find ways to shift money burning components out of their detrimental quadrant – be it by an upward or leftward movement, or both. Figure 4.23 illustrates these directions of action in the "money burners" quadrant. Successful strategies to make money burners budge must take into account the four complexity drivers and consider the reasons for the low functionality contribution:

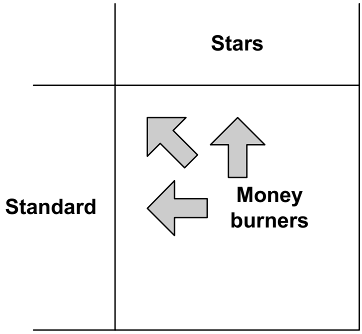


Figure 4.23 Directions of action for the "money burners" quadrant

- *Number of parts.* Does the component consist of an inappropriately large number of parts? Can the same functionality be provided with fewer piece parts? Decreasing the number of parts reduces, among others, assembly time, logistical effort, and design costs. These factors all contribute to complexity costs, often without letting the customer see an additional value. When reducing the number of piece parts in a component, one must bear in mind that at least in principle, this can result in a shift towards a more integral product architecture (e.g. one piece part responsible for several functions instead of only one). Therefore, the pros and cons must be weighed against each other carefully.

- *Number of variants.* Does the component show a variety that is not demanded by customers? Can variety on the component level somehow be reduced without affecting the product's overall variety (which is what customers care about)? Are there any low-sales variants that can be considered for dropping from the portfolio and, hence, eliminate the need for the respective component variants? If a variety reduction for a "money burner" component can be achieved, complexity costs can be lowered in a similar way as for a reduction in the number of parts. An evaluation of reducing component variety must always consider the effects of such a product change, especially where customers are concerned. Almost never are there easy trade-offs to be made in the area of product variety.

- *Number of interfaces.* Are there any interfaces with other components that do not carry an important function? Can the purpose of these interfaces be taken over by other interfaces? Does the component's (or the product's) entire concept have to be redesigned to attain a higher degree of modularity, which would lead to fewer and standardized interfaces between components or modules? Any additional interface entails constraints on the product's layout and causes complexity costs because of the attention it requires from R&D, manufacturing, logistics, and procurement. Blindly eliminating interfaces is not a viable option, of course. The functions of all interfaces and the effect of canceling them must be analyzed carefully, providing the basis for making the respective product architecture decisions.

- *Number of interface variants.* When considering interface variety, can any interfaces be identified that have two or more variants because they depend on the variants of the components they connect?[43] Are there any options to eliminate this dependency? Can other solutions be found to standardize interfaces? The concept of coupled and decoupled interfaces has been briefly introduced above in the coverage of the "stars" quadrant (see footnote 42 of this chapter). Interface decoupling is a very effective way of reducing interface variety and, even more importantly, reducing the interdependencies between components and, hence, eliminating the need to alter other components when actually only one is the source of the change. The box below presents an example of interface decoupling in a trailer. A further opportunity for positively influencing interface variety is the concept of monofunctional and multifunctional interfaces (Rapp, 1999, pp. 34-35). *Monofunctional interfaces* can accommodate several variants of one element, but not different elements with varying functions.[44] Contrary to that, many different elements with a variety of

[43] In the ballpoint pen example of Subsection 4.3.3, the interface between the rear housing and the pen has two variants (see Figure 4.13). The interface depends on the attribute "surface": in the case of a smooth surface, the interface between rear housing and pen looks different from the rough case because of the larger diameter required in the latter case.

[44] Monofunctional interfaces are encountered more frequently in practice than multifunctional interfaces. An example of a monofunctional interface is an automobile's cylinder block. Many different

functions can be connected to *multifunctional interfaces.*[45] According to Rapp (1999, p. 34), standardized interfaces are characterized by the advantage of reducing the product's complexity and increasing the combinability of component variants. However, such interfaces must be designed to handle the largest or most critical occurring variant, causing a tendency to over-engineer interfaces and increase costs.

Example: Coupled and decoupled interfaces in a trailer

The bed and the box of a trailer can be connected to each other according to the two example interfaces shown in Figure 4.24. In the left graphic exhibiting a coupled interface, the vertical gap in the box connection slot must be changed whenever the bed's thickness is changed (e.g. due to a change in the cargo load rating). A coupled interface therefore embodies a dependency between the two components. In the graphic to the right in Figure 4.24, the decoupled interface eliminates this dependency and allows the box to accommodate all bed thicknesses.

Of course, even with decoupled interfaces there still is – to a certain extent – a coupling between two components, as "there is almost always a change that can be made to one component that will require a change to the other component" (Ulrich, 1995, p. 423). In practical terms, however, interface coupling is only concerned with changes that modify the component in some useful way. Source: Ulrich (1995, pp. 423-424).

---

cylinder heads can be connected to the block, but to function properly, a cylinder head (and no other component) must always be attached to the block (Rapp, 1999, pp. 34-35).

[45] A socket can handle a large variety of different appliances, such as a TV set, lamp, drill, etc. Drawers, shelves, cupboards, etc. can be connected to the vertical rods of a book shelf system by the same interface (Rapp, 1999, p. 35). USB interfaces are another example of multifunctional interfaces.

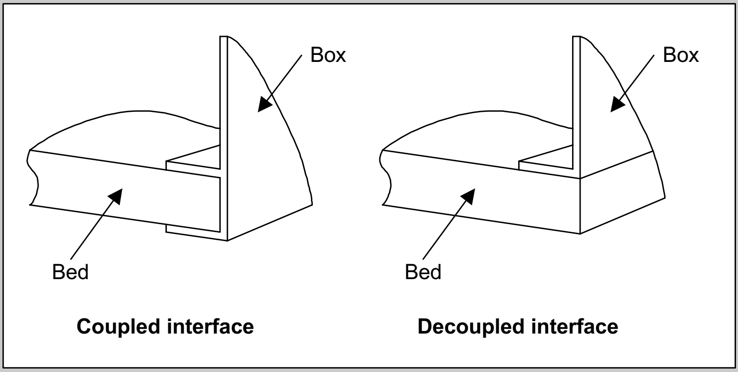


Figure 4.24 Coupled and decoupled interfaces in a trailer; source: Ulrich (1995, p. 423)

- *Functionality.* The functions carried out by money burner components must be identified and ways found to either fulfill the functions with less physical complexity (see bullet points above) or increase the importance for customers, i.e. raise components' appeal to customers. In this way, one or two functions are added to the product's structure of functionality, redistributing all functionality contributions of the product's components. If possible, functions from other components can be shifted to a money burner, which increases its functionality coordinate in the complexity matrix (but decreases the others'). Shifting functions often also involves a corresponding slight increase in physical complexity, i.e. the money burner wanders upwards and to the right, a movement that should be avoided. Changing a component's functionality coordinate usually is even more challenging than altering its physical complexity. Not only are product-internal issues involved, but also customer-specific requirements and sensitive decisions concerning product strategy.

Apart from treating the functionality and physical complexity axes separately, two further strategies are presented here briefly that affect both axes simultaneously:

- *Merging.* When two (or more) money burner components are merged, their functions are laid together, which increases the merged component's functionality coordinate. Depending on the product architecture, the physical complexity of the newly formed component can be reduced (in the fortunate case) or increases even more, in which case the components should not be merged. A rule that is applicable in general cannot be given here; advantages and disadvantages must both be considered anew for every product.[46]

- *Splitting.* The complement to merging components is splitting one component into two (or more) new components. In many cases, the decrease in functionality (due to distributing a given number of functions onto two or more components) is outweighed by a significant reduction in the new components' physical complexity. As underscored above, there are no general rules of thumb here. The product architecture must be analyzed in every application, laying the foundation for understanding the implications of component splitting.[47]

Table 4.7 and Figure 4.27 at the end of Section 4.4 summarize the most promising strategies to optimize product architecture with respect to components located in the "money burners" quadrant.

4.4.3 Influence of Strategic Considerations

In Subsection 4.2.1, several frameworks and models were introduced to assess a product's strategic surrounding. Here, the influence of these considerations on the basic

---

[46] Note that merging components leads towards a more integral product architecture as the mapping from the structure of functionality to the structure of physical components is affected by the merging process.

[47] Note that splitting components – if done intelligently – can lead towards a more modular product architecture as the mapping from the structure of functionality to the structure of physical components is affected by the splitting process. The degree of modularity increases only if those regions of a component form the new set of components that are each responsible for a particular fuction. For example, assume that component A consists of the three distinct regions A1, A2, and A3. If A1 fulfills function F1, A2 function F2, and A3 function F3, then splitting component A into the three new components A1, A2, and A3 makes the product architecture more modular.

norm strategies is presented and how they have to be adjusted to a company's specific strategic situation. The basic norm strategies presented in the previous subsection will not, however, be turned upside down by strategic considerations and will retain their basic structure.

The first and most important influence stems from the two strategic extremes of customization and standardization or, in Porter's (1980) terms, differentiation and cost leadership:

- Under a *customization* strategy, a firm markets products that are tailored to the very needs of each customer and draws its competitive advantage from differentiating itself from competitors. Hence, any envisaged action that is based on the complexity matrix must give priority to the *functionality* axis. Changes made to the product architecture must ensure that those functions that provide the product its competitive edge are not compromised under any circumstances.

- A company pursuing a *standardization* strategy strives to achieve a cost level that is as low as possible by seeking to profit from economies of scale, i.e. producing vast amounts of identical products. Catering to individual customer needs is not an objective, and the competitive advantage is primarily based on the intriguingly low cost level. Therefore, when applying the complexity management model, the *physical complexity* axis enjoys precedence. Reducing complexity and hence costs is at the forefront of every cost leader; providing superior functionality is but a second tier activity.

Figure 4.25 illustrates the above influences of differentiation and cost leadership on the implications drawn from the complexity matrix and depicts the primary focus of product architecture optimization under the two strategies. Note that the relative focus dimension plays an unimportant role in terms of the complexity matrix. Porter's (1980) focus strategy does not require yet another interpretation; differentiation focus entails an emphasis of functionality, and cost focus of physical complexity.

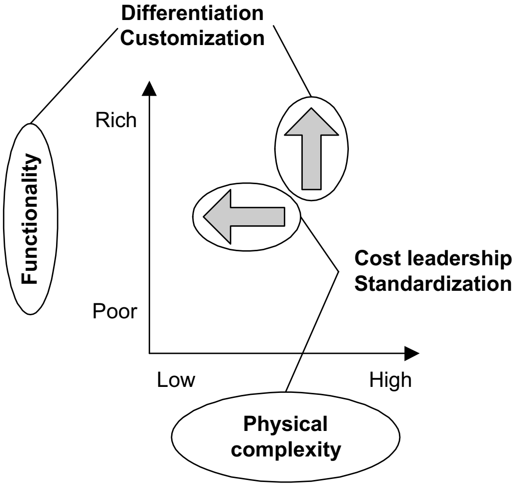


Figure 4.25 Influence of strategic considerations on complexity matrix

Hybrid competitive strategies, as their name suggests, require a blend of the above adjustments and do not make things easier when deriving guidelines for action from the complexity matrix. In the case of sequential hybrid strategies (like the outpacing strategy),[48] priority should be given to the two axes, one after the other, according to the sequence chosen by the hybrid strategy. For simultaneous hybrid strategies (such as mass customization), a real trade-off must take place, weighing the two often contradicting interests of providing supreme customer value and saving costs against each other. Cookbook-like advice cannot be given here, and adjusting the basic norm strategies to hybrid competitive strategies must necessarily take into account the specific application at hand. Even for the difficult situation of a hybrid strategic setting, the next subsection is able to provide answers for products depending on the phase of their life cycle they are in.

[48] See 4.2.1.3 for a definition of sequential and simultaneous hybrid strategies.

4.4.4 Influence of Product Life Cycle Considerations

Besides the strategic influences on the basic norm strategies discussed in the previous subsection, the life cycle phase of the investigated product also plays an important role in identifying appropriate strategies to optimize product architecture. Subsection 4.2.2 introduced the concept of product life cycle (PLC), subdividing a product's market phase into four further phases: introduction, growth, maturity, and decline. Depending on the phase the product is in, either the complexity matrix' functionality axis enjoys precedence, or the physical complexity axis is of priority, or a real trade-off between the two has to be found. Note that this approach is very similar to the influence of strategic considerations discussed in the previous subsection.

To appreciate which complexity matrix axis is of prior importance in what life cycle phase, an understanding of the driving objectives in every phase must be achieved:

- *Introduction.* The primary objective in the introduction phase is to gain customer acceptance and to successfully enter the market. This is the product's phase where differentiating itself from competing products is most important in order to be perceived on the market. Limiting costs is important, but by far not as important as product differentiation.

- *Growth.* Once the product is starting to become known by customers, the company's focus shifts towards rapidly increasing market share. Sales volume rises and, as a result, efforts to decrease the cost level (thanks to economies of scale and experience curve effects) slowly gain importance.

- *Maturity.* The primary objective here is to earn a profit, while competition is fierce due to the maturing market. Both the pressure for profits and competition intensify the need to further decrease costs. In this phase, efforts to differentiate the product are less important for reaping profits than reducing costs.

- *Decline.* Towards the end of a product's life, companies try to generate as much cash with the product as possible. Normally, costs are closely monitored and investments kept at a minimum (i.e. no further differentiating efforts). If a relaunch is planned (product line extension, added features, etc.) or any other measure to extend the product's life cycle, differentiation and, thus, focusing on functionality become once again a central issue.

The objectives and the primary focus (differentiation and / or cost) for the above four life cycle phases are shown in a condensed form as a postulate in Figure 4.26. The differentiation and cost lines in the lower portion of Figure 4.26 represent an idealized and simplified way of describing which of the two is more important in a specific life cycle phase. Any given application will, of course, deviate from the example shown in Figure 4.26 in one or another way. Whenever differentiation is at the forefront, the functionality axis in the complexity matrix enjoys precedence. If costs are the primary focus, the physical complexity axis is of priority. While in the introduction phase functionality is underscored, priorities start shifting in the growth phase, where a real trade-off between the two axes must take place. In the course of the maturity phase, the physical complexity axis becomes dominant and retains that position into the decline

Phase	Introduction	Growth	Maturity	Decline
Objective	Market entry	Market share	Profit	Cash

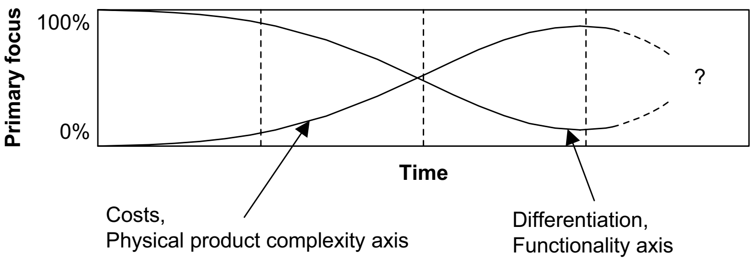


Figure 4.26 Relative importance of complexity matrix axes during the product life cycle[49]

[49] The figure is based on a discussion with Prof. Fahrni (F. Fahrni, personal communication, June 2, 2006).

phase. If product improvements (or an entire overhaul of the product concept) are scheduled during the decline phase to launch a new line extension and to boost sales, the balance between the two axes can once again be changed.

4.4.5 Summary of Guidelines for Action

In the previous three subsections, the basic norm strategies were derived from the complexity matrix, and the influence of strategic and product life cycle considerations was also shown. The detailed guidelines for action presented above are summarized in Figure 4.27and Table 4.7.

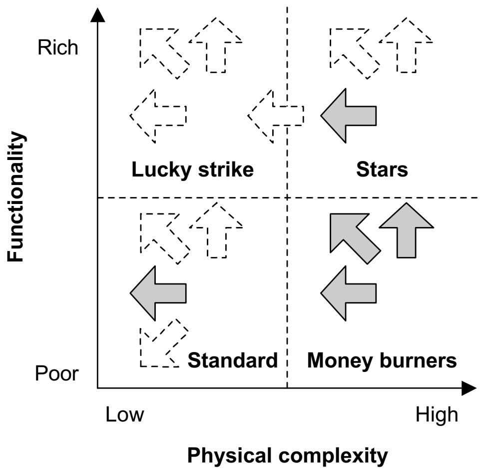


Figure 4.27 Summary of basic norm strategies in the complexity matrix

Table 4.7 Summary of guidelines for action

Situation		Guidelines for action	
Basic norm strategies	Quadrants: - Lucky strike - Stars - Standard - Money burners	Physical complexity axis	
		Decrease number of parts	- Provide same functionality with fewer parts - Identify and eliminate unnecessary parts
		Decrease component variety	- Identify component variety not contributing to customer value - Eliminate low-sales variants
		Decrease number of interfaces	- Use few and standardized interfaces - Increase degree of modularity
		Decrease interface variety	- Standardize interfaces - Increase degree of modularity - Introduce decoupled and / or multi-functional interfaces
		Functionality axis	
		Increase functionality	- Add new functions, i.e. increase appeal to customers - Shift functions among components
		Both axes	
		Merging	- Adds up functionality - Makes product architecture more integral
		Splitting	- Can decrease physical complexity of new components - Can make product architecture more modular
Strategic influence	Differentiation, customization	Priority for functionality axis	
	Cost leadership, standardization	Priority for physical complexity axis	
	Hybrid competitive strategy	Trade-off between the two axes	
PLC influence	Introduction phase	Priority for functionality axis	
	Growth phase	Trade-off between the two axes	
	Maturity phase	Trade-off between the two axes; priority for physical complexity axis at end of phase	
	Decline phase	Priority for physical complexity axis; product relaunch: trade-off between two axes	

4.5 Summary of Complexity Management Model

This chapter gave an introduction to the complexity management model with its three steps strategy and product life cycle assessment, product complexity assessment, and deriving guidelines for action. While the first step draws on existing concepts, the second step presents a new way of quantifying product complexity. The product complexity assessment step is therefore the model's centerpiece and its main contribution to complexity management theory. The third step combines the previous two steps to form recommendations as to how product architecture can be optimized. Figure 4.28 shows a summarizing depiction of the complexity management model.

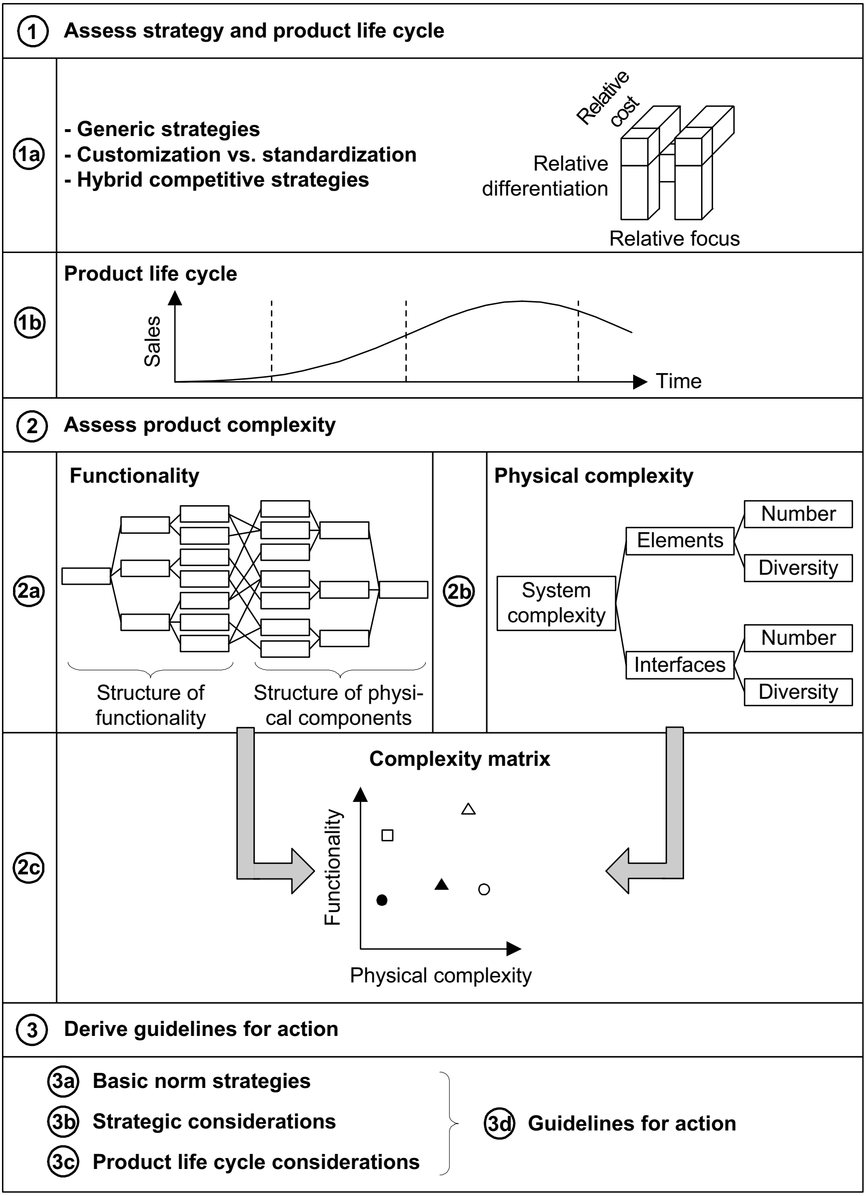


Figure 4.28 Summary of complexity management model

5 Case Studies

One of the tragedies of life is the murder of a beautiful theory by a gang of brutal facts.

Benjamin Franklin.[1]

5.1 Introduction

The research performed in this work is based on four case studies. Their purpose is to confirm the model, find its limitations, and show opportunities for future research. The case studies are combined with action research as the situation in every case is optimized and the effect of the action taken is assessed. The cases apply the complexity management model presented in the previous chapter to four different products:

- Railroad signal;
- Liquid handling platform;
- Process industry compressor;
- Railroad switch lock.

The first case study is used as a lead case and investigates the railroad signal. It emphasizes the model's product complexity assessment step to give further insight into the procedure employed to derive the complexity matrix and to clarify open questions that may have arisen while reading Chapter 4. Therefore, the first case study (Section 5.2) is longer than the others. In the remaining three case studies, for the sake of brevity, only the essentials of the complexity assessment step are presented. The fo-

1 As cited in Mayer (1993, p. 35)

cus there clearly lies on the guidelines for action resulting from the complexity management model. Discussing the research achievements, presenting the model's limitations, and outlining ideas for future work will also receive ample coverage at the end of every case study.

5.2 Railroad Signal

5.2.1 Introduction to the Case

5.2.1.1 Company Profile

This case study was conducted in a multinational company's regional division employing 600 people. Roughly 90% of total sales are generated in the country of the regional division, while the remaining 10% are exported. The company focuses on the rail automation business and maintains a broadly diversified product portfolio.

The product considered in this case study is part of the railroad signal product portfolio, which boasts a market share of around 90% in the country of the regional division, though with a slightly declining tendency. The railroad signals are divided into the product lines mainline signals, tunnel signals, and subsidiary signals and are characterized by a large product variety. Next, the situation met at the beginning of the case study is described in detail.

5.2.1.2 Situation at Beginning of Case Study

The research of this case study focused on an R&D project aimed at standardizing the tunnel and subsidiary signal products. These types of signals were poorly standardized and completely tailored to individual customers. Therefore, they had a wide product variety which proliferated over several years and in some cases even decades. The objective of the R&D project was to launch a completely new product line for tunnel signals and several types of subsidiary signals, based on a *signal module* that is characterized by a high degree of standardization. Mainline signals were not considered by the project since they were already based on a well established product platform.

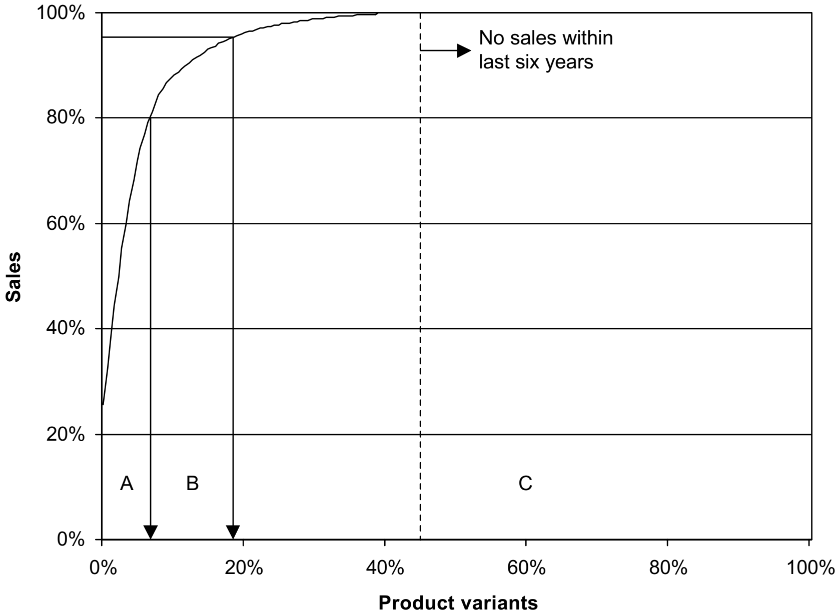


Figure 5.1 ABC analysis of the railroad signal sales[2]

Figure 5.1 shows an ABC analysis[3] of the product variants considered by the R&D project. 80% of the units sold were based on only 7% of all variants, while 55% of all variants had never been sold in the period considered in the ABC analysis (six years). Although most variants sold poorly, the strategy chosen by the company – to provide virtually all signal varieties demanded by customers – dictated that the broad product

[2] The A variants are defined as those variants that account for 80 percent of sales (7 percent of all product variants); the B variants as the next 15 percent of sales (from 80 to 95 percent); the C variants the rest.

[3] An ABC analysis as used here classifies every product variant with regard to its importance relative to all variants of a product. The importance can be measured by units sold, sales dollars, manufacturing costs, etc. In Figure 5.1, the fraction of sales volume is used. Frequently, it can be observed that a small number of products (or product variants) generate a large portion of sales (Schönsleben, 2002, p. 459). This is often associated with the *Pareto principle* (or the *80-20 rule),* which states that 80 percent of activities or consequences are due to 20 percent of the causes. The classification of A, B, and C products (or product variants) does not follow a strict rule; often, A products are defined as the top 20 percent, and B products the next 30 to 40 percent, and C products the rest.

range has to be maintained. This entailed a disproportionate amount of costs incurred by the low-sales variants because they were produced in small batches (often in batch sizes of one unit). Furthermore, they still caused disproportionately high expenses (e.g. when product updates were performed and all designs had to be changed) even though they did not generate much revenue.

The situation described above led to the decision to base the tunnel and subsidiary signal product lines on a module that can be used in all signal variants of any type. Variety should only be provided if required by customers (e.g. color, voltage, etc.). The idea was to assemble a complete signal from several modules and a signal plate which the modules are fixed to. By designing the product line in such a way, the underlying rationale was to push the degree of standardization to a high level, while all customer requirements can still be catered to.

The first concept of the new signal module presented by the design team showed a large percentage of components being standardized, while the variety relevant for customers was generated only by a few parts. The front and rear casings, for example, were completely standardized; they protected the inner parts of the module, such as the incandescent light bulb and the wiring. The frosted and colored glasses were chosen depending on the application and on customer specifications. An optional shade was riveted to the front casing. Figure 5.2 exhibits a schematic sketch of the signal module.

The next subsection shows how the complexity management model is used to assess the first concept of the signal module described above. Unlike the other three case studies, the focus here is placed on the model's product complexity assessment step to shed light on the details of calculating product complexity and deriving the complexity matrix. The objective of applying the complexity management model to the signal module is to further enhance the prototype and reduce the inherent product complexity. The estimated manufacturing costs of the module are still too high, and several details in the construction are over-engineered. The implications pointed out by the model are expected to serve as guidelines to the team developing the module, and to

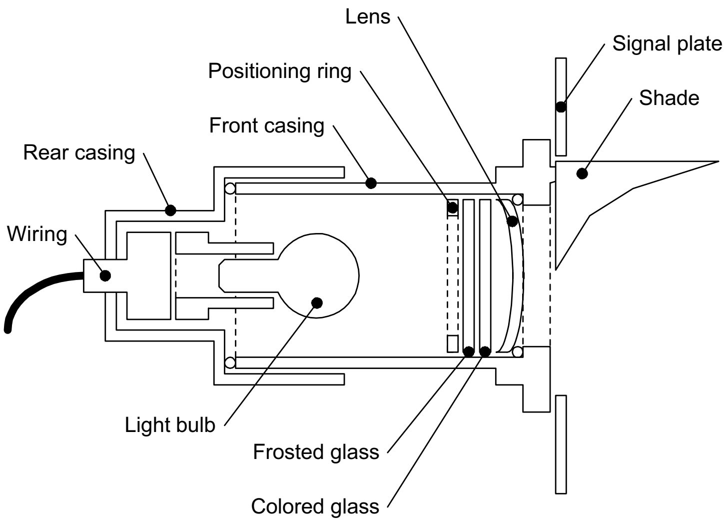


Figure 5.2 Schematic sketch of the signal module

give product managers and personnel of other functional areas an idea of the prototype's ability to reduce complexity.

5.2.2 Application of Complexity Management Model

5.2.2.1 Strategy and Product Life Cycle Assessment

As already mentioned above, the company's product strategy for railroad signals consists of fulfilling virtually any customer need and designing products to any particular application. In the Porter framework, differentiation is the competitive strategy here, as highly customized solutions are truly designed to order. The R&D project's attempt to introduce a certain level of standardization shifts the competitive strategy towards a hybrid strategy. After all, a major objective of the project is to significantly reduce costs, which is aimed at maintaining the product's market share and boosting its competitive edge. This shift from exclusive customization to a strategy also driven by strong cost awareness reflects a change in an enterprise formerly only concerned with providing highly functional products, no matter what they cost.

The terminology by Lampel and Mintzberg (1996) provides an excellent visualization of the direction targeted by the R&D project (see Figure 4.7). The company's status quo clearly fits that of thin industries. By introducing new signal modules, the level of standardization is increased for the products and processes involved, while the transaction (e.g. pre-sales activities, contacts between sales representatives and customers, etc.) remains customized. Thus, the situation is pushed towards that of tailoring industries or even menu industries.

All product lines of the railroad signal portfolio have shown a very long maturity phase with more or less constant and in some cases slightly decreasing sales.[4] Designing new modular-type signals can be classified as a product relaunch as the product's functions remain virtually unaltered – only the design and product architecture change. Table 5.1 summarizes the strategy and product life cycle assessment.

5.2.2.2 Product Complexity Assessment

As a first step in achieving the complexity matrix for the signal module, a workshop was organized resulting in the structures of functionality and physical components and the necessary information about the signal's variety. Further interviews and discussions with relevant employees revealed the missing items in the workshop data. This subsection describes how the functionality and physical complexity axes of the complexity matrix were derived and how the data was used to determine the location of every signal module component in the complexity matrix.

An excerpt of the structure of functionality of the signal module indicating the importance of each functional element on every hierarchical level is shown in Figure 5.3. The importance of the detail functions ranges from 0.6% to 7.6%. All components that

[4] Note that in the railroad business, life cycles for some products span several decades. In such circumstances, the maturity phase can easily take 10 or 20 years.

Table 5.1 Railroad signal: summary of strategy and product life cycle assessment

Criterion	**Railroad signal case**
Competitive strategy	From pure differentiation (fully customized) towards hybrid strategy
Strategy classification	Shift from thin industry to tailoring or menu industry, i.e. shift of products and processes from customization to tailored customization or customized standardization; transaction remains fully customized
Product life cycle	Relaunch within maturity phase

carry out a specific functional element are identified and each is assigned a fraction representing the component's importance in providing the respective function. For example, the elemental function "prevent dusting" (relative importance 6.3%) is carried out by the two following components: shade (33.33%), and lens (66.67%). Thus, the shade receives 2.1% (0.3333 × 6.3%) and the lens 4.2% (0.6667 × 6.3%) from the "prevent dusting" function. This is done for every functional element and results in a percentage assigned to each component. These numbers are eventually summed up for every component and result in a total percentage reflecting the component's relative importance for providing the overall product's functionality. The lens, for instance, provides 11.3% of the signal module's functionality. The result for all components is shown in Figure 5.3. The values found in this way form every component's coordinate on the functionality axis.

Now that the functionality axis coordinates are known, the physical complexity axis coordinates remain to be determined. The variants of the signal module are described by the attribute-value matrix shown in Table 5.2. From a product variety perspective, the module is fully characterized by four attributes, arithmetically leading to a number of 120 (5 × 4 × 2 × 3) possible variants. However, the actual number of variants is only 20 since not all values can be randomly combined (e.g. the combination of "red" and "very frosted" is not required by customers). Note that the attribute-value matrix in Table 5.2 is structured in the same way as Table 2.1, where coffee-makers were considered and the term *full profile* was introduced. A possible combination of values (one full profile) is indicated by the line in Table 5.2.

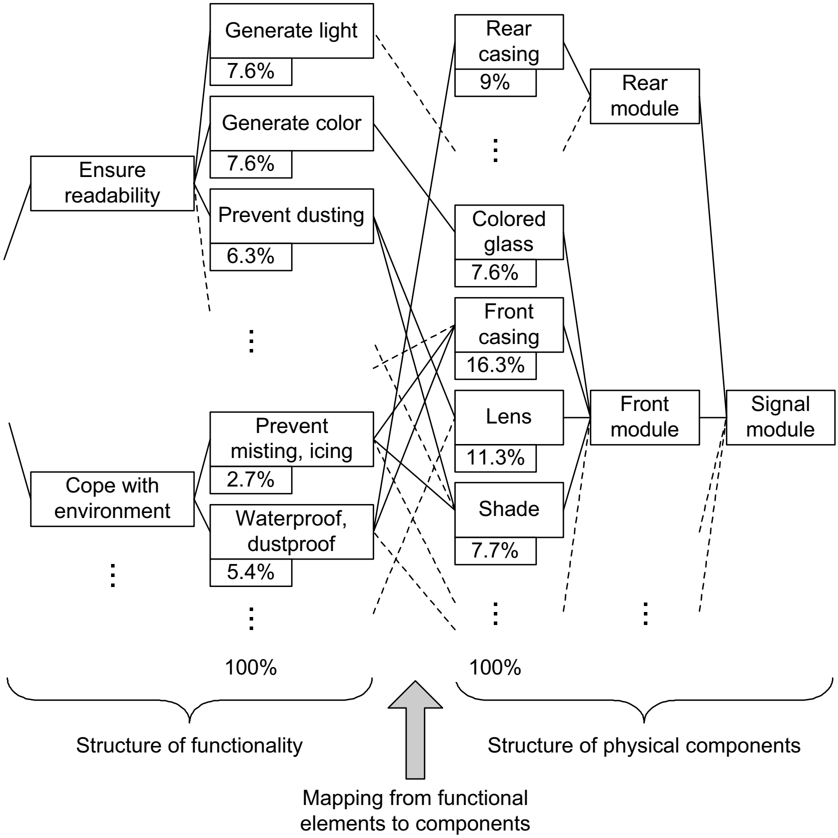


Figure 5.3 Product architecture of railroad signal module (excerpt)

In a next step, a variety analysis of the components constituting the signal module is performed. As described in Subsection 4.3.3, all components are categorized with respect to four complexity drivers: (1) number of parts, (2) number of variants, (3) number of interfaces, and (4) number of interface variants. As all components of the module are individual piece parts, the first complexity driver (number of parts) is not

Table 5.2 Attribute-value matrix of the railroad signal module[5]

Attributes	Values				
Color	red	orange	green	yellow	white
Voltage / power	12 V / 20 W	40 V / 20 W	220 V / 40 W	220 V / 60 W	
Shade length	short	long			
Frosted glass	very frosted	frosted	not frosted		

taken into account by the calculation.[6] The second complexity driver (number of variants of each component) is defined by the attributes the component depends on. Both complexity drivers are shown in the list of components in Table 5.3. Note that nine out of all 16 components constituting the signal module are standardized parts, reflecting the high degree of standardization of the signal module. The shade is an optional component and is used if customers order a long shade. (If they want a short shade length, the front casing has an integrated mini-shade, which eliminates the need for an additional component.) Note that the shade is counted as two variants because it is an optional component, even though it only has one variant (see Table 5.3).[7]

The classification of the components with regard to the third and fourth complexity drivers (number of interfaces and interface variants with neighboring components) draws on the design structure matrix (DSM) of the module, shown in Table 5.4. Two numbers at the intersection of two components indicate an interface. The upper left number in a cell indicates the number of interfaces between the two components. In the railroad signal module example, the entry is one for all interfaces. The lower right

[5] Note that the values of every attribute can be combined with several values of the other attributes. One possible combination is indicated by the line in Table 5.2. See 2.1.2 and 4.3.3.2 for more information on the attribute-value matrix.

[6] Subsection 4.3.3 gave an explanation for this case of all components having the same value for a complexity driver. For the sake of completeness, both Table 5.3 and Table 5.5 list the “number of parts” complexity drivers.

[7] Refer to Subsection 4.3.3 for more information on why the number of variants for optional components is counted as two.

Table 5.3 List of components with corresponding characteristics (excerpt)[8]

Component	Symbol	ID	Number of parts	Number of variants	Depends on attribute
Colored glass	△	VO	1	5	Color
Frosted glass	□	O	1	2	Frosted glass
Lens	○	S	1	1	–
Positioning ring	◇	VO	1	3	Frosted glass
Front casing		S	1	1	–
Rear casing		S	1	1	–
Shade		O	1	2	Shade length

number refers to the number of variants of the interface. Note that the interface between a standard and an optional part (e.g. front casing and shade) has two variants – either the interface exists or it does not. The environment is included in the DSM as well (not shown in Table 5.4), since two components have an interface with the signal plate and the interlocking system, respectively, which adds further complexity to the product. The total number of interfaces and the total number of interface variants are shown for each component in the third and second but last columns of Table 5.4. The last column shows the average number of variants per interface (number of interface variants divided by number of interfaces).

The four complexity drivers that are used to calculate the physical complexity axis are summarized for a selection of components in Table 5.5. Every component's coordinate on the complexity matrix abscissa can now readily be calculated according to Equation 4.1. As mentioned above, all components are single piece parts and thus all have an entry of one in the "number of parts" column. The coefficient α in Equation 4.1 is therefore set at zero because the "number of parts" complexity driver does not

8 The components printed in boldface in Table 5.3 will be considered in the derivation of guidelines for action.

have an influence on the relative physical complexity. The other coefficients are calculated based on the procedure outlined in Appendix D. The result is $\beta = 0.41$, $\gamma = 0.38$, and $\delta = 0.21$. Note that the sum of β, γ, and δ is one. The physical complexity coordinates for a selection of components is shown in the right column of Table 5.6.

Now that all necessary data have been gathered (summarized in Table 5.6), the complexity matrix for the railroad signal module can be drawn as shown in Figure 5.4. The four quadrants are indicated as discussed in Chapter 4, revealing that nine out of 16 components are located in the lower left quadrant, only one (the lens) in the "lucky strike," two in the "stars," and four in the "money burners" quadrant. The next subsection investigates the optimization potential in the product architecture based on the result depicted in the complexity matrix of the signal module.

Table 5.4 Design structure matrix (DSM) for the railroad signal module (excerpt)[9]

	Colored glass	**Frosted glass**	**Lens**	**Positioning ring**	Front casing	Rear casing	Shade	...	Number of interfaces	Number of interface variants	Average number of interface variants
Colored glass		1 2	1 2	1 2	1 2				4	8	2
Frosted glass	1 2		1 2	1 2	1 3				4	9	2.25
Lens	1 2	1 2		1 2	1 1				4	7	1.75
Positioning ring	1 2	1 2	1 2		1 3				4	9	2.25
Front casing	1 2	1 3	1 1	1 3		1 1	1 2		9	15	1.667
Rear casing					1 1				4	4	1
Shade					1 2				1	2	2
⋮								⋱			

[9] The components printed in boldface in Table 5.4 will be considered in the derivation of guidelines for action.

Table 5.5 Complexity drivers for the railroad signal module (excerpt)[10]

Component	Symbol	ID	Number of parts	Number of variants	Number of interfaces	Average number of interface variants
Colored glass	△	VO	1	5	4	2
Frosted glass	□	O	1	2	2	2.25
Lens	○	S	1	1	4	1.75
Positioning ring	◇	VO	1	3	4	2.25
Front casing		S	1	1	9	1.667
Rear casing		S	1	1	4	1
Shade		O	1	2	1	2
⋮			⋮	⋮	⋮	⋮
Maximum value			1	5	9	2.25

Table 5.6 Functionality and physical complexity coordinates (excerpt)

Component	Symbol	Functionality	Physical complexity
Colored glass	△	7.6%	0.77
Frosted glass	□	1.6%	0.54
Lens	○	11.3%	0.42
Positioning ring	◇	0.4%	0.63

[10] The components printed in boldface in Table 5.5 will be considered in the derivation of guidelines for action.

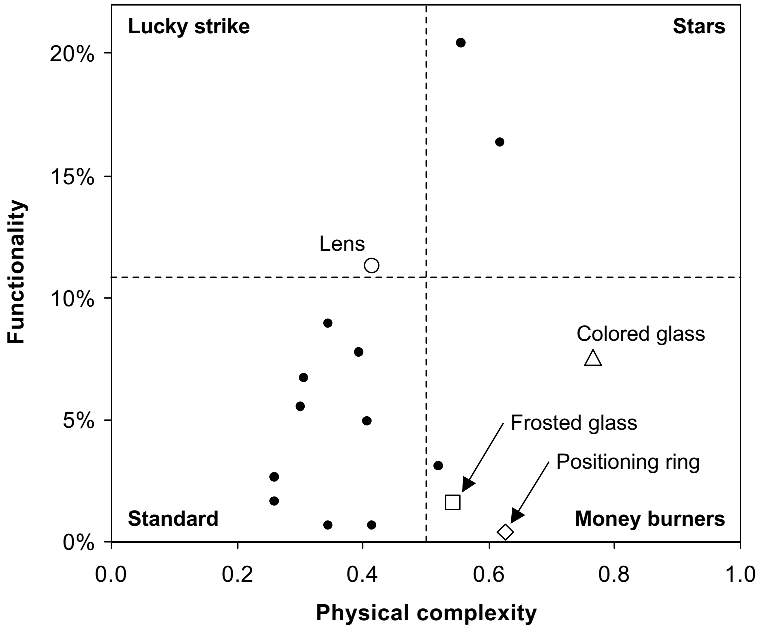


Figure 5.4 Complexity matrix for the railroad signal module with emphasis on four components (highlighted in Table 5.5 and listed Table 5.6) out of 16 components in total

5.2.2.3 Deriving Guidelines for Action

The task of enhancing the situation depicted in the complexity matrix of Figure 5.4 involves a wide variety of possibilities from which the most promising are presented here. The changes that can basically be applied to the product architecture start with merely relocating a component within the matrix (e.g. adding functionality, or decreasing physical complexity). But they can go all the way to merging several components, or eliminating a certain component by distributing its functions to other components. Section 4.4 presented in detail the options at hand to influence components' location in the complexity matrix and, thus, optimize product architecture.

In this case study, the company's strategy clearly is to provide highly customized products and maintain a competitive edge based on its ability to differentiate itself from competitors through its profound product and market knowledge and by offering services additional to the actual product. Therefore, the functionality aspect must not be compromised and enjoys precedence over attempts to decrease physical complexity when deriving guidelines for action. Furthermore, the importance of the functionality axis is boosted even more when considering the fact that a product relaunch is being planned. When a product in its maturity phase is being relaunched, the functionality aspect gains importance compared to the need to reduce costs (see Subsection 4.4.4). Nevertheless, the R&D project's objective is to develop a modularized product that is standardized as much as possible and reduces the level of product complexity. In terms of Figure 4.4, this is paramount to a shift from pure customization towards mass customization. The complexity matrix' abscissa should hence be monitored closely as well.

The component at the far right of Figure 5.4, the colored glass, must receive special attention as the need to change its present location in the "money burners" quadrant is evident. It causes considerable complexity to the module without providing a correspondingly high share of the product's functionality. Moving it upward, i.e. by adding functionality, is not very effective as this would negatively affect other components (decrease the functionality coordinate of those components) because functions are merely redistributed. The colored glass's high value in physical complexity stems not only from its five variants, but is also due to the high average number of interface variants (see Table 5.5). By merging the lens and the colored glass, the number of interfaces and interface variants can be reduced significantly. Additionally, the merged component moves vertically upward in the matrix as the functionality of the lens and the colored glass is combined in one single component. On a physical level, combining the lens and the colored glass essentially means producing colored lenses, which does not pose any particular problems.

A second component in the "money burners" quadrant causing trouble is the positioning ring (see Figure 5.4). A possible solution to its conversion it from a varying optional component (ID: VO) to an optional component (ID: O) is to always use a frosted glass instead of a light bulb with frosted surface. This eliminates the need for the wide version of the positioning ring (which is used in those cases where no frosted glass is needed in the module). In this way, the element variants of the positioning ring are reduced to two (present and absent, as for all optional components) and the interface variants are decreased, as well. Such an optimization slashes the abscissa of the positioning ring by roughly 20% (from 0.63 to approx. 0.5), moving it just into the "standard" quadrant. Furthermore, the frosted glass is a standard part after the optimization and no longer an optional part.

In the "standard" quadrant, where the majority of the standardized parts are located, several combinations of components to single junks seem promising. Many of these components share the same functions but still are designed as individual piece parts in the first prototype. As a consequence, they need to be welded together in the course of the assembly process of the signal module, incurring additional costs. By altering the design of these components, the complexity in the design and production process can be further reduced. In one case, three formerly separate components can be merged to one single component as they are barely perceived by customers and share several identical functions. In the second case, two components responsible for the electric wiring should be laid together because the on-site assembly of the two separate components causes a considerable increase in through time. (The complete outsourcing of the manufacturing process might also be considered a viable option in both cases described: the two newly formed components could be purchased as single components ready for assembly in the signal module.)

5.2.3 Result of Optimization

When applying the above guidelines for action to the railroad signal module, the result of the optimization process can be visualized by the complexity matrix of the new product architecture (see Figure 5.5). It can be seen at first glance that the situation has been enhanced greatly. The combination of lens and colored glass has moved to the "stars" quadrant, the colored glass no longer being a "money burner." The positioning ring has made a jump to the left into the "standard" quadrant. Thanks to the simplified architecture, also the frosted glass has altered its location from the lower right to the lower left quadrant. When comparing Figure 5.5 with Figure 5.4, all four “money burners” were successfully shifted out of their undesirable quadrant. However, it must be said that the product architecture optimization could only be performed by sacrificing a “lucky strike” component – the lens.

The recommendations regarding strategic and product life cycle aspects were able to be followed. The functionality axis was not compromised by any of the measures, and several components received an increase in functionality because they were combined.

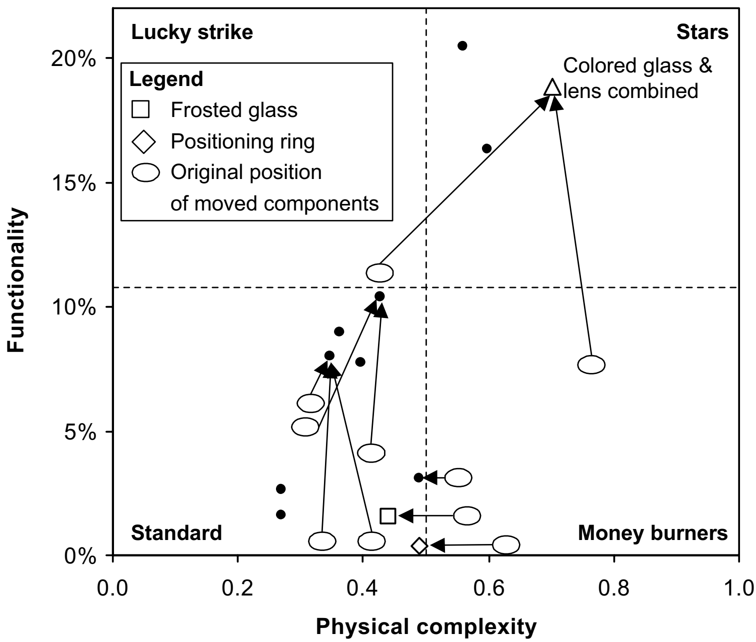


Figure 5.5 Complexity matrix for the railroad signal module after the optimization process with emphasis on 3 components out of 15 components in total

5.2.4 Discussion

The previous subsection showed that after applying the complexity management model, the railroad signal's product architecture was significantly optimized. The module no longer contains components that excessively contribute to physical complexity while not providing a significant share of the module's functionality. The reduced complexity in product architecture (less component variety, fewer interfaces, etc.) implies that the development, manufacturing, and logistics processes for the railroad signal are simplified and that complexity costs in general can be saved. At the same time, the product was not altered from a customer perspective – all required functions are still integrated in the module, and customers can choose from the same product variants. In summary, the balance between functionality and physical complexity was restored by first pointing out those components that cause a mismatch between the external and internal complexity dimensions. Then, guidelines for action were given that successfully optimized the railroad signal module's product architecture.

As already argued in Section 4.4, merging components makes product architectures more integral (i.e. decreases their degree of modularity). The optimization presented in this case study draws heavily on combining components, which somewhat offsets the R&D project's original goal to create a highly modular product.[11] A further drawback is that the model's advice was viewed as slightly static in nature by some users. While the amount of work needed to collect all necessary data for the complexity matrix is low compared to other complexity management tools, and even though the information contained in the complexity matrix is regarded as very valuable (especially knowing the product's "bad components"), the model's advice concerning the architectural changes are very much confined to the existing product concept. Thus, radical changes are not encouraged by the model; it is more appropriate for "fine-tuning" activities.

[11] It must be said, though, that modular product architectures are not "good" per se. They often are the source of higher development and direct per unit costs. See Subsection 3.3.5 for more information on this subject.

In the course of the case study, it became apparent that comparing several products by means of the complexity matrix would create great value to product designers, marketing managers, etc. A product manager, for instance, could then gauge his / her product against other products with regard to its customer benefit and complexity generation within the enterprise. A "complexity benchmark" could then be established that everyone would strive to achieve. This is not possible in the model's present form because the data condensed in the complexity matrix is relative in nature (each component's positioning relative to the others). Translating such ideas into action would require a new version of the model that does not only provide relative information for one product.

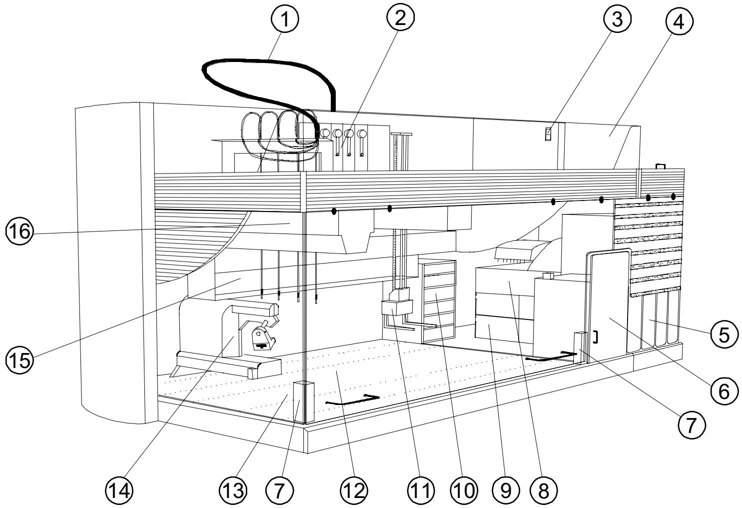
5.3 Liquid Handling Platform

5.3.1 Introduction to the Case

5.3.1.1 Company Profile

The second case study was conducted at a company in the life sciences supply industry. It specializes in developing, producing, and distributing solutions for the discovery of pharmaceutical substances, as well as for genomics, proteomics, and diagnostics. Maintaining R&D sites and sales offices around the world, the company employs roughly 1,000 people. Customers are segmented as follows: biopharmaceuticals (research labs, academic sites, etc.), diagnostics (hospital labs, blood banks, etc.), and forensics (government labs, the police, etc.).

The above market segments are covered by five product areas, one of which is the liquid handling and robotics area. The product investigated by this case study is a product platform in that area. It can be supplemented with a large variety of modules, each performing different analyses. Figure 5.6 gives an overview of the liquid handling platform.



1 Pipeting tubing
2 Diluters
3 Main power switch
4 Top safety panel
5 Wash bottle rack
6 Loading Port (RT-Incubator)
7 Access door lock
8 Microplate Washer
9 Absorbace reader
10 Incubator with optional shaker
11 Robotic Manipulator arm (RoMa)
12 Access door
13 Worktable
14 Positive identification (PosID)
15 Shelf
16 Liquid Handling arm (LiHa)

Figure 5.6 Product overview of liquid handling platform

5.3.1.2 Situation at Beginning of Case Study

The liquid handling platform analyzed in this case study is in the maturity phase of its life cycle. The purpose of the research performed here mainly was to indicate areas of improvement. Over its life cycle, the platform had become so complex and its architecture so intertwined that most employees admitted that only a few co-workers fully knew the platform and could judge whether (and why) certain modules were necessary, how they could be improved, etc.

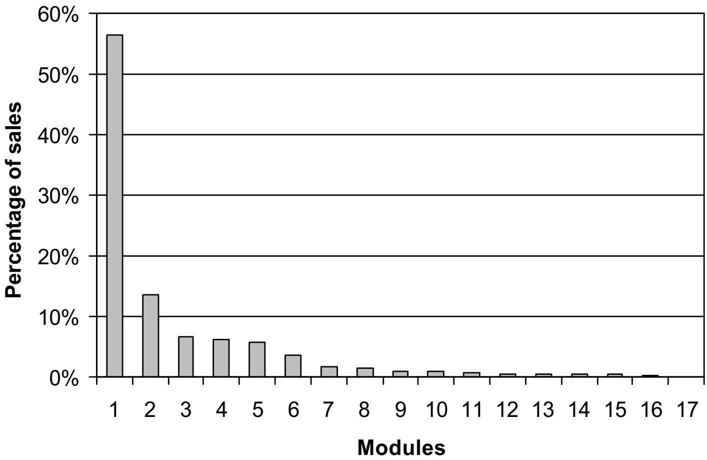


Figure 5.7 Sales values of individual modules contained in the product (period considered: one year)

The modules constituting the platform are available in a startling variety of several hundred for many modules, one module even boasting 1,836 variants. Even though not all module variants make sense to combine, one can imagine the overwhelming number of platform variants and the correspondingly great difficulties in handling complexity. Sales figures of all modules as indicated in Figure 5.7 show the familiar pattern of few actors providing most activity.

To summarize these brief introductory notes, the company's interest in participating in the case study was to apply a tool (the complexity management model) that offers a way to describe complexity in the existing platform and to learn from the implications for the next generation platform.

5.3.2 Application of Complexity Management Model

5.3.2.1 Strategy and Product Life Cycle Assessment

Many of the products sold are unique and configured to the wishes of one particular customer. This is especially true for the biopharmaceutical segment, where the research aspect is more important and, hence, new types of liquid handling devices to test novel applications are needed. Nevertheless, the company attempts to organize its liquid handling product portfolio around a limited number of fairly standardized modules that can be combined according to customer requirements.[12] The diagnostics segment in particular is a market with a lower price level and thinner margins, requiring lower-cost and standardized solutions.

In the course of time, the original platform concept to keep variety at a level that can be overlooked and handled has become increasingly diluted. The company was well aware of the corresponding increase of complexity costs and therefore decided to launch a new liquid handling platform that would allow for a more standardized and flexible product architecture. In terms of the differentiation and standardization matrix, the company's original, present, and envisaged strategic positioning can be sketched as shown in Figure 5.8.

The company's competitive edge in the biopharmaceutical segment clearly is differentiation by means of superb product quality and customer services. The other two segments require a strategy that also incorporates serious cost considerations, thus entailing a hybrid competitive strategy. This twofold, segment-specific strategy complicates matters somewhat and necessitates a differentiated approach when deriving guidelines for action. The strategic situation for the three segments is also shown schematically in Figure 5.8. In the desired state (which exhibits a higher level of standardization as compared with the present state), the highly customized requirements in the biopharmaceutical segment can be catered to by using a platform as a starting point and tailoring it to the specific needs. Simultaneously, the more cost-sensitive diagnostics and forensics segments can be served by means of the same product platform.

The existing liquid handling platform, as mentioned above, is in the maturity phase of its life cycle. The extended product therefore becomes more important, and cau-

[12] Note that modularizing products in such a way is termed sectional modularity (see Figure 3.9).

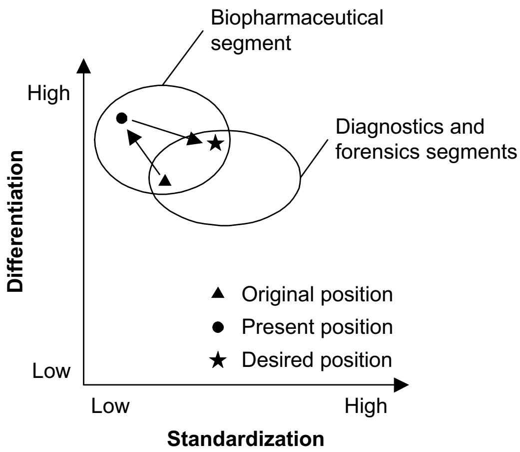


Figure 5.8 Evolving strategic positioning for the liquid handling platform

tiously planning expenditures is at a premium. But while the complexity management model is applied to the existing platform, the model's results are used by the company to support the development of the new platform. As a consequence, the implications of a product relaunch and new product development must also be kept in mind when deriving guidelines for action for optimizing the product architecture in the liquid handling platform. Table 5.7 summarizes the strategy and product life cycle assessment for the liquid handling platform.

Table 5.7 Liquid handling platform: summary of strategy and product life cycle assessment

Criterion	Liquid handling platform case
Competitive strategy	From differentiation (highly customized) towards hybrid strategy; intensify platform aspect
Strategy classification	Shift from tailoring to menu industry, i.e. shift of products and processes from tailored customization to customized standardization; transaction remains fully customized
Product life cycle	Existing platform that the model is applied to: maturity phase; new platform (which the results are also used for): under development

5.3.2.2 Product Complexity Assessment

Just as in the railroad signal case study, the necessary data were acquired by conducting workshops and interviews with the relevant employees. First, the structure of functionality of the liquid handling platform was derived, showing five levels of hierarchy.[13] The importance of every functional element was weighted and a relative percentage assigned to each. Deploying the functional importance to the physical components (called modules in this case) first required an answer to the question of what should be considered a module (and what not). The result of this process was a set of 17 modules that constitute the platform. Every module received a percentage showing its contribution to the platform's functionality, depicted on the vertical axis of Figure 5.9.

For the physical complexity axis, the four complexity drivers (number of parts, number of module variants, number of interfaces, and number of interface variants) were determined for each module, just as described in Subsection 4.3.3. The number of parts and number of variants differed significantly from one module to the other. Some had only two or three variants, while others showed several hundred. Therefore, for the first two complexity drivers the logarithm with base ten was calculated and used in the computation of physical complexity.[14] The number of interfaces and interface variants for each module was found by using the DSM. The four complexity drivers are shown for a few modules in

Table 5.8, providing the inputs for Equation 4.1 and every module's coordinate on the abscissa of the complexity matrix. When combining physical complexity and functionality, the complexity matrix for the liquid handling platform can be depicted as in Figure 5.9.

---

[13] As an explanation, consider Figure 2.18. The structure of functionality there consists of three levels of hierarchy (overall function, sub-functions, and functional elements). The five levels encountered here reflect the product's inherent complexity.

[14] Refer to Subsection 4.3.3 and Appendix D.3 for more details on the subject of calculating the logarithm for complexity drivers instead of using the actual numbers.

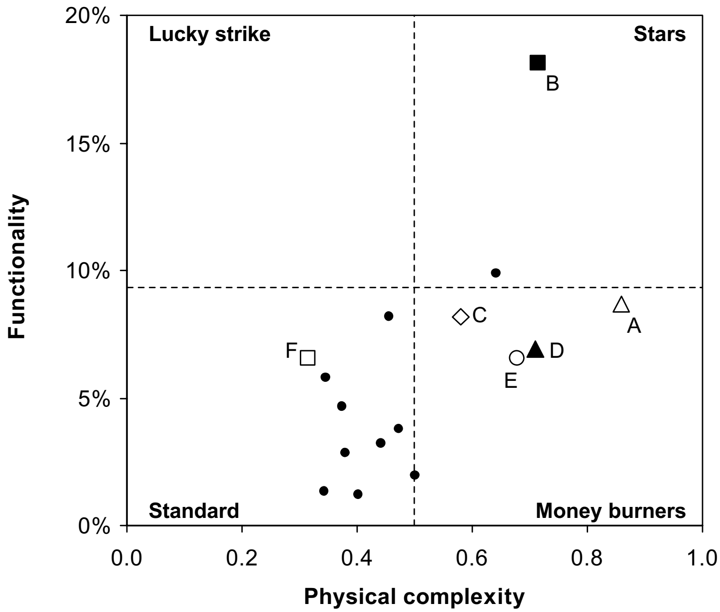


Figure 5.9 Complexity matrix for the liquid handling platform with emphasis on modules A to F (see Table 5.8) out of 16 modules in total

When looking at the complexity matrix in Figure 5.9, ten modules are located in the "standard" quadrant, four in the "money burners" quadrant, one on the border between the two quadrants, and two in the "stars" quadrant. No module even comes close to the "lucky strike" quadrant. The next subsection investigates the options to optimize the product architecture of the liquid handling platform.

5.3.2.3 Deriving Guidelines for Action

In Section 4.4, the options at hand to influence components' location in the complexity matrix and, thus, optimize product architecture were presented in detail. The situation encountered in this case study is depicted in Figure 5.9, and it becomes clear that the modules A, C, D, and E are the source of the greatest concern. Even so, most modules of the liquid handling platform were reconsidered, and slight changes to the product architecture were also able to be recommended to the company in those cases.

The general direction of product architecture optimization chosen here is based on the idea of decoupling those platform variants that generate a considerable level of sales volume from those that are customer-tailored and unique developments. In doing so, the basic platform does not have to be designed to accommodate the widest possible range of variety, which saves development time and effort, and complexity costs in general. Thanks to the platform's high degree of modularity, customized add-ons can be supplemented to the platform for special customer orders via standardized interfaces. If this logic is extended and applied to the modules in the complexity matrix, those module variants that were sold only once or twice to one single customer can be considered customer-specific solutions that are not part of the basic platform and be

Table 5.8 Complexity drivers for the liquid handling platform (excerpt)

Module	Symbol	Number of parts		Number of variants		Number of interfaces	Average number of interface variants
		Actual	$\log_{10}$	Actual	$\log_{10}$		
Module A	△	288	2.46	304	2.48	17	1.82
Module B	■	1,067	3.03	1,836	3.26	8	1.13
Module C	◇	469	2.67	12	1.08	12	1.17
Module D	▲	3,206	3.51	960	2.98	6	1.5
Module E	○	464	2.67	216	2.33	7	2
Module F	□	434	2.64	1	0	3	1.33
⋮		⋮	⋮	⋮	⋮	⋮	⋮
Maximum value		3,206	3.51	1,836	3.26	12	2

designed only upon customer request.[15] Therefore, several low-sales variants are canceled from the basic platform modules in the guidelines presented below. Recommendations for action for module A are presented here because the most pronounced improvement can be achieved with that particular module and the best insight is given into the options at hand.

Module A's large contribution to physical complexity mainly stems from its many variants and interfaces. Hence, these are the two complexity drivers that must be reduced. As for the variants, an analysis of the units sold revealed that 94 percent can be covered by a mere set of ten module variants – instead of the total of 76 variants maintained at present (see left portion of Figure 5.10). The remaining six percent should not be part of the basic platform as the corresponding module variants cause excessive complexity without generating an accordingly high sales volume. This six percent represents the solutions mentioned above that are tailored to specific customer requirements and must be treated (and priced) separately. The recommended ten module variants can be described by the attribute-value matrix in the right portion of Figure 5.10. The first six variants (producing the highest sales volume) represent the combination of the three different lengths (100 mm, 150 mm, 200 mm) and the two surface types (stainless and coated), all without gap. Additionally, the 100 mm / stainless and the 150 mm / stainless variants are offered with one gap, while the 200 mm / stainless variant is offered with one and two gaps, respectively, adding up to a total of ten variants.

Concerning the interfaces of module A, things are slightly trickier. Because module A can be viewed as the "bus" to which many other modules can be connected,[16] it

[15] If extremely customized solutions (i.e. only one or two units of the same new variant for one single customer) are deliberately kept separate from the "normal" business, one major advantage is the ability to better trace the actual costs incurred. This allows an accordingly accurate pricing of the special version.

[16] In this sense, module A can be viewed as the bus in the "bus modularity" type of Figure 3.9, where a variety of modules can be attached to the bus. Note that the bus in the bus modularity type does not allow for variants; it is considered a common, standardized structure. Module A does show variety, which – in a strict sense – sets it apart from bus modularity.

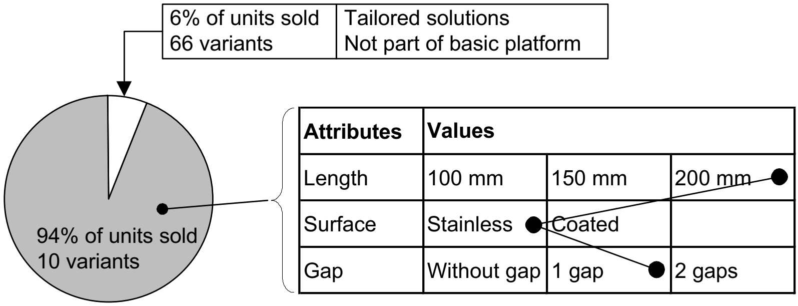


Figure 5.10 Attribute-value matrix and corresponding sales figures for module A[17]

shows a large number of interfaces. One option for reducing interfaces is dividing module A into clearly distinguishable sections, each of which shares interfaces with clearly defined modules. (In the present platform layout, interfaces with other modules intertwine and their location is not pre-defined.) In such a way, not only can interfaces be reduced, but the flexibility of adding (and leaving out) modules is increased.

Besides reducing physical complexity (which is the focus here according to the strategy and product life cycle assessment), an increase in functionality for module A is desirable as well. Adding more functions to the module would also avoid conflicts between two other main modules – module B and D.

5.3.3 Result of Optimization

The effect of the guidelines given above can be visualized by the complexity matrix shown in Figure 5.11. The optimizations as described for module A as well as a list of other architectural changes resulted in several modules having shifted to the left, and matters decidedly cleared up in the “money burners” quadrant. Module A – thanks to

[17] Note that the values of every attribute can be combined with several values of the other attributes. One possible combination is indicated by the line in Figure 5.10. See 2.1.2 and 4.3.3.2 for more information on the attribute-value matrix.

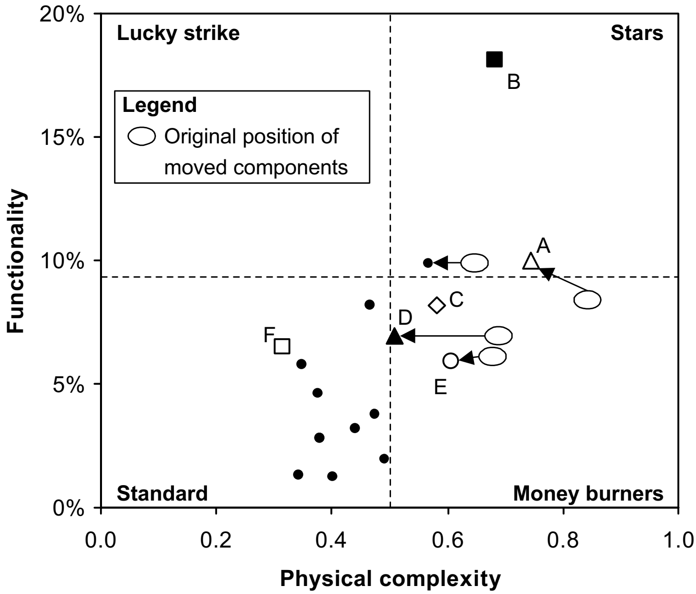


Figure 5.11 Complexity matrix after the optimization process with emphasis on modules A to F out of 16 modules in total

the added functionality – was able to be pushed into the "stars" quadrant, and module D sits on the border between the "standard" and "money burners" quadrant. However, module E and D still remain in their original quadrant. The reason for this is discussed in the next subsection.

5.3.4 Discussion

The previous subsection showed that by applying the complexity management model, valuable recommendations can be given as to how the platform's architecture can be optimized. The physical complexity was reduced for several modules without compromising their functionality. For module A, an increase in functionality could even be achieved.

The complexity management model was considered a valuable source of information by the employees involved in the case study for several reasons. It provides a tool for product designers, systems engineers, product managers, marketing managers, etc. to identify those components and modules that cause excessive complexity. Furthermore, it presents an unusual approach and provides support in making decisions. Interestingly, the location of many modules in the complexity matrix confirmed what the company already seemed to know or at least was suspecting – only the answers to the whys were blurred.[18] Obviously, it is reassuring to have a mere gut feeling backed by a model that is based on quantitative figures taken directly from the product's nuts and bolts.

Even though the balance of functionality and physical complexity can greatly be improved in the liquid handling platform, two modules in particular were not able to be removed from the "money burners" quadrant. Module E comprises a large variety of extensions for module A that are all important not so much because of the actual functions they convey, but because they are believed to contribute a great deal to differentiating the product – because they are supplemented to the functionally important module A. Hence, any change made to module E involves strategically sensitive issues and requires careful management decisions. For this reason, module E was left as is by the complexity management model.[19] After thoroughly considering module C (which, among other functions, is responsible for the transporting of samples), the employees involved agreed that reducing its physical complexity is virtually impossible if the optimization is restricted to the four complexity drivers. A significant enhancement can only be achieved by completely redesigning the sample transporting system, which would require a major R&D project. Therefore, module C was left unchanged.

---

[18] In one instance, a product designer looking at the complexity matrix in Figure 5.9 exclaimed to his colleagues, "I always told you that … [module A] is way too complex compared to the other modules!"

[19] The effects on module E of altering neighboring modules (e.g. changing interfaces) were accounted for. But these effects are minimal (reflected by the short arrow next to module E in Figure 5.11), and module E as such was left as is.

An impression that had already been expressed in the discussion of the railroad signal case study was confirmed by this second case. The model's strength can unfold where changes are incremental and the basic product concept is already firmly in place. Only then does the model provide reliable information and useful guidelines for action. As soon as the basics of the product architecture are touched (e.g. completely redesigning a component, as in the case of module C above), the model has to be applied to the new situation all over again to give meaningful recommendations. Thus, the model's guidelines for action do not foster radical change in product architecture and are less successful in very early stages of product development, where the focus is still broad and concepts vary greatly.

The guidelines for action that are targeted at reducing complexity in this case study are predominantly based on the idea of splitting the liquid handling portfolio into platform variants that cater to a large portion of the sales volume and into the customer-tailored low-sales variants. Keeping these two order types more neatly apart than at present increases transparency (with regard to costs, resources, product variety, etc.) and supports decisions about whether to develop, manufacture, and sell a particular customized variant. However, it must be kept in mind that the highly customized solutions typically often boast higher margins, which makes them potentially more interesting for the manufacturer.[20] Therefore, splitting the portfolio can only be successful when all aspects of both the "normal" portfolio and the highly customized variants are fairly weighted against each other.

A further issue that arose while discussing the results with the people in charge of the platform is concerned with the complexity inside the module as opposed to the complexity exhibited by the module in the complexity matrix. Module A shows a high

[20] The problem of correctly allocating overhead to all variants of a product was introduced in Subsection 2.1.3. Figure 2.13 visualizes the gap of perceived and actual costs of low-sales variants and concludes that the seemingly attractive margins of "exotic" variants do not reflect reality properly. Having said this, product features or product variants that provide a competitive edge through differentiation (e.g. customizing a product exactly to the wishes of one particular customer) can in general produce higher margins – provided that costs are kept within certain limits. A widely used tool to correctly quantify costs for all variants of a product is activity-based costing (ABC). For more information and literature on ABC, see Subsection 2.1.3.

level of physical complexity in Figure 5.9 and Figure 5.11 because of its many interfaces and variants. But by simply considering its physical structure and how it looks, module A does not seem very complex. On the other extreme, module F (see Figure 5.9 and Figure 5.11) is the least complex of all modules on the physical complexity scale. However, the module is mechanically highly complex inside. Obviously, thanks to its few and standardized interfaces (and its high degree of modularity) module F is not regarded as being complex in terms of the complexity matrix. The number of parts is the only complexity driver that takes into account the complexity “inside” of the module, while other aspects are neglected. To alleviate this drawback, the interfaces inside the module could be considered (not only the module’s interfaces with other modules). Such a model extension would, of course, compromise the model’s ease of application.

Even though software and electronics were excluded explicitly in the reference frame (see Section 1.3), the model’s inability to integrate the two proved to be an important drawback in the course of this case study. As the product considered here (and most of the company’s other products) consists of software and electronics to a considerable degree, the complexity picture drawn by the model is incomplete and covers only the product’s mechanical portion. If a tool were available that could consider the entire product and also quantify the complexity contributed by software and electronics, a great improvement would be achieved and much benefit added to the user of the model.

5.4 Process Industry Compressor

5.4.1 Introduction to the Case

5.4.1.1 Company Profile

The third case study was conducted at a multinational company’s division that is grouped around the four business lines of industrial power generation, oil and gas industry solutions, process compression, and service. The division develops and manufactures industrial steam turbines, industrial gas turbines, and compressors. These products are marketed to power generators (e.g. power plants) and power consumers, such as the oil and gas industries, petrochemical and refining industries, pulp, paper, and fertilizer industries, etc. The division employs roughly 10,000 people, with R&D and manufacturing sites in Europe and the U.S. and sales offices around the world.

The product considered here is a centrifugal compressor of the process compression business line used in process industries, such as chemical and petrochemical industries, oil refineries, air separation plants, etc. The compressor spins around a shaft and comprises several compression stages to attain high pressures, i.e. it is a single-shaft, multi-stage compressor. The company offers a wide range of process industry compressors that are all tailored to the specific needs of every single customer. Figure 5.12 exhibits a typical centrifugal compressor of the type considered here.

5.4.1.2 Situation at Beginning of Case Study

While performing the research for this case study, I was taking part in a consulting project in the company's division. The project aimed at identifying the sources of complexity in the process industry compressor, eventually leading the way to a more modularized product. Even though a basic product portfolio with a limited range of compressor types has been defined and set into place by the company, compressors are tailored to the specific needs of every customer, which leads to a very broad portfolio. Furthermore, the necessarily highly complex product architecture of compressors obscures the potential to standardize, to form modules, and to simplify in general.

That is the point where the complexity management model came in. The main purpose of applying the model to the compressor was to investigate an approach suggested by the consultants that involved a split of the component assumed to be the most complex within the compressor. Splitting the component into well-defined subassemblies (termed "design chunks" in the project) was motivated by two issues: (1) smaller and simpler components can be handled better (provided, for example, that a clearer partitioning of functionality can be achieved); (2) an increased degree of modu-

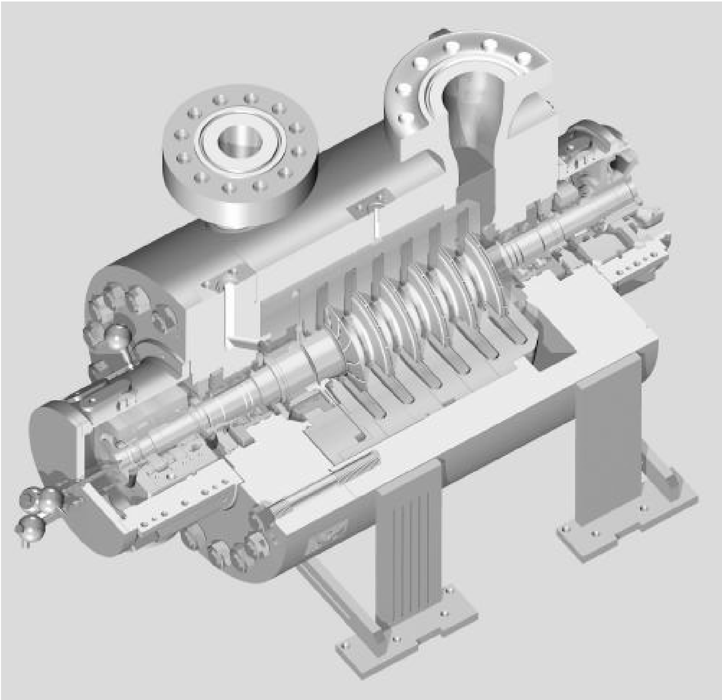


Figure 5.12 Multi-stage centrifugal compressor

larity means fewer and standardized interfaces. It was decided to monitor the component split (among other measures) by means of the complexity matrix in order to assess its effect from a complexity management point of view. It was agreed that the split can be viewed as successful only if complexity can be reduced within the compressor. In that case, all the entailing advantages become effective throughout the value chain.

To give an impression of the inherently complex situation in the process industry compressor, Figure 5.13 depicts, on the one hand, the compressor modules and the corresponding functional elements they fulfill. On the other, the attributes (each of which has several values) are shown which the modules depend on. The intertwined and very dense connections reflect the product's complexity.

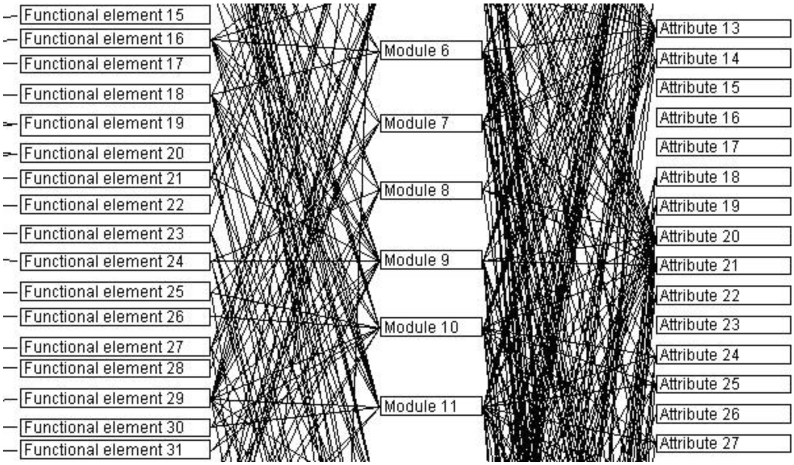


Figure 5.13 Compressor modules and their corresponding functional elements and attributes[21]

5.4.2 Application of Complexity Management Model

5.4.2.1 Strategy and Product Life Cycle Assessment

As was briefly mentioned in the company profile above, the company bases its competitive edge on the ability to tailor its compressors to the very needs of its customers. Some basic versions of the compressor do exist, but all units sold are optimized (and, therefore, customized) with respect to energy efficiency, fluid dynamics, and the customer's specific situation (power needed, ambient conditions, etc.). In terms of Porter's competitive strategies, the company is a differentiator serving the broad market.

The objective of the consulting project was to increase the compressor's degree of modularity and find ways to foster standardization in the product. According to the complexity classification of industries introduced in 4.2.1.4, this attempt is paramount

[21] Figure 5.13 is an excerpt from the compressor's product architecture and its attributes. The graphic was generated by the commercial software METUS.

with a shift from pure customization (i.e. a “thin industry” company) towards a “tailoring industry” company. Under such a regime, the latent potential to standardize products and processes is exploited, while the transaction (e.g. relationship between sales force and customers) remains fully customized. The company was well aware of the fact that the scope for standardization is very limited for such complex products that deliberately draw their competitiveness from exactly matching customer requirements. Therefore, even in the extended Porter framework (see 4.2.1.3), a hybrid competitive strategy cannot be the aim here.

The process industry compressor is in the maturity phase of its life cycle, with product variants proliferating and service offerings intensifying to keep profits rising. As discussed in Subsection 4.2.2, the cost aspect becomes more and more pressing in the maturity phase. Thus, several initiatives to cut complexity costs were started, among them the consulting project covered here.

Table 5.9 summarizes the strategy and product life cycle assessment for the process industry compressor.

5.4.2.2 Product Complexity Assessment

The investigation of the compressor’s complexity was based on 22 modules constituting the product. The modules were defined in the course of the consulting project and aimed at dividing the product into distinct regions. These regions (or modules) should

Table 5.9 Process industry compressor: summary of strategy and product life cycle assessment

Criterion	Process industry compressor case
Competitive strategy	Differentiation (fully customized); exploit limited potential to standardize
Strategy classification	Shift from thin industry towards tailoring industry, i.e. shift of products and processes from customization towards tailored customization; transaction remains fully customized
Product life cycle	Maturity phase

be functionally as independent as possible.[22] Some of the modules were physically separable compressor components, while others merely formed a part of a larger physically separable unit.

The first task in the product complexity assessment step is deriving every module's functionality coordinate in the complexity matrix. Due to the large number of functional elements,[23] every functional element is assumed to be of the same importance, and the number of functional elements fulfilled by every module is simply counted.[24] This eventually leads to a functionality percentage assigned to each module, and thus the coordinate on the vertical axis of the complexity matrix, shown in the complexity matrix in Figure 5.14.

Calculating the physical complexity axis follows the procedure outlined in Subsection 4.3.3. The fourth complexity driver – number of interface variants – cannot be determined in this case study due to the difficulties in deciding under what circumstances an interface varies and when it does not.[25] Furthermore, the computation of the number of variants for every compressor module (the second complexity driver) deviates from the "normal" procedure presented in Subsection 4.3.3. Many compressor components depend on parameters that can be changed according to customer requirements. Therefore, the exact number of variants is not confined to a limited and pre-defined set and is thus not of great relevance to the company. That is why information on how many variants of every module exist was not available. As a result, the attributes and their values are taken as a good approximation for modeling a module's number of variants. As an example, assume a module's variety to be determined by three attributes, each

---

[22] The modules were called "functional modules" in the consulting project to emphasize the goal of defining modules that are functionally as independent as possible.

[23] 258 functional elements were identified for the compressor; the structure of functionality exhibits four hierarchical levels.

[24] Refer to Subsection 4.3.2 for further explanations on calculating the functionality coordinates of components by counting the functional elements.

[25] In terms of Equation 4.1, not considering the fourth complexity driver essentially leads to setting the fourth coefficient at zero, $\delta = 0$.

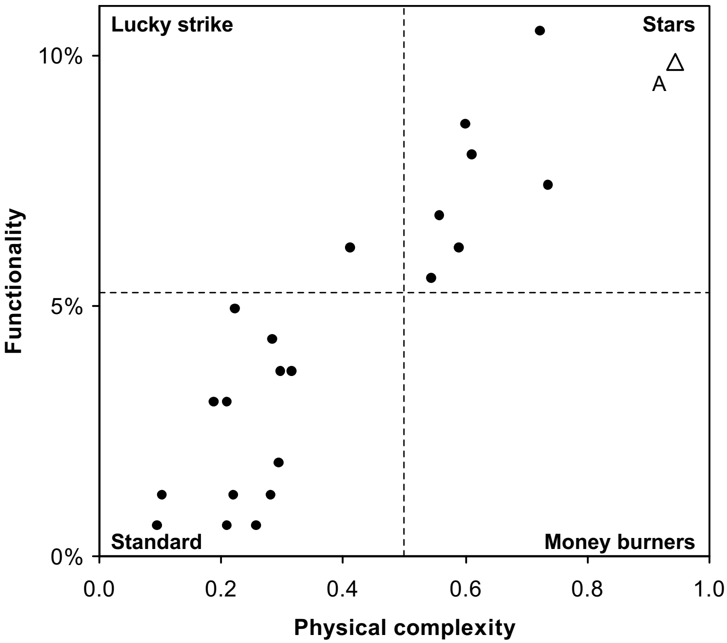


Figure 5.14 Complexity matrix for the process industry compressor with emphasis on module A (out of 22 modules in total)

having two, three, and five values, respectively.[26] The module's maximum variety therefore is 30 (2 × 3 × 5). This arithmetically calculated number does not necessarily have to be equal to the actual number of variants offered, but it is still a good indicator of the module's variety. Because multiplying attribute values soon reaches very large numbers, the logarithm with base ten was calculated instead of simply taking the multiplied numbers. In this way, very large numbers can be prevented from exerting a disproportionately heavy influence on the complexity matrix.[27] The two remaining com-

[26] For a more detailed coverage of attributes and values (and the attribute-value matrix), see 2.1.2 and 4.3.3.2.

[27] Refer to Subsection 4.3.3 and Appendix D.3 for more information on the subject of calculating the logarithm for complexity drivers instead of using the actual numbers.

plexity drivers – number of parts and number of interfaces – are calculated in the normal way.

When combining the functionality and physical complexity axes, the complexity matrix can be drawn as shown in Figure 5.14. It becomes apparent immediately that all modules are neatly distributed along the matrix diagonal, with no module in the "money burners" quadrant, one "lucky strike", and all the others either in the "stars" quadrant (8 modules) or the "standard" quadrant (13 modules). The next subsection presents how the complexity matrix is used to support the compressor's modularization.

5.4.2.3 Deriving Guidelines for Action

As was mentioned above, the purpose of applying the complexity management model in this case study was mainly to investigate the effect of splitting the module assumed most complex within the compressor. First of all, it is interesting to see that the respective module (marked "A" in Figure 5.14) is pointed out as the physically most complex one also by the complexity matrix. The quantitative calculation thus matches with the experience accumulated within the company – a significant achievement of the model as such. Because no modules are in the "money burners" quadrant demanding profound attention, the guidelines for action presented here are exclusively focused on splitting module A into smaller and less complex sub-units that can be handled more easily.

In a first step, module A is split into two smaller modules that, on the one hand, are functionally as independent as possible and, on the other, can significantly reduce physical complexity. Due to the inherently complex situation, identifying the most promising areas to reduce complexity (which interfaces, what variants, etc.) is not as straightforward as in the railroad signal case study of Section 5.2. In some cases, the suggested changes strongly relied on the experience of certain employees. The preliminary result is shown in Figure 5.15, where the newly formed modules B and C are indicated. Because module A has been split, both of the modules B and C have a lower functionality. It can also be seen that B and C are physically less complex than A, a

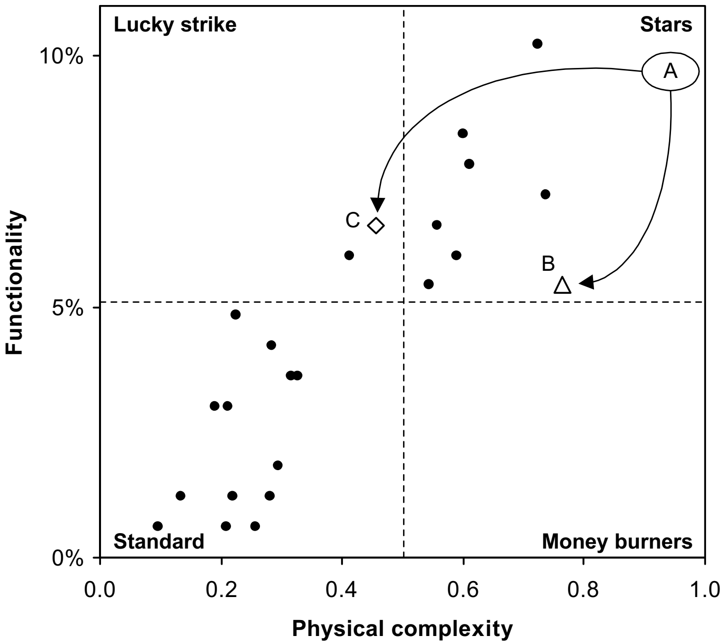


Figure 5.15 Splitting module A into two new modules with emphasis on modules B and C (out of 23 modules in total)

major objective of splitting module A. While C is located in the "lucky strike" quadrant, B sits very close to the "money burners" quadrant, a situation that should be investigated in more detail and be improved if possible. In the strategy and product life cycle assessment step above it was shown that even though the competitive strategy clearly is differentiation, the cost aspect must receive ample attention because the product is in its maturity phase. Furthermore, the consulting project's focus primarily lies on reducing complexity costs. Thus, ways must be found to shift module B further to the left.

In an attempt to improve the situation depicted in Figure 5.15, both modules B and C are further split into sub-units, termed "design chunks" in the consulting project. In the present compressor configuration, the design chunks are not physically separable units. They are part of a larger unit (modules B and C, respectively), and the boundaries between the chunks are merely defined mentally – hence the name design chunks. Once a full-scale R&D project is launched after the consulting project, the compressor's architecture can be optimized by introducing new (and physically separable) modules that are defined by the present design chunks. For the time being, the benefit of identifying design chunks stems from better understanding of where complexity is caused and how it can be reduced. While forming the chunks, just as in splitting module A into B and C, emphasis is placed on defining regions that are functionally as independent as possible and reduce physical complexity.

5.4.3 Result of Optimization

The result of further splitting modules B and C into design chunks is shown in Figure 5.16. The design chunks B_1, B_2, and B_3 are based on module B, while C_1, C_2, and C_3 stem from module C. All design chunks are located well within the "standard" quadrant, except B_1 sits right on the border between the "standard" and "money burners" quadrants. In summary, the formerly fairly complex module A is transformed via the two modules B and C into six much less complex design chunks, one of which (B_1) still exhibits a slight mismatch between the functionality it offers and the physical complexity it causes. The next subsection discusses the findings of this case study.

5.4.4 Discussion

The complexity management model's application to the process industry compressor exhibits, at a first glance, two interesting issues:

- All modules are more or less grouped along the complexity matrix diagonal. On the one hand, this shows that the compressor's product architecture is well balanced with regard to customer aspects (functionality) and internal complexity (physical complexity). On the other, it supports the model's validity because the modules are not scattered randomly throughout the complexity matrix. It underscores the fact that there is a relationship between functionality and physical complexity as defined by the model.

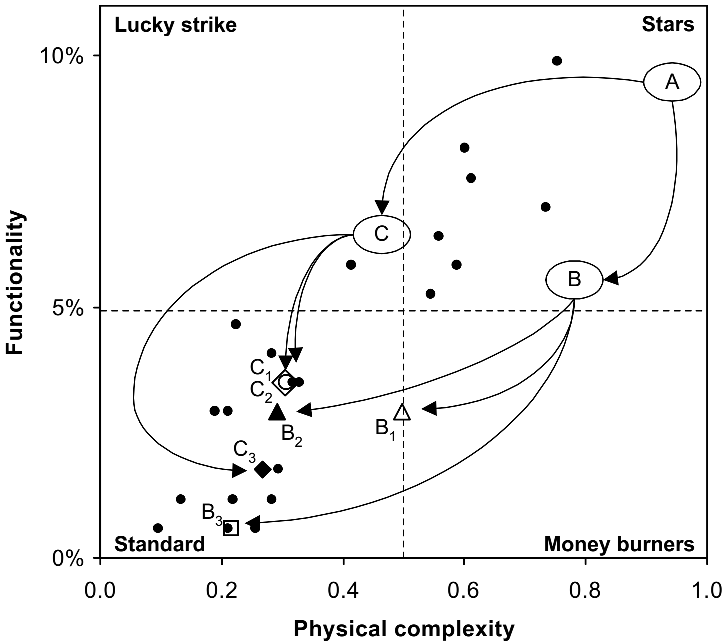


Figure 5.16 Complexity matrix after forming design chunks with emphasis on design chunks B_1 to B_3 and C_1 to C_3

- Module A was detected by the model as the compressor's physically most complex (and functionally second most complex) module. This finding fits perfectly with the company's experience and its knowledge about the product – even though "complexity" as such has never been quantified either for the compressor or any modules. The major impact here is the model's ability to transform latent knowledge into a quantitative framework that clearly points out the sources of complexity.

When further investigating the model's achievements, it was able to track the modularization of a highly complex component with respect to functionality and physical complexity. The path from the original module via the two sub-modules to the six design chunks was clearly delineated and the implications regarding complexity pointed out. Certain remaining problems with design chunk B_1 were identified (too much physical complexity compared to its contribution to functionality) and must be addressed in subsequent design overhauls and modularization attempts. All in all, splitting module A into six design chunks is considered beneficial by the consulting team and the company alike to tackle the inherent complexity problems in the compressor. The design chunks are smaller and can be handled more easily, which translates to a less intertwined architecture, more clarity with respect to interfaces, the opportunity of accomplishing faster changes, and a better product overview in general.

Once again – as with the previous two case studies – the guidelines given by the model proved to be of somewhat limited scope. While the model's strength lies in its ability to draw a clear picture of the complexity situation in a product, the suggestions for improvement are all oriented towards the four complexity drivers and functionality. This does not come as a surprise, of course, as these are the complexity matrix' ingredients, but in real-life projects questions like "what other optimization possibilities do we have to resolve this situation?" inevitably arise and can rarely be answered by the model.

In the original setting, 22 modules were considered in the complexity matrix (see Figure 5.14), while after the optimization (including the design chunks), 27 modules and design chunks are investigated by the model. Compared to the two previous case studies (railroad signal: 16 components; liquid handling platform: 17 components), applying the model's product complexity assessment step becomes slightly cumbersome simply due to the increased number of components that must be handled. The user can become overwhelmed easily, and deriving meaningful guidelines for action is often obscured by the sheer number of parts, component variants, interfaces between components, etc. What the model states as such is not compromised, though. It must be assumed that somewhere in the region above 30 components, applying the model does not make sense. To resolve the problem, the investigation's level of detail must either be reduced (i.e. decrease the number of components considered), or some numeric tool should be developed specifically to the needs of the complexity management model. Such a tool can potentially support the model's user by taking care of the calculations and assisting in investigating alternative optimization scenarios.

5.5 Railroad Switch Lock

5.5.1 Introduction to the Case

5.5.1.1 Company Profile

The fourth case study was conducted at the same company as the railroad signal case study – a multinational company's regional division employing 600 people and focusing on the rail automation business. The railroad switch lock considered in this case study is developed and manufactured at the regional division's site and marketed to domestic (approx. 10 percent of sales) as well as international (approx. 90 percent) railroad companies. While the market share in the regional division's home country is at a high level of roughly 90 percent (though increasingly challenged by competitors), the product competes on the international market with many other switch lock systems. Figure 5.17 exhibits an overview of the railroad switch lock.

5.5.1.2 Situation at Beginning of Case Study

Even though customers are highly satisfied with the product due to its low life cycle costs and high level of reliability, the company is pursuing a twofold path regarding its railroad switch lock:

- *Cost reduction.* Customers demand an ever more attractive product pricing, while metal prices on the world market are soaring (the product mainly consists of different alloys). An additional challenge stems from the increasing complexity costs due to a proliferating product portfolio. This situation exerts considerable pressure on the company to find ways to reduce costs in the existing product quickly.
- *New product development.* As virtually every customer poses slightly different requirements for the switch lock (e.g. rail gauge, rail profile, switch type, climate, in-

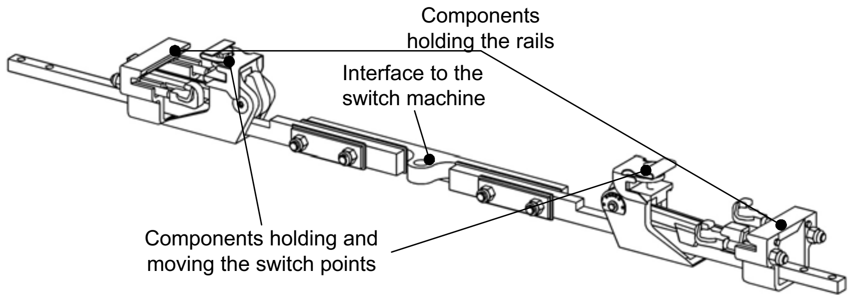


Figure 5.17 Overview of the railroad switch lock

spection times, etc.), the company is forced to offer a large number of product variants. In order to better cope with demand complexity, developing a new product is being scheduled. The vision is to introduce a new product architecture that allows for covering the variety demanded with a minimum of costs.

The above two threads of action – the first a short-term cost reduction program for the existing product, the second a medium and long-term endeavor to develop a new product – form the basis of applying the complexity management model. The benefit for the company to investigate the results provided by the model is to gain insight into the current product's complexity situation and to receive information on where complexity and costs can potentially be reduced. The planned R&D project will also draw on the case study's results because the new product will serve a very similar set of functions required by customers – even though with a different product concept.

An investigation of the railroad switch lock product line revealed the familiar situation of very few product variants generating a large portion of the entire product line's sales, as depicted in Figure 5.18. 11 percent of the variants are responsible for 80 percent of sales in the period considered (six years). 74 percent of all product variants are maintained in the product portfolio by the company, even though they only contribute 5 percent to total sales. The picture is even more accentuated if one focuses on the variants that are designed specifically for one customer. The respective sales distribution is exhibited in Figure 5.19. The chart shows all variants of the product line sold to that particular customer, listed in the order of their percentage of total sales with the customer. It can be seen immediately that only a handful of variants generate sales worth mentioning, while the others are kept "on hold" for special and infrequent applications. Even though these low-sales variants generate only a small fraction of profits (if any at all), the company's credibility as a viable supplier of railroad switch locks strongly depends on its ability and willingness to produce these seldom needed variants.

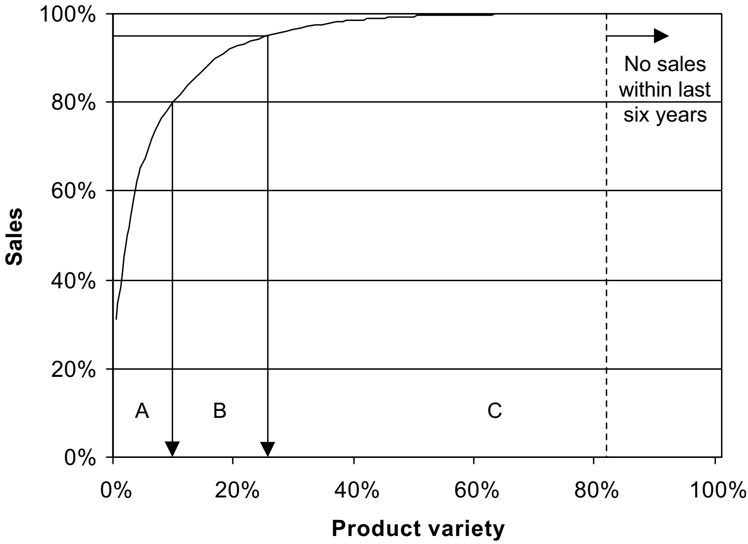


Figure 5.18 ABC analysis of the railroad switch lock sales[28]

---

[28] The A variants are defined as those variants that account for 80 percent of sales (11 percent of all product variants); the B variants as the next 15 percent of sales (from 80 to 95 percent); the C variants the rest.

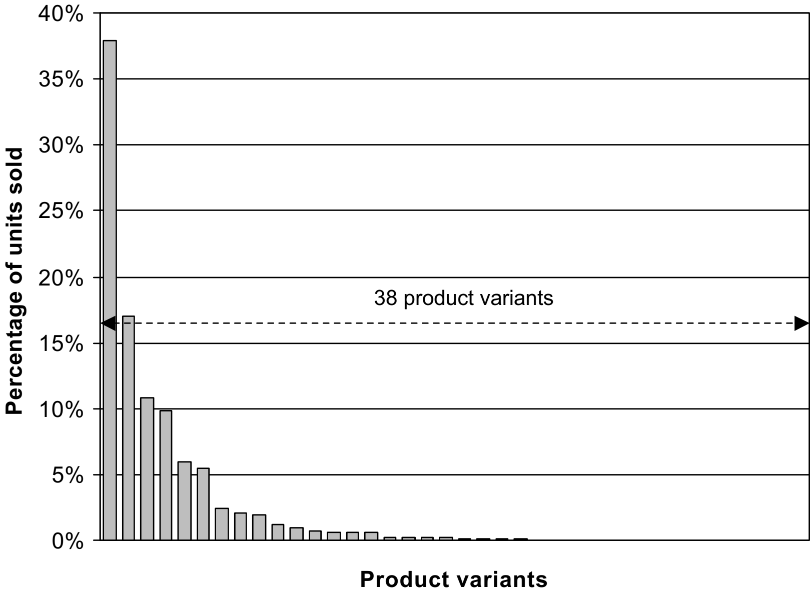


Figure 5.19 Sales distribution of product variants designed for one customer (as percentages of total sales with that customer over a period of six years)

5.5.2 Application of Complexity Management Model

5.5.2.1 Strategy and Product Life Cycle Assessment

Due to the strict requirements of railroad customers, the company is forced to customize its switch lock to those very needs. This translates to a product designed to the conditions of every single customer (e.g. rail gauge, rail profile, regulations, etc.) and – in extreme cases – of individual turnouts. In Porter's framework, the company's competitive advantage stems from differentiation, i.e. its ability to understand exactly what customers need and meet those needs with a corresponding product offering.

As was outlined above, the company's goal is to tap sources to reduce its switch lock's costs and – in the long run – to develop a new product generation. The former objective is in part expected to be achieved by rethinking the product architecture and increasing the level of standardization where feasible. In the latter, the aim is also to find ways of offering the required product variety while incurring the least costs possible. The new product development offers advantages because the product architecture can be designed from scratch. In both cases, however, the company plans to shift from pure customization to a certain level of standardization (in terms of Figure 4.4, a slight movement to the right). Here, developing a new product clearly aims at exploiting the implications of Figure 2.19. The idea is to optimize the new product architecture by enhancing the trade-off between commonality and distinctiveness, i.e. allow for a high level of distinctiveness and commonality at the same time. While a truly hybrid competitive strategy for the switch lock is viewed as undesirable by the company, it does strive to convert from a thin industry to a tailoring industry (refer to Figure 4.7 and Subsection 4.2.1.4), i.e. further standardizing its processes and products, but leaving the transaction (e.g. relationship between sales people and customer) fully customized.

The switch lock considered here is in the maturity phase of its life cycle. The attempt to search for further standardization potential does not take the form of a full-scale product relaunch, but should be regarded as a measure to extend the product's life cycle and maintain its attractiveness for customers. It must be kept in mind, though, that the complexity management model's application and the guidelines for action presented below are also used as background information for the new product development project. Table 5.10 summarizes the strategy and product life cycle assessment for the railroad switch lock.

Table 5.10 Railroad switch lock: summary of strategy and product life cycle assessment

Criterion	Railroad switch lock
Competitive strategy	Differentiation (highly customized); find ways to further standardize
Strategy classification	Shift from thin industry to tailoring industry, i.e. shift of products and processes from customization towards tailored customization; transaction remains fully customized
Product life cycle	Existing product that the model is applied to: maturity phase; new product (which the results are also used for): development planned

5.5.2.2 Product Complexity Assessment

The railroad switch lock consists of 26 components that were all integrated into the product complexity assessment. They range from simple parts like screws and bolts all the way to cast metal components that are further processed on the company's site and must fulfill strict requirements defined in part by railroad regulations.

Calculating the functionality axis followed the procedure outlined in Subsection 4.3.2, i.e. the structure of functionality was derived in a workshop and interviews, and the functions were weighted and deployed to the components by employees of three different functional areas (design, sales, and product management). This resulted in a percentage assigned to every component reflecting its contribution to the overall product's functionality, shown in the vertical axis of the complexity matrix in Figure 5.20.

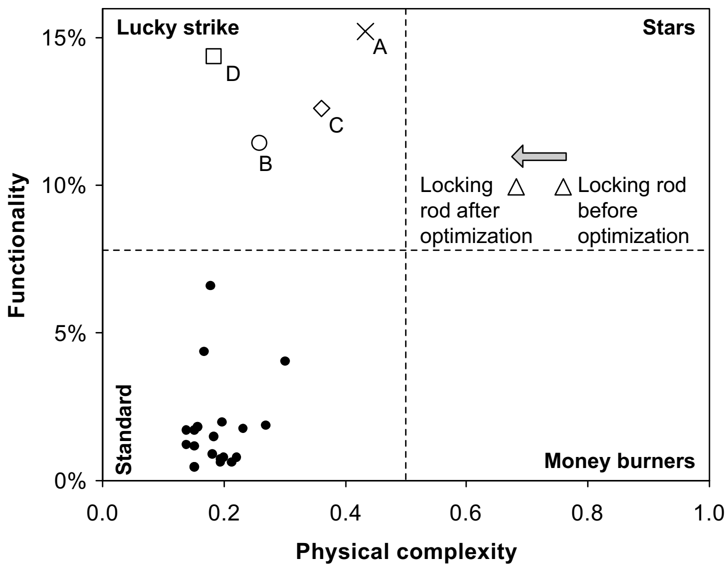


Figure 5.20 Complexity matrix for the railroad switch lock with emphasis on components A to D and the locking rod (see Table 5.11) out of 26 components in total

Table 5.11 Functionality and physical complexity coordinates (excerpt)

Component	Symbol	Functionality	Physical complexity
Locking rod	△	10.0%	0.76
A	×	15.2%	0.43
B	○	11.4%	0.26
C	◇	12.6%	0.36
D	□	14.4%	0.18

The second step in assessing product complexity, computing physical complexity, also exactly followed the procedure described in Subsection 4.3.3. All four complexity drivers (number of parts, number of variants, number of interfaces, and number of interface variants) could be readily calculated. When combining the two axes, the complexity matrix can be drawn as in Figure 5.20. Table 5.11 shows the functionality and physical complexity coordinates of those components that are labeled in Figure 5.20.

5.5.2.3 Deriving Guidelines for Action

It can be seen immediately from Figure 5.20 that no components of the railroad switch lock are located in the "money burners" quadrant and only one in the "stars" quadrant. All other components are in the matrix' left half – either in the "lucky strike" or the "standard" quadrant.[29] The results with respect to the functionality axis are not surprising and closely reflect the company's experience. The switch lock can be divided into two types of components: those conveying the product's functionality and holding a prominent and visible position; and those fulfilling rather peripheral functions (such as screws, nuts, and bolts). The complexity matrix in Figure 5.20 tells exactly the same story: a few components at the top of the matrix and the majority of components clustered in the lower left corner.

[29] The components in the "lucky strike" quadrant (components A, B, C, and D in Figure 5.20) are labeled because they will be referred to in Subsection 5.5.4.

It is much more astonishing, though, that no components – except the locking rod – are classified as being physically very complex compared to the other components. According to the complexity management model, there is no imminent need to further optimize the product architecture as it already represents a highly desirable state. No component causes excessive physical complexity without contributing to the product's functionality.

Nevertheless, as the locking rod is located somewhat close to the "money burners" quadrant, it was decided to investigate the potential for pushing the locking rod to the left. It was discovered that several variants of the rod are manufactured exclusively for one customer, but the same functionality can be provided (and the customer's requirements satisfied) if the rods are machined in the same way as for another customer. In doing so, the specially designed product variants can be abandoned, and the other variants receive an increase in sales. Talks with the respective customer were promising, indicating that they would accept the changeover.

5.5.3 Result of Optimization

The effect of the above guidelines for shifting the locking rod to the left is indicated in Figure 5.20 by the grey arrow. Such a move reduces the number of variants of the rod by 7.7 percent, and the average number of interface variants decreases by 12.5 percent. As a result, the rod's physical complexity coordinate is reduced by 10.3 percent compared to its original value.

5.5.4 Discussion

The fourth case study produced a highly unexpected result. While much room for optimization was always found in the previous case studies, the railroad switch lock obviously already boasts a product architecture that translates customer requirements into the physical product without causing excessive internal complexity. Not much was left to be done, therefore, at least from the point of view of the complexity management model. This definitely is good news for the company.

In order to tap a second source of information, the product was looked at from a slightly different angle. The target costing process according to Tanaka (1989)[30] was applied to the railroad switch lock, the result of which is shown in Figure 5.21. The value control chart shown there compares every component's contribution to functionality with its contribution to the product's overall manufacturing costs. Note that the vertical axis of the value control chart in Figure 5.21 is the same as in the complexity matrix (Figure 5.20). While the locking rod now is located quite far to the left and well within the optimal value zone, components A, B, and C find themselves in the right portion of the value control chart, in the "too expensive" zone. Component D contributes a great deal to functionality but does not cost much.[31] It is interesting to see the parallels and differences in results between the complexity matrix and the value control chart. The two visualization tools essentially agree on the location of component D and the commodity components in the lower left corner. However, while the locking rod is physically complex according to the complexity matrix, its manufacturing costs are close to average. Components A, B, and C are the opposite. In the complexity matrix, they are "lucky strikes." In the value control chart, they are considered too expensive.

The differences between target costing and complexity management model obviously stem from the different definitions of their axes. While the value control chart states manufacturing costs, the complexity matrix measures physical complexity, which is based on the four complexity drivers (number of parts, number of variants, number of interfaces, and number of interface variants). My conclusion – which can only be a preliminary one – from comparing the complexity matrix in Figure 5.20 and the value control chart in Figure 5.21 is as follows.

---

[30] For more information on target costing, refer to Subsection 3.3.2.

[31] According to Tanaka (1989, p. 67), for components outside the optimal value zone (to the left of the zone in Figure 5.21), "cost increases may be necessary to ensure that the product performs its functions satisfactorily." Cost increases do not always make sense, but the potential should be investigated nevertheless. Therefore, the upper left portion of the value control chart in Figure 5.21 is labeled "check whether under-engineered."

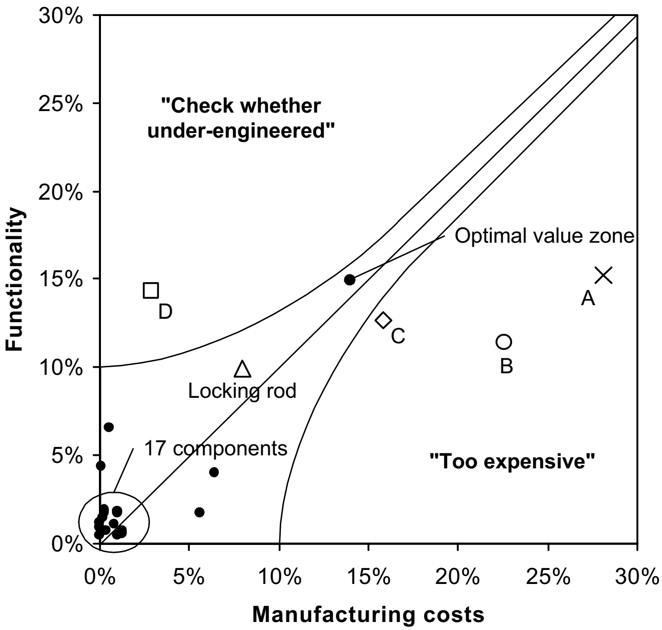


Figure 5.21 Value control chart for the railroad switch lock with emphasis on components A to D and the locking rod (out of 26 components in total)

- Functionally simple (commodity) components causing a low degree of physical complexity usually also boast low manufacturing costs.
- Functionally more complex components that incur high manufacturing costs do not necessarily cause a high degree of physical complexity. The sum of all the complexity costs they cause is strongly determined by manufacturing costs. Other types of complexity costs (R&D, logistics, administration, overhead in general, etc.) are less important.[32]

[32] See Figure 2.10 for a company's functional areas where complexity costs are incurred.

Functionally more complex components that cause a high degree of physical complexity do not necessarily incur high manufacturing costs. Manufacturing costs only contribute a small fraction of the sum of all complexity costs. Other types of complexity costs are more important.

The above three points could just as well form three new hypotheses for future research. One necessary ingredient for answering them is the integration of some cost estimation tool into the complexity management model. If complexity costs could be estimated and linked to physical complexity, the fraction of manufacturing costs could be easily determined. Furthermore, the causes of complexity costs could be traced back directly to the physical product, and guidelines could be derived highlighting the "complexity spots" within product architecture. The advantage of such an extended model mainly stems from its ability to provide complexity information that is cost-based. At the end of the day, the amount of costs saved by the model is of interest.

6 Conclusion

6.1 Reflecting on the Research Achievements

6.1.1 Answering the Research Question

My research question as stated in Section 1.2 is whether a product's competitiveness can be increased by designing the product architecture according to functionality and physical complexity. The complexity management model combines these two dimensions in the complexity matrix and was applied to several products in the machinery and process equipment industries. The research results suggest that the product's balance with respect to the two dimensions is enhanced, i.e. the same or an increased level of functionality is provided while causing less physical complexity. This means that what the product does from a customer perspective (its benefit for users) is accomplished by generating less complexity within the product. As was shown in Chapter 2, complexity within a product's architecture has an effect on virtually all functional areas of a company and causes complexity costs. An improved proportion of functionality and physical complexity therefore entails cost savings for the same customer benefit. Generally speaking, this translates to an enhanced competitiveness of the product because customer value – customer benefit minus total customer costs – can be increased. While the research indicates a basically positive answer to the research question for the cases considered, the limitations to the answer are presented in Section 6.2.

6.1.2 Concluding Model Assessment

This work presented a new approach to enhance a product's architecture with regard to its complexity by means of quantitatively assessing functionality (which models external complexity) and physical complexity (which models internal complexity). This quantitative procedure is supplemented with qualitative aspects concerning company and product strategy, and the product's life cycle. The model is able to provide guidelines with respect to forming standardized chunks and modules within the product, reducing component variety, redesigning interfaces between components, and merging certain components. In doing so, it manages to alleviate the mismatch of functionality and physical complexity within products and balances the two dimensions.

The action research cases presented in Chapter 5 showed that the complexity management model can be applied to industrial products with a reasonably low effort. It requires workshops and interviews with employees of different functional areas, resulting on the one hand in a broad understanding of the product's strategic setting and, on the other, the complexity matrix, which identifies the contribution of the product's components to functionality and physical complexity.

The benefit provided by the model stems from its ability to combine strategic and market issues with a quantification of product complexity, and to condense all relevant complexity information into a handy graphical representation – the complexity matrix. Furthermore, the guidelines for action given to product designers, systems engineers, product managers, marketing managers, etc. lend valuable support when making decisions concerning complexity reduction in product architectures. The model's usefulness to industry is heightened by the fact that the recommendations for product architecture optimization are very concrete and "hands-on" because they are derived based on a nuts-and-bolts analysis of the product's complexity. Nevertheless, the product's broader strategic frame remains on the model's radar and ensures well-balanced recommendations. In the case studies, the employees involved felt that the model supplied them with a viable complexity management tool to assess a product's complexity positioning and provide ideas as to how their product's specific situation can be improved.

As was promised in Section 4.1, the complexity management model is assessed here with respect to the five criteria introduced in Section 3.1, where the framework for evaluating the existing concepts was established. Table 6.1 shows to what extent the model fulfills the criteria and where additional effort is needed.

Table 6.1 Model assessment with respect to the five criteria of Section 3.1

Criteria	Assessment
Company and product strategy	The model's strategy and product life cycle assessment step is specifically designed to incorporate strategic issues. A strict procedure is not provided, though, and the model's user is required to have a certain level of strategic know-how.
Market aspects	Customer requirements are fed into the model via the structure of functionality (and the complexity matrix' functionality axis) in the product complexity assessment step. The quality of depicting customer needs strongly depends on how well they are reflected by the structure of functionality.[1]
Product architecture	Optimizing product architecture lies at the heart of the model presented here. Shifting and adding functions, limiting the number of piece parts used, reducing component variety, splitting components into modules, rethinking interfaces, etc. all directly affect product architecture.
Quantification of complexity	By means of the complexity matrix, complexity of products is described quantitatively with respect to their functionality (which models external complexity) and physical complexity (which models internal complexity). It must be said that the framework is relative and does not provide an absolute measure of complexity.
Applicability in practice	The case studies showed that the model can be applied to problems in industry with reasonably low effort and within a limited timeframe. It was also seen that the model's strength unfolds when incremental (and not radical) product changes are needed. The ease of application is given for less than approx. 30 components.

6.2 Limitations

As with probably any model in management science, there are certain limitations to the theory. Especially in the course of conducting the case studies, I developed a list containing shortcomings, drawbacks, and general observations regarding particular situations where caution must be used when applying the model. At the end of each case study, some limitations concerning the model were discussed briefly (see Subsec-

[1] Results provided by the model are better, of course, if customers are involved in deriving the structure of functionality and the functions' weightings. However, this would add considerable additional effort and is often regarded by companies as not worth the hassle. They trust in their sales people's know-how about customer needs and wants. Also, customers sometimes do not see the benefit of participating in such an exercise. Note that the same limitation applies also for such well-known and long-standing tools as target costing and QFD.

tions 5.2.4, 5.3.4, 5.4.4, and 5.5.4). The aspects presented there are summarized in the following and supplemented with some additional issues:

- While the model's strength is to point out the "bad guys" of a product (or the "hot spots" of complexity in general), the guidelines for action prove to be rather incremental and of limited scope. They are valuable in situations where good agreement on the basic product concept has been achieved and "fine-tuning" is the quest of the day, such as improving an existing prototype. The model does not provide appropriate advice when fundamental, radical changes are needed (e.g. in the early product development stage) – the model cannot replace innovation.

- Related to the previous point, the model was considered by some users as somewhat static and its advice confined to changing the four complexity drivers and functionality. These users would welcome a broader and less strict definition of complexity.

- Depending on the constellation in the product architecture, the guidelines for action sometimes lead towards a less modular (i.e. more integral) product architecture (when merging components). If the original goal was to modularize the product, the model's advice is not of much value, of course. It must be kept in mind, though, that modular product architectures are not "good" per se and often cause higher development and direct per unit costs.[2]

- All complexity drivers except "number of parts" gauge a component's complexity with the outside world, i.e. the variety it shows and the interfaces it shares with neighboring components. If a component is highly complex inside (e.g. many interfaces between the piece parts it consists of), this is not reflected by the complexity matrix – except for the number of parts within the component. It can thus happen that a component generally regarded as complex does not receive an accordingly high physical complexity coordinate in the complexity matrix.

[2] See Subsection 3.3.5 for more details on modularization.

- In the case study research I discovered that the complexity management model reaches its limits when it is applied to highly complex products. The number of components in the process industry compressor and railroad switch lock cases (Sections 5.4 and 5.5) is considered to indicate some sort of delimitation: above approx. 30 components the model's application starts to become cumbersome.[3]

- Because the model does not account for software and electronics, some misleading conclusions can arise. For example, a "star" component can be shifted into the "lucky strike" quadrant by replacing its mechanical parts by electronics and software. In this way, the contribution to physical complexity as defined by the model decreases, and the component moves to the left. It is questionable, though, whether overall complexity is successfully reduced or merely shifted elsewhere. The component's physical complexity has no doubt decreased, but product development and product support, for instance, become more complex and time-consuming due to the increased share of software and electronics.

- The complexity matrix provides a relative (and not absolute) frame. While the information can be used for one product (such as before and after optimization), comparing several different products (e.g. with respect to their "absolute" level of complexity) is not possible.[4]

- Only companies that sell highly customized products were considered in the case studies. These "differentiators" all try to introduce a certain level of standardization, increase their products' degree of modularity, and strive to cut complexity

---

[3] In the model's present version, the only method to avoid this situation is defining the constituting components differently (i.e. reducing the level of detail) so that their number remains below approx. 30. In many cases, however, this simplification does not model reality properly.

[4] In some cases however, products can be compared with each other if just their physical complexity is considered (i.e. the four complexity drivers). In such a way, all products can be assessed with regard to the number of parts they consist of, the number of variants they have, and the number and variety of interfaces to other products they share. This would provide some sort of "absolute" complexity measure for products. The functionality axis must necessarily be excluded from such calculations.

costs. No case studies were conducted at mass producers, even though the model claims to be also applicable for companies pursuing a cost leadership strategy.

I regard it as the most important limitation of my research that an immediate link from "physical complexity" to internal complexity and to complexity costs was not established. Assumptions are the only basis for connecting decreasing physical complexity with a cut in complexity costs, even though the fundamentals for such a connection were laid and the respective reasoning thoroughly elaborated on in Chapters 2 and 4. Due to this profound limitation, the model cannot directly estimate how much in complexity costs can be saved when the guidelines for action are implemented. The model fails to answer the often-asked – and fundamentally important – question "how much do your recommendations save us?"

The above limitations of the complexity management model were discovered during the case study research and provide the basis for suggestions for future work (see Section 6.4). The limitations must all be taken seriously but should not obscure the major research findings. Probably the most obvious limitation is the fact that decisions should never be based solely on the model presented in this work. Several viewpoints must always be considered to achieve well-founded complexity management.

6.3 Reflecting on the Research Methodology

The research methodology used in this work is action research conducted as case studies. According to Greenwood and Levin (1998, pp. 7-8), action research is characterized by three elements: research, participation, and action. Participating as a researcher in applying the complexity management model and investigating its effects proved to be an important feature of optimizing product architecture. Limiting my role to that of an outside observer would not have provided the necessary insight. Furthermore, my participation as a researcher did not essentially change the research setting, at least not to such an extent as it would have in situations where the research outcome can be strongly influenced by the researcher. Taking action and comparing product complexity before and after is one way of testing the effectiveness of the model. As one of this work's objectives is to provide practitioners with a model they can use for managing complexity, action by means of directly applying the model in an industry setting is an outstanding source of credibility. In summary, pursuing action research as a research methodology was very advantageous for achieving the research results in this work.

By using the framework of a series of case studies to apply the model, I was able to capture a good deal of the complex setting that surrounded the immediate application context. As was cited in Section 1.4 from Yin (2003, p. 13), case studies are the research methodology of choice when the research object and its surroundings cannot be clearly kept apart. Therefore, the many interrelationships between altering a product's architecture and the company's competitive strategy, its cost position, its functional areas, etc. do not have to be neglected and can be integrated in the research.

It has to be said, though, that the research object in this work is still more easily distinguished from its wider context than in other research settings where much more intertwined and less concrete problems are investigated. Therefore, it would have been an interesting alternative approach to drop the case study research idea of integrating the wider context and to focus on the mere product and its architecture. In such a way, the effects of applying the model can be identified more easily because the research object has been strongly simplified. This would greatly improve the ability to evaluate the model's effectiveness. However, it must be questioned whether such research that excludes its context would prove to be a viable option for industry practice. The desire to directly link cause and effect must always be weighted against the usefulness for practitioners.

6.4 Suggestions for Future Work

One never notices what has been done;
one can only see what remains to be done.

Marie Curie.[5]

As product complexity continues to pose a major challenge to enterprises, there is considerable potential to further elaborate on the research presented in this work. Assessing strategic and product life cycle issues, quantifying product complexity, and deriving guidelines aimed at enhancing product architecture are merely a first step. While working on the case studies, many ideas for future work came to my mind. Some of them are based on the limitations presented in Section 6.2, while others are general improvements and extensions of the complexity management model.

- For complex products (i.e. more than approx. 30 components), a robust numeric tool can offer improvements that should be considered in detail. In such a way, the product complexity assessment step becomes less cumbersome for complex products.
- Ways must be found to quantify complexity in non-mechanical products (services, electronics, and software) and products that consist of mechanical and non-mechanical parts. The model at present is restricted to mechanical products and fails to properly incorporate the non-mechanical portions of a product. A tremendous model improvement would be achieved by extending the model for non-mechanical products since most originally mechanical products nowadays consist of mechanical parts as well as software and electronics (e.g. in the area of mechatronics).

---

[5] As cited in "Quotable Women" (1994)

- Future work should also apply and test the complexity management model with mass producers and companies pursuing a hybrid competitive strategy, not only enterprises that concentrate on a differentiation strategy.

- Attempts should be made to describe and quantify the link from “physical complexity” to internal complexity and to complexity costs. For example, the effect of complexity drivers on complexity costs should be investigated, which in turn would lead to an understanding of what complexity drivers are more dominant than others. Additional complexity drivers not used so far could also be integrated if the present four drivers prove to be insufficient. In any case, an extended complexity management model must be able to estimate the complexity costs that can be saved thanks to its recommendations.

- The interdependencies between product complexity and product architecture must be investigated more closely to establish a direct connection between the factors causing complexity and their physical representation in the product. Questions such as “what are the major drivers of complexity in the product architecture (and why)?” must be addressed.

- Instead of (or in addition to) dividing the complexity matrix into four quadrants, an “optimal value zone” similar to the one used by target costing (see Figure 3.8) could be employed to account for the importance of the diagonal. “Money burning” components in the upper left corner of their quadrant (i.e. very close to the diagonal) would then no longer be earmarked as “bad” components, and “stars” in the lower right corner of their quadrant would receive more attention because they are far away from the diagonal.

- The model’s strategy and product life cycle assessment step is deliberately designed to be relatively open and flexible to adaptations by users. If strategic know-how is a problem for users or a more guided procedure is simply desired, a more structured approach should be sought (e.g. by outlining a strict strategy and product life cycle assessment procedure, by integrating key strategic figures, etc.).

- The model in its present form provides a relative frame and allows comparisons only within one product. If the model could be extended to provide comparisons between products, an absolute measure of product complexity could be created. A "product complexity benchmark" could then be established within a company that all product teams would strive to achieve.

- A possible way of providing a solution to the previous point is to express the two complexity matrix axes in monetary units (e.g. dollars). The functionality axis would not show percentages, but the component's share of the market price in dollars.[6] The abscissa would be more difficult to compute as it would account for complexity costs, i.e. the enterprise-internal effect of complexity.[7] In such a way, the complexity matrix would provide absolute data and enable comparisons across different products. I am convinced that future research should go in the direction of providing absolute measures of complexity that quantify the effects of complexity in dollar terms.

The above suggestions for future work are all fairly focused and are based on the complexity management model presented in this work. Even though the options mentioned are confined to the narrow field of improving and extending the model, the list does not mean to be complete. I am convinced that many other opportunities exist to build on this work's research, even more so as the complexity management model is designed to provide an open platform that allows for flexible adaptation.

---

[6] Assume that component A contributes 7.3 percent to the product's overall functionality. Let the product's market price (i.e. the amount customers are willing to pay for the benefit they receive) be $100. Thus, component A's value from a customer perspective is $7.30. This is an absolute measure and can be compared across different products.

[7] Complexity costs are generally difficult and cumbersome to determine. Moreover, the calculation can only be brought to a certain (and often insufficient) level of accuracy.

Appendix A List of Definitions

Table A.1 lists the definitions of important terms used throughout this work. It does not mean to be a complete glossary.

Table A.1 Summary of definitions

Term	Definition
Attribute	Any characteristic quality or property ascribed to a system, subsystem, or element
Complexity costs	The cost of indirect functions at a company and its suppliers that are caused by product variety
Component	Separable physical unit at the lowest hierarchic level of a product's structure of physical components; either an individual piece part or composed of several piece parts connected through interfaces; any distinct region of the product can be defined as a component
Coupled interface	Interface between two components resulting in the need to change both components if a change is made to only one component
Customer benefit	What the buyer receives by purchasing the product: functionality, assistance, warranty, brand name, etc.
Customer value	Customer benefit less total customer costs
Decoupled interface	Interface between two components enabling changes to be made to one component without affecting the other component
Element	Basic unit that cannot be decomposed further
Environment	Everything outside a system
External complexity	The sum of all influences external to the enterprise exerted on its products
Full profile	A full product description by one possible combination of its attributes and values (i.e. a complete description of one variant)
Functional element	Basic functional unit at the lowest hierarchic level of a product's structure of functionality; synonyms: functional requirement, functive, elemental function
Integral product architecture	Product architecture composed of strongly coupled (and therefore not autonomous) subassemblies; complex (non one-to-one) mapping from functional elements to components; coupled interfaces

Term	Definition
Internal complexity	The sum of all consequences within the enterprise its products entail by responding to the external complexity
Modular product architecture	Product architecture composed of relatively autonomous subassemblies, or modules; one-to-one mapping from functional elements to components; decoupled interfaces
Module	Relatively autonomous subassembly, often with few and standardized interfaces to other modules
Product architecture	The scheme by which a product's functionality is allocated to its physical components; consists of the structure of functionality, the structure of physical components, and the mapping from functionality to physical components
Product variety	Offering of several different product configurations
Relationship	Interaction between two elements
Structure of functionality	Hierarchic arrangement of a product's functionality; synonyms: function structure, function diagram, functional description, schematic description, functional structure
Structure of physical components	Hierarchic arrangement of a product's physical components and subassemblies
Subassembly	Clearly defined and bounded collection of individual piece parts and components with their corresponding interfaces
Subsystem	Clearly defined and bounded collection of elements and relationships within a system; a subsystem is itself a system
System	Clearly defined and bounded collection of elements and relationships
System complexity	A system's attribute characterized by a large number of elements and relationships, and a large diversity of elements and relationships
System structure	The way in which the elements are related to each other
Variant	One specific product configuration

Appendix B Classification of Strategies

In Subsection 4.2.1.4, a classification of strategies by Lampel and Mintzberg (1996) was introduced. Figure B.1 illustrates the five respective strategies and the location of their order penetrations points (OPP), i.e. the point where the product – thus far standardized – becomes determined by a particular customer order.

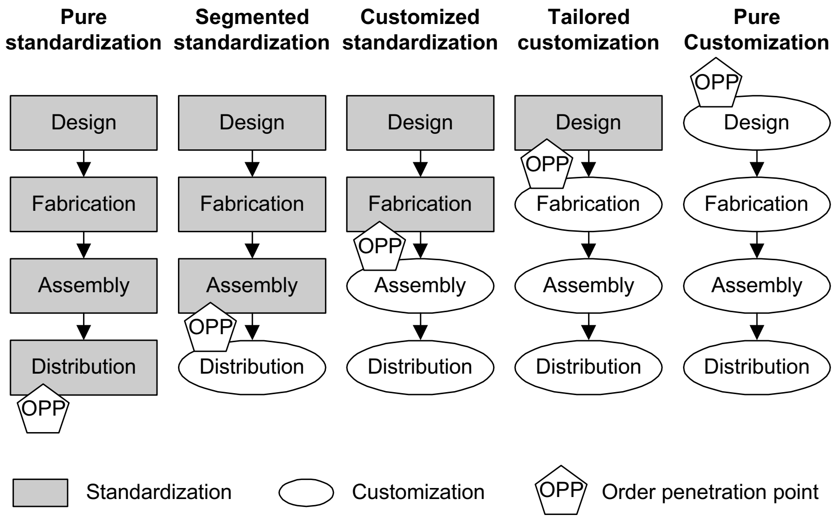


Figure B.1 Continuum of strategies; based on Lampel and Mintzberg (1996, p. 24), and Schuh and Schwenk (2001, p. 211)

Appendix C Ballpoint Pen Example

In Section 4.3, the introduction to the product complexity assessment step was illustrated by means of a ballpoint pen. Figure C.1 sketches the five components the ballpoint pen consists of. Note that the spacer is fixed between the front and rear housings when the pen is assembled.

Table C.1 lists the five components with all their variants. For example, the front housing occurs in four variants, numbered from 1.1 to 1.4. The front housing is a variable component, thus a “V” in the ID column. The last column provides information on what attributes the component variants depend on. As the front housing depends on the two attributes “surface” and “length” (see Table 4.4), variant 1.1 is used with the smooth surface and the length of 50 mm; variant 1.2 with the smooth surface and the 70 mm; variant 1.3 with the rough surface and the 50 mm; and so forth.

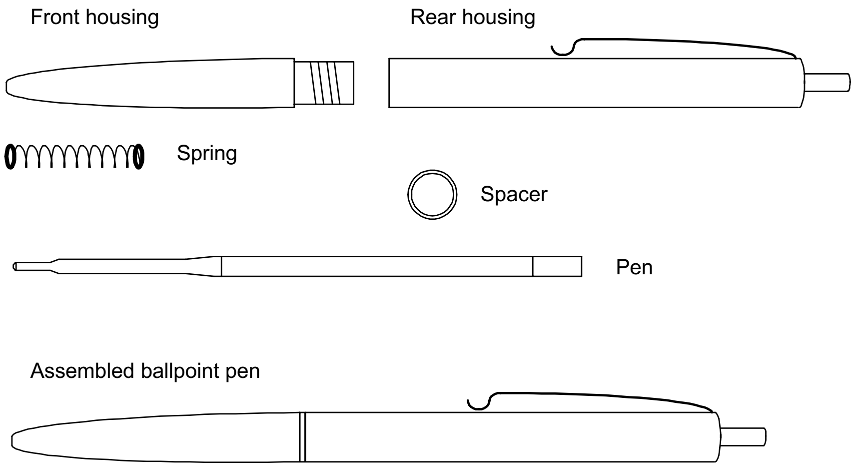


Figure C.1 Sketch of ballpoint pen showing all components

Table C.1 Bill of materials for the ballpoint pen

No.	ID	Component	Used with
1.1	V	Front housing	Smooth, 50 mm
1.2	V	Front housing	Smooth, 70 mm
1.3	V	Front housing	Rough, 50 mm
1.4	V	Front housing	Rough, 70 mm
2.1	V	Rear housing	Red, smooth
2.2	V	Rear housing	Blue, smooth
2.3	V	Rear housing	Green, smooth
2.4	V	Rear housing	Red, rough
2.5	V	Rear housing	Blue, rough
2.6	V	Rear housing	Green, rough
3.1	V	Pen	Red, 50 mm
3.2	V	Pen	Blue, 50 mm
3.3	V	Pen	Green, 50 mm
3.4	V	Pen	Red, 70 mm
3.5	V	Pen	Blue, 70 mm
3.6	V	Pen	Green, 70 mm
4.1	S	Spring	–
5.1	VO	Spacer	Red
5.2	VO	Spacer	Blue
5.3	VO	Spacer	Green

Appendix D Complexity Matrix Calculations

D.1 Calculating Physical Complexity

Subsection 4.3.3 introduced the procedure for calculating physical complexity – the abscissa coordinate in the complexity matrix. How the four complexity drivers (number of parts, number of variants, number of interfaces, and number of interface variants) are derived was also presented there. Physical complexity was defined by Equation 4.1, which is repeated here for convenience.

$$C_i = \alpha \cdot \frac{N_{e,i}}{N_{e,\max}} + \beta \cdot \frac{V_{e,i}}{V_{e,\max}} + \gamma \cdot \frac{N_{r,i}}{N_{r,\max}} + \delta \cdot \frac{V_{r,avg,i}}{V_{r,avg,\max}} \qquad \text{Equation 4.1}$$

The meaning of the symbols in Equation 4.1 is also repeated here:

C_i Physical complexity of component *i*

$N_{e,i}$ Number of elements (parts) constituting component *i*

$N_{e,max}$ Maximum occurring number of elements (parts) within a component of the product

$V_{e,i}$ Variety (number of variants) of component *i*

$V_{e,max}$ Maximum occurring variety (number of variants) of a component of the product

$N_{r,i}$ Number of relationships (interfaces) of component *i*

$N_{r,max}$ Maximum occurring number of relationships (interfaces) of a component of the product

$V_{r,avg,i}$ Average relationship variety of component *i* (average number of interface variants per interface)

$V_{r,avg,max}$ Maximum occurring average relationship variety of a component of the product (maximum average number of interface variants per interface)

In order to provide the full details of the calculation in Equation 4.1, the following explanations refer to the ballpoint pen example introduced in Section 4.3. Table 4.5 listed the four complexity driver values (or $N_{e,i}$, $V_{e,i}$, $N_{r,i}$, and $V_{r,avg,i}$ in terms of Equation 4.1) for every component of the ballpoint pen. The maximum value of every complexity driver (or $N_{e,max}$, $V_{e,max}$, $N_{r,max}$, *and* $V_{r,avg,max}$ in terms of Equation 4.1) can easily be determined from Table 4.5, as shown in the "maximum value" row of Table D.1. Next, the complexity driver values for each component are divided by the maximum values, shown in those columns of Table D.1 with fractions in their headings. These fractions are one of the inputs for Equation 4.1.

The remaining inputs still to be determined are the coefficients α, β, γ, and δ. The purpose of these coefficients is to ensure the same weighting of all four complexity drivers. To calculate the coefficients, the averages of all fractions for every complexity driver *(*$F_{1,avg}$, $F_{2,avg}$, $F_{3,avg}$, and $F_{4,avg}$*)* are computed in a first step (see last row of Table D.1). For the first complexity driver (number of parts), the average is defined as

$$F_{1,avg} = \frac{1}{n} \sum_{i=1}^{n} \frac{N_{e,i}}{N_{e,\max}},$$

where n is the number of components considered. For the ballpoint pen example, n equals five. For the other three complexity drivers, the averages are defined in an analogous way,

$$F_{2,avg} = \frac{1}{n} \sum_{i=1}^{n} \frac{V_{e,i}}{V_{e,\max}},$$

$$F_{3,avg} = \frac{1}{n} \sum_{i=1}^{n} \frac{N_{r,i}}{N_{r,\max}}, \text{ and}$$

$$F_{4,avg} = \frac{1}{n} \sum_{i=1}^{n} \frac{V_{r,avg,i}}{V_{r,avg,\max}}.$$

Table D.1 Inputs for Equation 4.1 for the ballpoint pen example

Component	$N_{e,i}$	$\frac{N_{e,i}}{N_{e,max}}$	$V_{e,i}$	$\frac{V_{e,i}}{V_{e,max}}$	$N_{r,i}$	$\frac{N_{r,i}}{N_{r,max}}$	$V_{r,avg,i}$	$\frac{V_{r,avg,i}}{V_{r,avg,max}}$
Front housing	1	0.2	4	0.667	4	1	1.25	0.75
Rear housing	5	1	6	1	3	0.75	1.333	0.8
Pen	4	0.8	6	1	3	0.75	1.667	1
Spring	1	0.2	1	0.167	2	0.5	1	0.6
Spacer	1	0.2	4	0.667	2	0.5	1	0.6
Maximum value	$N_{e,max}$		$V_{e,max}$		$N_{r,max}$		$V_{r,avg,max}$	
	5		6		4		1.667	
Average value		$F_{1,avg}$		$F_{2,avg}$		$F_{3,avg}$		$F_{4,avg}$
		0.48		0.7		0.7		0.75

Because all complexity drivers receive the same weighting, $F_{1,avg}$, $F_{2,avg}$, $F_{3,avg}$, and $F_{4,avg}$ must be scaled down to

$$w_{CD} = \frac{1}{n_{CD}},$$

where n_{CD} is the number of complexity drivers considered. In the ballpoint pen example, w_{CD} equals 0.25 since all four complexity drivers are part of the calculation.[1] The scaling factor for the first complexity driver is therefore defined as

$$\alpha' = \frac{w_{CD}}{F_{1,avg}}.$$

For the other complexity drivers, the scaling factors are defined in the same way,

---

[1] If one complexity driver is not considered (e.g. number of parts if the components are individual piece parts), w_{CD} equals 0.333 (because $n_{CD} = 3$).

$$\beta' = \frac{w_{CD}}{F_{2,avg}},$$

$$\gamma' = \frac{w_{CD}}{F_{3,avg}}, \text{ and}$$

$$\delta' = \frac{w_{CD}}{F_{4,avg}}.$$

For the ballpoint pen example, the scaling factors are as follows: $\alpha' = 0.5208$, $\beta' = 0.3571$, $\gamma' = 0.3571$, and $\delta' = 0.3333$. If $F_{1,avg}$, $F_{2,avg}$, $F_{3,avg}$, and $F_{4,avg}$ are multiplied by α', β', γ', and δ', respectively, all four complexity drivers are weighted equally.

Because the physical complexity C_i of each component must be between zero and one (i.e. $0 \leq C_i \leq 1$), the sum of the coefficients α, β, γ, and δ must be one, too,

$$\alpha + \beta + \gamma + \delta = 1.$$

Therefore, the scaling factors α', β', γ', and δ' are scaled by their sum:

$$\alpha = \frac{\alpha'}{\alpha' + \beta' + \gamma' + \delta'};$$

$$\beta = \frac{\beta'}{\alpha' + \beta' + \gamma' + \delta'};$$

$$\gamma = \frac{\gamma'}{\alpha' + \beta' + \gamma' + \delta'};$$

$$\delta = \frac{\delta'}{\alpha' + \beta' + \gamma' + \delta'}.$$

For the ballpoint pen example, the coefficients are as follows: $\alpha = 0.33207$, $\beta = 0.22770$, $\gamma = 0.22770$, and $\delta = 0.21252$. Now that the coefficients have been computed, the physical complexity of all components can readily be calculated:

$$C_{front\ housing} = 0.33207 \cdot \frac{1}{5} + 0.22770 \cdot \frac{4}{6} + 0.22770 \cdot \frac{4}{4} + 0.21252 \cdot \frac{1.25}{1.667} = 0.60531\,;$$

$$C_{rear\ housing} = 0.33207 \cdot \frac{5}{5} + 0.22770 \cdot \frac{6}{6} + 0.22770 \cdot \frac{3}{4} + 0.21252 \cdot \frac{1.333}{1.667} = 0.90057\,;$$

$$C_{pen} = 0.33207 \cdot \frac{4}{5} + 0.22770 \cdot \frac{6}{6} + 0.22770 \cdot \frac{3}{4} + 0.21252 \cdot \frac{1.667}{1.667} = 0.87666\,;$$

$$C_{spring} = 0.33207 \cdot \frac{1}{5} + 0.22770 \cdot \frac{1}{6} + 0.22770 \cdot \frac{2}{4} + 0.21252 \cdot \frac{1}{1.667} = 0.34573\,;$$

$$C_{spacer} = 0.33207 \cdot \frac{1}{5} + 0.22770 \cdot \frac{4}{6} + 0.22770 \cdot \frac{2}{4} + 0.21252 \cdot \frac{1}{1.667} = 0.45958\,.$$

D.2 Quadrant Borders in the Complexity Matrix

Table 4.6 listed the functionality and physical complexity coordinates of the ballpoint pen components. It is repeated here for convenience (see Table D.2). All components receive their specific location within the complexity matrix based on these coordinates. Which one of the four quadrants they are part of is determined by the borders between the quadrants.

The vertical border between the left and right halves of the complexity matrix is always located at the 0.5 coordinate of the abscissa. The maximum physical complexity value a component can theoretically attain is one. Those components with a physical complexity above half of the maximum possible value are therefore to the right of the vertical border. And those components with a physical complexity below half of the maximum possible value are located to the left of the vertical border.

The horizontal border dividing the complexity matrix into an upper and lower half is defined by the average of the maximum and minimum functionality coordinates. For the ballpoint pen example, the maximum occurring functionality coordinate is 40% (pen), while the minimum is 2% (spacer). Thus, the horizontal border is located at 21%. Figure D.1 depicts the complexity matrix for the ballpoint pen with the four quadrants.

Table D.2 Functionality and physical complexity coordinates of the ballpoint pen

Component	Functionality	Physical complexity
Front housing	20%	0.60531
Rear housing	26%	0.90057
Pen	40%	0.87666
Spring	12%	0.34573
Spacer	2%	0.45958

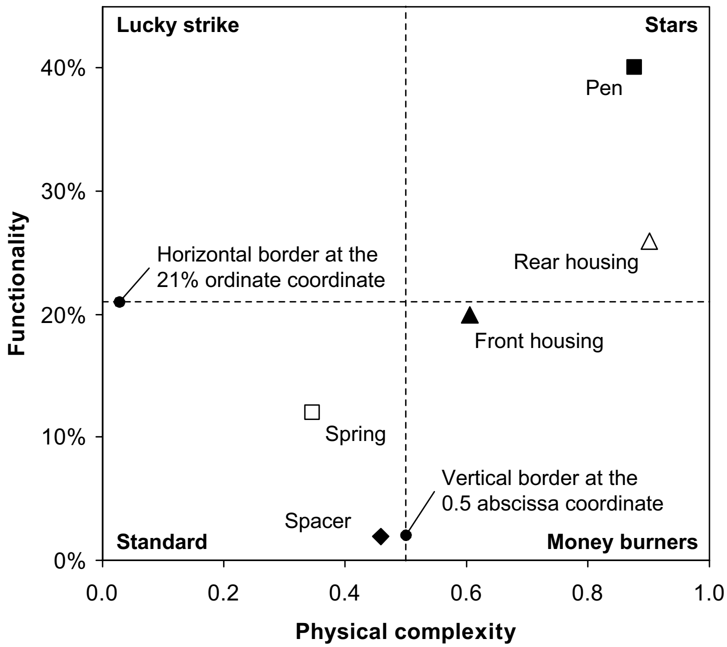


Figure D.1 Complexity matrix for the ballpoint pen example

D.3 Taking the Logarithm for Complexity Driver Calculations

In some cases, the complexity driver values of certain components are very large compared to the other components.[2] These values would exert a disproportionately heavy influence on the complexity matrix if they were to be taken as the basis for the physical complexity calculations (as they normally would). Therefore, the logarithm with base ten is used in these cases to moderate the large differences.[3] A component with many variants is still detected by the model as a highly variable component and can be distinguished from a component with few variants. But the advantage is that those components with large complexity driver values do not get an unduly high physical complexity coordinate.

What would happen if the logarithm procedure were not employed in products consisting of components with widely varying complexity driver values? The complexity drivers would then be dominated by the components with large values, and they would essentially be canceled out for the components with low values. The pattern I discovered in my case study research was as follows: most components showed rather low complexity driver values and only a few components had medium to high values (see Figure D.2 as an example). The components would thus almost become digitalized – a few components with values close to one and most with values close to zero.[4]

The effect of calculating the logarithm with base ten for the number of variants is shown in Figure D.3. The modules are now distributed more evenly, avoiding the "digitalization" mentioned above. Even though the logarithm procedure as presented here proved to be a viable method, it is merely a first attempt to cope with the problem

---

[2] For example, in the case study about the liquid handling platform the number of variants ranged from one for standardized modules all the way to 1,836 for the most variable module (see Table 5.8).

[3] Note that the logarithm procedure as presented here was employed in the liquid handling platform case study for the two complexity drivers "number of parts" and "number of variants." In the process industry compressor case study, the logarithm with base ten was taken for the "number of variants" complexity driver.

[4] Recall that the complexity driver values are divided by the maximum occurring value, thus giving numbers between zero and one.

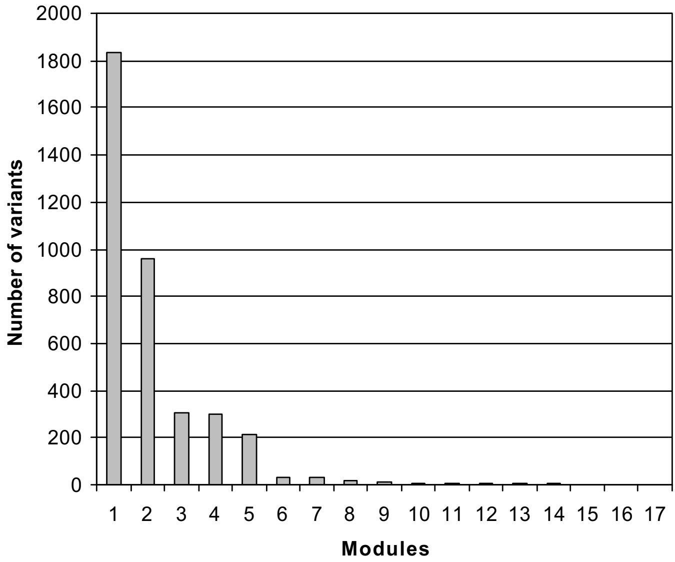


Figure D.2 Number of variants for the modules of the liquid handling platform[5]

of components having widely ranging complexity driver values. Future research involving a model refinement must address this phenomenon and provide additional solutions.

[5] Note that the module numbers in Figure D.2 do not correspond with the module numbers in Figure 5.7.

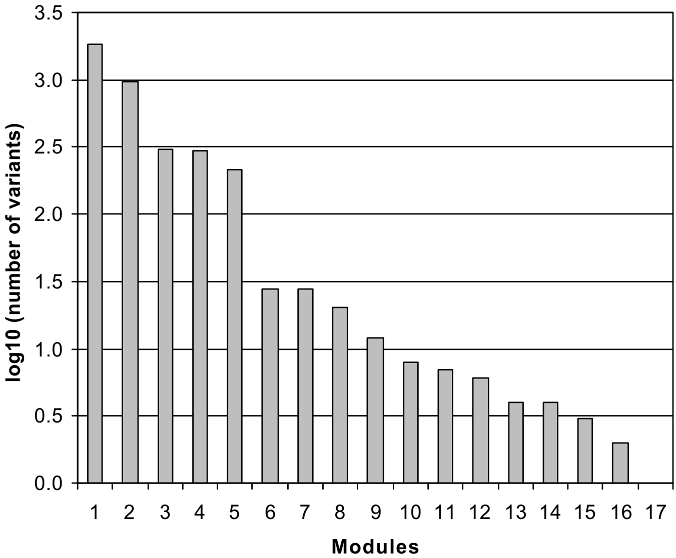


Figure D.3 Logarithm with base ten of the number of variants (liquid handling case study)[6]

[6] Note that the module numbers in Figure D.3 do not correspond with the module numbers in Figure 5.7.

Reference List

Adam, D., & Johannwille, U. (1998). Die Komplexitätsfalle. In D. Adam (Ed.), *Komplexitätsmanagement* (pp. 5-28). Wiesbaden: Gabler.

Akao, Y. (Ed.). (1990). *Quality function deployment: Integrating customer requirements into product design.* New York: Productivity Press.

Baldwin, C. Y., & Clark, K. B. (1997). Managing in an age of modularity. *Harvard Business Review, 75* (5), 84-93.

Baldwin, C. Y., & Clark, K. B. (1999). *Design rules: The power of modularity.* Boston: MIT Press.

Begeisterung für das Besondere. (2007, January 3). *Neue Zürcher Zeitung,* p. 36.

Bliss, C. (2000). *Management von Komplexität: Ein integrierter, systemtheoretischer Ansatz zur Komplexitätsreduktion.* Wiesbaden: Gabler.

Bongulielmi, L. (2003). *Die Konfigurations- und Verträglichkeitsmatrix als Beitrag zur Darstellung konfigurationsrelevanter Aspekte im Produktentstehungsprozess.* Düsseldorf: VDI.

Boutellier, R., Dinger, H., & Lee, H. (1997). Plattformen – ein Erfolgsfaktor im Wettbewerbsdruck. *Technische Rundschau, 89* (37-38), 58-61.

Brown, J. S., & Hagel III, J. (2005). Innovation blowback: Disruptive management practices from Asia. *The McKinsey Quarterly* (1), 35-45.

Browning, T. R. (2001). Applying the design structure matrix to system decomposition and integration problems: A review and new directions. *IEEE Transactions on Engineering Management, 48* (3), 292-306.

Bryman, A. (1999). The debate about quantitative and qualitative research. In A. Bryman & R. G. Burgess (Eds.), *Qualitative research, 1* (pp. 35-69). London: Sage Publications.

Burn, G. R. (1990). Quality function deployment. In B. G. Dale & J. J. Plunkett (Eds.), *Managing Quality* (pp. 66-88). London: Philip Allan.

Caesar, C. (1991). *Kostenorientierte Gestaltungsmethodik für variantenreiche Serienprodukte: Variant Mode and Effects Analysis (VMEA).* Düsseldorf: VDI.

Campbell, D. J. (1988). Task complexity: A review and analysis. *Academy of Management Review 13* (1), 40-52.

Checkland, P. (1981). *Systems thinking, systems practice.* Chichester et al.: John Wiley & Sons.

Child, P., Diederichs, R., Sanders, F.-H., & Wisniowski, S. (1991). SMR Forum: The management of complexity. *Sloan Management Review 33* (1), 73-80.

Clark, K. B., & Fujimoto, T. (1990). The power of product integrity. *Harvard Business Review,* Nov.-Dec., 107-118.

Collier, D. A. (1981). The measurement and operating benefits of component part commonality. *Decision Sciences, 12* (1), 85-97.

Cooper, R., & Kaplan, R. S. (1988a). How cost accounting distorts product costs. *Management Accounting 69* (10), 20-27.

Cooper, R., & Kaplan, R. S. (1988b). Measure costs right: Make the right decisions. *Harvard Business Review,* Sept.-Oct., 96-103.

Corsten, H., & Will, T. (1993). Reflections on competitive strategy and its impact on modern production concepts. *Management International Review, 33* (4), 315-334.

Cummings, P. (1991). Symptoms of complexity. *The McKinsey Quarterly 27* (4), 60-61.

Davis, S. M. (1987). *Future perfect.* Reading, MA: Addison-Wesley.

Dhalla, N. K., & Yuspeh, S. (1976). Forget the product life cycle concept! *Harvard Business Review, 54* (1), 102-112.

Dyllick, T., & Probst, G. J. B. (1984). Einführung in die Konzeption der systemorientierten Managementlehre von Hans Ulrich. In T. Dyllick & G. J. B. Probst (Eds.), *Management* (pp. 9-17). Bern: Paul Haupt.

Edmondson, G. (2005, November 28). A nice little thing going. *Business Week European Edition,* 20.

Eisenhardt, K. M. (1989). Building theories from case study research. *Academy of Management Review, 14* (4), 532-550.

Erixon, G. (1998). *Modular function deployment – A method for product modularisation.* Stockholm: The Royal Institute of Technology.

Feitzinger, E., & Lee, H. L. (1997). Mass customization at Hewlett-Packard: The power of postponement. *Harvard Business Review, 75* (1), 116-121.

Fisher, M., Ramdas, K., and Ulrich, K. (1999). Component sharing in the management of product variety: A study of automotive braking systems. *Management Science, 45* (3), 297-315.

Fleck, A. (1995). *Hybride Wettbewerbsstrategien: Zur Synthese von Kosten- und Differenzierungsvorteilen.* Wiesbaden: Deutscher Universitäts-Verlag.

Flood, R. L., & Carson, E. R. (1988). *Dealing with complexity: An introduction to the theory and application of systems science.* New York: Plenum Press.

Friedli, T. (2004). *Strategisches Produktionsmanagement: Management produzierender Unternehmen im Spannungsfeld zwischen Fokus und Flexibilität.* St. Gallen: Universität St. Gallen.

Fritz, W., & Oelsnitz, D. (1996). *Marketing: Elemente marktorientierter Unternehmensführung.* Stuttgart: Kohlhammer.

Gilbert, X., Strebel, P. (1987). Strategies to outpace the competition. *The Journal of Business Strategy, 8* (1), 28-36.

Gilmore, J. H., & Pine II, B. J. (1997). The four faces of mass customization. *Harvard Business Review,* Jan.-Feb., 91-101.

Göpfert, J. (1998). *Modulare Produktentwicklung: Zur gemeinsamen Gestaltung von Technik und Organisation.* Wiesbaden: Deutscher Universitäts-Verlag.

Göpfert, J., & Steinbrecher, M. (2000). Modulare Produktentwicklung leistet mehr. *Harvard Business Manager, 22 (3),* 20-30.

Govers, C. P. M. (1996). What and how about quality function deployment (QFD). *International Journal of Production Economics, 46-47,* 575-585.

Green, P. E., & Krieger, A. M. (1991). Segmenting markets with conjoint analysis. *Journal of Marketing, 55* (4), 20-31.

Green, P. E., & Rao, V. R. (1971). Conjoint measurement for quantifying judgmental data. *Journal of Marketing Research, 8,* 355-363.

Green, P. E., & Srinivasan, V. (1990). Conjoint analysis in marketing: New developments with implications for research and practice. *Journal of Marketing, 54* (4), 3-19.

Green, P. E., & Srinivasan, V. (1978). Conjoint analysis in consumer research: Issues and outlook. *The Journal of Consumer Research, 5* (2), 103-123.

Green, P. E., & Wind, Y. (1975). New way to measure consumers' judgments. *Harvard Business Review,* Jul.-Aug., 107-117.

Greenwood, D. J., & Levin, M. (1998). *Introduction to action research: Social research for social change.* Thousand Oaks: Sage Publications.

Griffin, A. (1992). Evaluating QFD's use in US firms as a process for developing products. *Journal of Product Innovation Management 9,* 171-187.

Hall, W. K. (1980). Survival strategies in a hostile environment. *Harvard Business Review, 58* (5), 75-92.

Hauser, J. R., & Clausing, D. (1988). The house of quality. *Harvard Business Review,* May-Jun., 63-73.

Hax, A. C., & Majluf, N. S. (1982). Competitive cost dynamics: The experience curve. *Interfaces, 12* (5), 50-61.

Heina, J. (1999). *Variantenmanagement: Kosten-Nutzen-Bewertung zur Optimierung der Variantenvielfalt.* Wiesbaden: Deutscher Universitäts-Verlag.

Hichert, R. (1986a). Probleme der Vielfalt, Teil 2: Was kostet eine Variante? *wt-Zeitschrift für industrielle Fertigung 76,* 141-145.

Hichert, R. (1986b). Probleme der Vielfalt, Teil 3: Was bestimmt die optimale Erzeugnisvielfalt? *wt-Zeitschrift für industrielle Fertigung 76,* 673-676.

Hichert, R. (1987). Probleme der Vielfalt, Teil 4: Erzeugnisvielfalt im Wettbewerbsvergleich. *wt-Zeitschrift für industrielle Fertigung 77,* 223-227.

Hill, C. W. L. (1988). Differentiation versus low cost or differentiation and low cost: A contingency framework. *Academy of Management Review, 13* (3), 401-412.

Hino, S. (2006). *Inside the mind of Toyota: Management principles for enduring growth.* New York: Productivity Press.

Hiromoto, T. (1988). Another hidden edge – Japanese management accounting. *Harvard Business Review,* Jul.-Aug., 22-26.

Hobday, M., Rush, H., & Tidd, J. (2000). Innovation in complex products and systems. *Research Policy 29,* 793-804.

Hofer, A. P. (2001). Management von Produktfamilien: Wettbewerbsvorteile durch Plattformen (Doctoral dissertation, University of St. Gallen, 2001). *University of St. Gallen,* 2495.

Horváth, P., & Mayer, R. (1989). Prozesskostenrechnung: Der neue Weg zu mehr Kostentransparenz und wirkungsvolleren Unternehmensstrategien. *Controlling, 1* (4), 214-219.

Horváth, P., & Seidenschwarz, W. (1992). Zielkostenmanagement. *Controlling, 4* (3), 142-150.

Hubka, V., & Eder, W. E. (1988). *Theory of technical systems: A total concept theory for engineering design.* Berlin: Springer.

Johnson, C. N. (2003). QFD explained. *Quality Progress 36* (3), 104.

Johnson, H. T. (1992). It's time to stop overselling activity-based concepts. *Management Accounting,* Sept., 26-35.

Kaiser, A. (1995). Integriertes Variantenmanagement mit Hilfe der Prozesskostenrechnung (Doctoral dissertation, University of St. Gallen, 1995). *University of St. Gallen,* 1742.

Kaluza, B. (1989). *Erzeugniswechsel als unternehmenspolitische Aufgabe: Integrative Lösungen aus betriebswirtschaftlicher und ingenieurwissenschaftlicher Sicht.* Berlin: Schmidt.

Kano, N., Seraku, N., Takahashi, F., & Tsuji, S. (1984). Attractive quality and must-be quality. *Hinshitsu, 14* (2), 39-48.

Kirchhof, R. (2003). *Ganzheitliches Komplexitätsmanagement: Grundlagen und Methodik des Umgangs mit Komplexität im Unternehmen.* Wiesbaden: Deutscher Universitäts-Verlag.

Klir, G. J., & Elias, D. (2003). *Architecture of systems problem solving* (2nd ed.). New York: Kluwer Academic.

Kotha, S. (1995). Mass customization: Implementing the emerging paradigm for competitive advantage. *Strategic Management Journal, 16,* 21-42.

Kotha, S. (1996). From mass production to mass customization: The case of the National Industrial Bicycle Company of Japan. *European Management Journal, 14* (5), 442-450.

Kotler, P., & Keller, K. (2006). *Marketing management* (12th ed.). Upper Saddle River: Prentice Hall.

Lampel, J., & Mintzberg, H. (1996). Customizing customization. *Sloan Management Review, 38* (1), 21-30.

Lancaster, K. (1990). The economics of product variety: A survey. *Marketing Science 9* (3), 189-206.

Leenders, M. R., & Erskine, J. A. (1989). *Case research: The case writing process* (3rd ed.). London: The University of Western Ontario.

Levering, V. (2003). Mass Customization: Ein Konzept für das Komplexitätsmanagement von Investitionsgütern (Doctoral dissertation, University of St. Gallen, 2003). *University of St. Gallen,* 2718.

Levitt, T. (1980). Marketing success through differentiation – of anything. *Harvard Business Review, 58* (1), 83-91.

Lexico Publishing Group. (2007). Retrieved February 26, 2007, from http://dictionary.reference.com/browse/complex

Liker, J. K. (2004). *The Toyota way: 14 management principles from the world's greatest manufacturer.* New York: McGraw Hill.

Löfgren, M., & Witell, L. (2005). Kano's theory of attractive quality and packaging. *The Quality Management Journal 12* (3), 7-20.

Luce, R. D., & Tukey, J. W. (1964). Simultaneous conjoint measurement: A new type of fundamental measurement. *Journal of Mathematical Psychology 1,* 1-27.

Martin, M. V., & Ishii, K. (1996). Design for variety: A methodology for understanding the costs of product proliferation. Proceedings of: *The 1996 ASME Design Engineering Technical Conferences.* Irvine, CA: ASME.

Martin, M. V., & Ishii, K. (1997). Design for variety: Development of complexity indices and design charts. Proceedings of: *The 1997 ASME Design Engineering Technical Conferences.* Sacramento, CA: ASME.

Matern, C. (2000). *Systemwirtschaftliches Variantenmanagement: Operationalisierung und Quantifizierung des integrierten Variantenmanagements.* Göttingen: Vandenhoeck und Ruprecht.

Mayer, R. (1993). *Strategien erfolgreicher Produktgestaltung: Individualisierung und Standardisierung.* Wiesbaden: Deutscher Universitäts-Verlag.

McCarthy, E. J. (1996). *Basic marketing: A managerial approach* (12^{th} ed.). Homewood, IL: Irwin.

Meyer, M. H., & Lehnerd, A. P. (1997). *The power of product platforms: Building value and cost leadership.* New York: Free Press.

Meyer, M. H., & Utterback, J. M. (1993). The product family and the dynamics of core capabilities. *Sloan Management Review, 34* (3), 29-47.

Mikkola, J. H. (2006). Capturing the degree of modularity embedded in product architectures. *Journal of Product Innovation Management, 23* (2), 128-146.

Miller, A., & Dess, G. G. (1993). Assessing Porter's (1980) model in terms of its generalizability, accuracy and simplicity. *Journal of Management Studies, 30* (4), 553-585.

Miller, J. G., & Vollmann, T. E. (1985). The hidden factory. *Harvard Business Review,* Sept.-Oct., 142-150.

Moore, W. L., Louviere, J. J., & Verma, R. (1999). Using conjoint analysis to help design product platforms. *Journal of Product Innovation Management 16,* 27-39.

Murray, A. I. (1988). A contingency view of Porter's "generic strategies." *Academy of Management Review, 13* (3), 390-400.

Needles, B. E., Powers, M., & Crosson, S. V. (2002). *Principles of accounting.* Boston: Houghton Mifflin.

Page, A. L., & Rosenbaum, H. F. (1987). Redesigning product lines with conjoint analysis: How Sunbeam does it. *Journal of Product Innovation Management 4,* 120-137.

Pahl, G., & Beitz, W. (2003). *Engineering design: A systematic approach* (2nd ed.). London: Springer.

Patzak, G. (1982). *Systemtechnik – Planung komplexer innovativer Systeme.* Berlin et al.: Springer.

Piller, F. T. (2003). *Mass Customization: Ein wettbewerbsstrategisches Konzept im Informationszeitalter* (3rd ed.). Wiesbaden: Deutscher Universitäts-Verlag.

Pimmler, T. U., & Eppinger, S. D. (1994). Integration analysis of product decompositions. In T. K. Hight & F. Mistree (Eds.), *Design theory and methodology – DTM '94, DE-Vol. 68* (pp. 343-351). New York: The American Society of Mechanical Engineers.

Pine II, B. J. (1993a). *Mass customization: The new frontier in business competition.* Boston: Harvard Business School Press.

Pine II, B. J. (1993b). Mass customizing products and services. *Planning Review 22* (4), 6-13, 55.

Pine II, B. J., Victor, B., & Boynton, A. C. (1993). Making mass customization work. *Harvard Business Review 71* (5), 108-119.

Porter, M. E. (1980). *Competitive strategy: Techniques for analyzing industries and competitors.* New York: Free Press.

Porter, M. E. (1985). *Competitive advantage: Creating and sustaining superior performance.* New York: Free Press.

Porter, M. E. (1996). What is strategy? *Harvard Business Review 74* (6), 61-78.

Pulkkinen, A., Lehtonen, T., & Riitahuhta, A. (1999). Design for configuration – Methodology for product family development. In U. Lindemann, H. Birkhofer, H. Meerkamm & S. Vajna (Eds.), *Proceedings of the 12th International Conference on Engineering Design* (pp. 1495-1500). München: Technische Universität München.

Punch, K. F. (2005). *Introduction to social research: Quantitative and qualitative approaches.* London: Sage Publications.

Quotable women: A collection of shared thoughts. (1994). Philadelphia: Running Press.

Ramdas, K. (2003). Managing product variety: An integrative review and research directions. *Production and Operations Management 12* (1), 79-101.

Rapp, T. (1999). *Produktstrukturierung: Komplexitätsmanagement durch modulare Produktstrukturen.* Wiesbaden: Deutscher Universitäts-Verlag.

Rathnow P. J. (1993). *Integriertes Variantenmanagement: Bestimmung, Realisierung und Sicherung der optimalen Produktvielfalt.* Göttingen: Vandenhoeck und Ruprecht.

Riitahuhta, A. (2001). Views and experiences of configuration management. In A. Riitahuhta & A. Pulkkinen (Eds.), *Design for configuration: A debate based on the 5th WDK workshop on product structuring* (pp. 1-21). Berlin: Springer.

Robertson, D., & Ulrich, K. (1998). Planning for product platforms. *Sloan Management Review, 39* (4), 19-31.

Rommel, G., Brück, F., Diederichs, R., Kempis, R., & Kluge, J. (1993). *Einfach überlegen – Das Unternehmenskonzept, das die Schlanken schlank und die Schnellen schnell macht.* Stuttgart: Schäffer-Poeschel.

Sakurai, M., & Keating, P. J. (1994). Target Costing und Activity-Based Costing. *Controlling, 6* (2), 84-91.

Schönsleben, P. (2002). *Integrales Logistikmanagement.* Berlin: Springer.

Schuh, G. (1989). *Gestaltung und Bewertung von Produktvarianten: Ein Beitrag zur systematischen Planung von Serienprodukten.* Düsseldorf: VDI.

Schuh, G., & Schwenk, U. (2001). *Produktkomplexität managen: Strategien, Methoden, Tools.* München: Hanser.

Schulte, C. (1991). Aktivitätsorientierte Kostenrechnung: Eine Strategie zur Variantenreduktion. *Controlling, 3* (1), 18-23.

Seghezzi, H. D. (2003). *Integriertes Qualitätsmanagement: Das St. Galler Konzept* (2nd ed.). München: Hanser.

Seidenschwarz, W. (1991). Target Costing: Ein japanischer Ansatz für das Kostenmanagement. *Controlling, 3* (4), 198-203.

Seiler, A. (1998). *Accounting: BWL in der Praxis 1.* Zürich: Orell Füssli.

Seiler, A. (2000a). *Marketing: BWL in der Praxis 4.* Zürich: Orell Füssli.

Seiler, A. (2000b). *Planning: BWL in der Praxis 3.* Zürich: Orell Füssli.

Sekolec, R. (2005). *Produktstrukturierung als Instrument des Variantenmanagements in der methodischen Entwicklung modularer Produktfamilien.* Düsseldorf: VDI.

Shank, J. K., & Fisher, J. (1999). Target costing as a strategic tool. *Sloan Management Review,* Fall, 73-82.

Shimano. (2006). Retrieved September 13, 2006, from http://cycle.shimano-eu.com/publish/ content/cycle/seh/nl/en.html

Simon, H. A. (1962). The architecture of complexity. *Proceedings of the American Philosophical Society 106* (6), 467-482.

Simon, H. A. (1973). The organization of complex systems. In H. H. Pattee (Ed.), *Hierarchy theory: The challenge of complex systems* (pp. 1-27). New York: George Braziller.

Susman, G. I., & Evered, R. D. (1978). An assessment of the scientific merits of action research. *Administrative Science Quarterly, 23* (4), 582-603.

Suzue, T., & Kohdate, A. (1990). *Variety reduction program: A production strategy for product diversification.* Cambridge, MA: Productivity Press.

Tanaka, M. (1989). Cost planning and control systems in the design phase of a new product. In Y. Monden & M. Sakurai (Eds.), *Japanese management accounting: a world class approach to profit management* (pp. 49-71). Portland: Productivity Press.

Teboul, J. (1991). *Managing quality dynamics.* New York et al.: Prentice Hall.

Thomke, S., & Reinertsen, D. (1998). Agile product development: Managing development flexibility in uncertain environments. *California Management Review, 41* (1), 8-30.

Tidd, J., Bessant, J., & Pavitt, K. (2001). *Managing innovation: Integrating technological, market and organizational change* (2nd ed.). Chichester et al.: John Wiley & Sons.

Thonemann, U. W., & Brandeau, M. L. (2000). Optimal commonality in component design. *Operations Research 48* (1), 1-19.

Tseng, M. M., & Jiao, J. (2001). Mass customization. In G. Salvendy (Ed.), *Handbook of industrial engineering: technology and operations management* (3rd ed., pp. 684-709). New York: John Wiley.

Turney, P. B. B. (1992). Activity-based management. *Management Accounting,* Jan., 20-25.

Tushman, M. L., & Rosenkopf, L. (1992). Organizational determinants of technological change: Toward a sociology of technological evolution. In L. L. Cummings & B. M. Straw (Eds.), *Research in organizational behavior 14* (pp. 311-347). Stamford, CT: JAI Press.

Ulrich, H. (1981). Die Betriebswirtschaftslehre als anwendungsorientierte Sozialwissenschaft. In M. N. Geist & R. Köhler (Eds.), *Die Führung des Betriebes* (pp. 1-25). Stuttgart: Poeschel.

Ulrich, K. (1995). The role of product architecture in the manufacturing firm. *Research Policy, 24* (3), 419-440.

Ulrich, K. T., & Eppinger, S. D. (1995). *Product design and development.* New York: McGraw-Hill.

Ulrich, K., & Tung, K. (1991). Fundamentals of product modularity. In A. Sharon, R. Behun, F. Prinz, & L. Young (Eds.), *Issues in design / manufacture integration, DE-Vol. 39* (pp. 73-79). New York: The American Society of Mechanical Engineers.

Ulrich, P., & Hill, W. (1979). Wissenschaftstheoretische Aspekte ausgewählter betriebswirtschaftlicher Konzeptionen. In H. Raffée & B. Abel (Eds.), *Wissenschaftstheoretische Grundfragen der Wirtschaftswissenschaften* (pp. 161-190). München: Verlag Franz Vahlen.

Vriens, M. (1994). Solving marketing problems with conjoint analysis. *Journal of Marketing Management 10,* 37-55.

White, R. E. (1986). Generic business strategies, organizational context and performance: An empirical investigation. *Strategic Management Journal, 7* (3), 217-231.

Wildemann, H. (1993). Komplexitätskostenrechung durch Variantenmanagement. In C. Nedess (Ed.), *Produktion im Umbruch – Herausforderung an das Management* (pp. 297-337).

Womack, J. P., Jones, D. T., & Roos, D. (1990). *The machine that changed the world.* New York: Rawson Associates.

Womack, J. P., & Jones, D. T. (1994). From lean production to the lean enterprise. *Harvard Business Review, 72* (2), 93-103.

Womack, J. P., & Jones, D. T. (2003). *Lean thinking: Banish waste and create wealth in your corporation* (2nd ed.). New York: Free Press.

Worthy, F. S. (1991, August 12). Japan's smart secret weapon. *Fortune, 124* (4), 72-75.

Wright, T. P. (1936). Factors affecting the cost of airplanes. *Journal of Aeronautical Sciences, 3* (4), 122-128.

Yin, R. K. (2003). *Case study research: Design and methods* (3rd ed.). Thousand Oaks: Sage Publications.