

Capturing the Degree of Modularity Embedded in Product Architectures*

Juliana H. Mikkola

This article focuses on integrating various perspectives on product architecture modularity into a general framework and proposes a way to measure the degree of modularization embedded in product architectures. The article addresses trade-offs between modular and integral product architectures and how components and interfaces influence the degree of modularization. The article identifies the following key elements of product architecture modularity: components (standard and new-to-the-firm), interfaces (standardization and specification), degree of coupling, and substitutability. A mathematical model, termed the modularization function, is applied to measure the key elements and their combined effect on the degree of modularization embedded in product architectures. The application of the modularization function is illustrated by two distinct sets of product architectures: Chrysler Jeep's windshield wipers controllers and Schindler's hydraulic and traction-pull transmission elevators. The analysis of the Chrysler case shows that the silent-relay architecture produces more opportunities for modularization than the solid-state architecture due to the higher substitutability factor and lower new-to-the-firm component composition. The Schindler case captures the dynamics of modularity created by three types of components (standard, customizable, and new to the firm) and two types of interfaces (fundamental and optional). Based on the case studies, the article outlines testable propositions and discusses the managerial and theoretical implications for the modularization function.

Introduction

There is increasing pressure on firms to shorten the new product development (NPD) lead time and to make a wide selection of customized products available to the customers quickly and cheaply. Many firms are searching for better ways to integrate their NPD capabilities with organizational and supply chain management capabilities. The new dilemma is to gain not only from economies of scale (e.g.,

through standardization of components for mass production) but also from economies of scope (e.g., through customization, incremental innovations, product variations with flexible manufacturing systems), supply flexibility (e.g., from technology leverage through outsourcing), and fast customer responsiveness (e.g., through mass advertising and distribution).

This article addresses issues related to the management of such complexities and focuses especially on management of product architecture modularity. In general, modularity refers to an approach to organize complex products and processes efficiently (Baldwin and Clark, 1997) by decomposing complex tasks into simpler portions to allow the tasks to be managed independently and yet work together as a whole without compromising performance. For instance, the design

Address correspondence to: Juliana H. Mikkola, Copenhagen Business School, Department of Operations Management, Solbjerg Plads 3, DK-2000 Frederiksberg, Denmark. Tel: +45 3815 2441. Fax: +45 3815 2440. E-mail: jh.om@cbs.dk.

*The author is solely responsible for the interpretation of the data and findings of this article.

of an automobile can be decomposed into four levels of complexity: system (e.g., automobile); subsystem (e.g., powertrain, instrument panel); module (e.g., engine, power, rotating blocks of the powertrain), and component (e.g., gear box within the rotating block of the power train). How a firm decides to decompose its product architectures and related tasks depends on sourcing strategies and the firm's scope of knowledge about the system as a whole (Mikkola, 2003). A motivation behind task decomposition is to gain flexibility and cost savings through economies of scale. Decomposition of a complex system into smaller, more manageable parts has been well covered in literature. For example, Taylor (1967) used scientific management principles with respect to standardized work designs and specialization of labor; Milgrom and Roberts (1990) discussed Henry Ford's transfer line technology for mass production; and Simon (1962) reported on his nearly decomposable systems. The effects of modularization have an impact not only on industry standards in the value chain but also on the firm's long-term technology strategy and policy with respect to architectural (Henderson and Clark, 1990) and modular innovations (Christensen and Rosenbloom, 1995). Modular products may protect a firm's market power and architectural control, especially when a firm possesses unique assets or accessibility to complementary assets (Teece, 1986) enabling it to resist the pressure created by customer demands for modular products (Schilling, 2000).

The notion of modularization as a strategy emerged during the 1960s, and many optimization models were introduced to investigate the modularity problem (Evans, 1963; Passy, 1970; Shaftel, 1971) and the modular production concept (Starr, 1965), which described the essence of how to design, develop, and produce parts that can be combined in a maximum number of ways to deal with consumers' demand for

variety and uniqueness. Since then, the literature on modularization has highlighted various aspects of product architecture design and management, such as trade-offs between modular and integral product architectures (MacDuffie, Sethuraman, and Fisher, 1996; Mikkola and Gassmann, 2003; Robertson and Ulrich, 1998; Schilling, 2000; Ulrich and Eppinger, 1995); loosely coupled systems (Orton and Weick, 1990); cost and performance implications (Baldwin and Clark, 1997; Christensen and Rosenbloom, 1995; Langlois and Robertson, 1992; Muffatto, 1999); economies of scale and scope (Friedland, 1994; Pine, 1993); standardization of interfaces (Link and Tassej, 1987; Sanchez, 1999; Tassej, 2000; Ulrich, 1995); substitutability (Garud and Kumaraswamy, 1993, 1995); synergistic specificity (Schilling, 2000; Schilling and Steensma, 2001); mixing and matching (Hsuan, 1999; Sanchez and Mahoney, 1996; Schilling, 2000); and interfirm learning (Mikkola, 2003). A firm's ability to develop and manufacture new products depends largely on its product architecture design strategy. Product configurations and related variations are rooted in the product architecture designs, whereas the way in which components can be disaggregated and recombined into new configurations without losing functionality and performance is based on the degree of modularization embedded in product architectures. The constituent components, which may be standard (STD) or new to the firm (NTF), and how they are linked to one another determine the performance and cost benefits of present and future generations of product architectures. Using STD components minimizes investment, exploits economies of scale from production volume, and preserves organizational focus. NTF components, on the other hand, have the potential to maximize product performance, to minimize the size and mass of a product, and to minimize the variable cost of production (Ulrich and Ellison, 1999). Integration of NTF components into product architectures also prevents imitation by competitors, thus creating competitive advantages for the firm—at least in the short run. But too many NTF components may delay product development lead time and may increase the technological complexity of the product architecture. The degree of product architecture modularity also is affected by the extent to which components can be customized to fit a firm's manufacturing processes.

The present article argues that to understand why some product architectures are more modular, or integral, than others, it is necessary to understand the

BIOGRAPHICAL SKETCH

Dr. Juliana H. Mikkola is assistant professor in the Department of Operations Management at Copenhagen Business School. She received her Ph.D. from Copenhagen Business School and her licentiate from Helsinki School of Economics. She has worked as an automotive design engineer and an executive trainee for Motorola. Dr. Mikkola's research interests include portfolio management of R&D projects, new product development, and modularity strategies in supply chain management. Her research has appeared in *IEEE Transactions on Engineering Management*, *Technovation*, *R&D Management*, *European Journal of Purchasing and Supply Management*, *Global Journal of Flexible Systems Management*, and *Production Planning and Control*.

fundamental relationships shared between components and interfaces. Thus, how can the degree of modularization embedded in product architectures be analyzed systematically? What are some of the trade-offs between STD (to gain from economies of scale) and NTF components (to gain from product performance)? How does component substitutability impact the dynamics of modularization in product architectures? A mathematical model, the *modularization function*, is applied to address these questions. The function considers the following elements: components, interfaces, degree of coupling, and substitutability factor. The article is organized as follows. First, a literature review on modularity and definitions of the key elements of the mathematical model are presented. Second, the modularization function is described and applied with the following sets of product architectures for comparison: (1) Chrysler Jeep's windshield-wiper controllers (silent-relay and solid-state-based technologies); and (2) Schindler elevators (traction-pull and hydraulic-based transmissions). Finally, testable propositions are outlined, followed by a discussion of some managerial and theoretical implications for the modularization function.

Key Elements of Product Architecture Modularity

Product architecture can be described as the arrangement of a product's functional elements into a number of physical building blocks, including the mapping of functional elements into physical components and the specification of interfaces between interacting physical components (Ulrich, 1995). The present article refers to *modularity* as an NPD strategy in which the interfaces shared among the components of a given product architecture become standardized and specified to allow for greater substitutability of components across product families. Product architectures may range from modular to integral arrangements, in which one-to-one mapping between functional and physical product components is nonexistent. In devising a modular product architecture strategy, the goal should be to strike a balance between the gains achieved through recombination (e.g., mixing and matching) of components and the gains achieved through specificity (e.g., higher performance through NTF components) by determining the pressure for or against the decomposition of a system (Schilling, 2000). Although modular components increase flexibility in the product by allowing a

variety of possible configurations to be assembled (Baldwin and Clark, 1997; Garud and Kumaraswamy, 1995), they also increase the coordination effort. Too much product variety from which to choose may actually add to customers' uncertainty. For example, Volkswagen faces an uncertainty regarding order volume and mix; product variety only adds to the obsolescence risks. Consequently, the strategy of limiting variety (e.g., through platform sharing) is actively pursued in the supply chain. Although large volumes are considered as favorable for efficiency, they aggravate the long cycle times and poor service. This is reflected in the Volkswagen Passat's delivery lead time from the factory to a consumer of about one year (van Hoek, 2001). Some contrasting characteristics of modular and integral product architectures are summarized in Table 1.

Many studies on modularization tend to be qualitative and exploratory in nature (cf. Baldwin and Clark, 1997; Garud and Kumaraswamy, 1995; Henderson and Clark, 1990; Muffatto, 1999; Robertson and Ulrich, 1998; Sanchez and Mahoney, 1996). The few quantitative studies on modularity typically apply optimization models to address manufacturing issues (cf. Baker, Magazine, and Nuttle, 1986; Dogramaci, 1979; Emmons and Tedesco, 1971; Evans, 1963; Passy, 1970; Rutenberg and Shaftel, 1971). These models, although sophisticated, are confined to production constraints and offer limited insight and guidance into how firms can measure the degree of modularity embedded in product architectures. One of the challenges faced by practitioners and researchers regarding modularization as a NPD strategy is the difficulty with operationalizing various dimensions into measurable constructs or testable hypothesis. Recently, a few studies have dealt with this issue (cf. Collier, 1982; Fisher, Ramdas, and Ulrich, 1999; Mikkola and Gassmann, 2003; Ulrich and Ellison, 1999; Ulrich and Pearson, 1998; Ulrich et al., 1993). The present article integrates different NPD perspectives on modularization and identifies the following key elements that determine the degree of modularization embedded in product architectures, $M(n_{NTF})$: components, n_{STD} ; n_{NTF} ; N , interfaces, k , degree of coupling, δ , and substitutability, s (Figure 1).

Most of the assembled systems can be decomposed into simpler portions (i.e., subsystems, modules, and components), where each of the components has to be linked with other components within the product architecture in a very specific way—not randomly. This suggests that product architecture designs are idiosyncratic to firms and therefore are difficult

Table 1. Characteristics of Modular and Integral Product Architectures^a

	Modular Product Architecture	Integral Product Architecture
Design Criteria	Commonality sharing	Maximum performance
Component Boundaries	Easy identification	Difficult identification
Redesign to Architecture	Without modification	With modification
Interfaces	Decoupled	Coupled
Outcome	Economies of scale	Craftsmanship
Product Variants	High	Low
Nature of Components	Standardized/generic	Unique/dedicated
Component Outsourcing	Easy	Difficult
Learning	Localized/Dispersed	Interactive
Synergistic Specificity	Low	High
Component Substitutability	High	Low
Component Recombinability	High	Low
Component Separability	High	Low
Nature of Innovation	Autonomous	Systemic
System Design Strategy	Decomposition	Integration

^a Adapted from Baldwin and Clark (1997), Chesbrough and Kusunoki (2001), Christensen and Rosenbloom (1995), Fine (1998), Garud and Kumaraswamy (1993, 1995), Henderson and Clark (1990), Hsuan (1999), Langlois and Robertson (1992), Muffatto (1999), Orton and Weick (1990), Pine (1993), Sanchez and Mahoney (1996), Sanderson and Uzumeri (1997), Schilling (2000), Tassej (2000), Ulrich (1995), Ulrich and Eppinger (1995).

for competitors to imitate. The basic units of analysis of product architecture modularity are simply the components and respective interfaces. Whereas STD components capture mixing-and-matching possibilities, cost advantages, and time-to-market dimensions, NTF components capture performance and the outsourcing strategy dimensions of product architecture designs. Furthermore, the extent to which components can be customized to fit a firm’s manufacturing processes also influences the degree of product archi-

ture modularity. However, when interfaces are considered, the degree of coupling—or how tightly coupled the product architecture is—and the substitutability of NTF components also should be considered, both of which have significant implications for economies of substitution, reusability, and commonality sharing.

The main focus of the present study is to show how product architecture modularity can be measured. This is accomplished by applying a mathematical

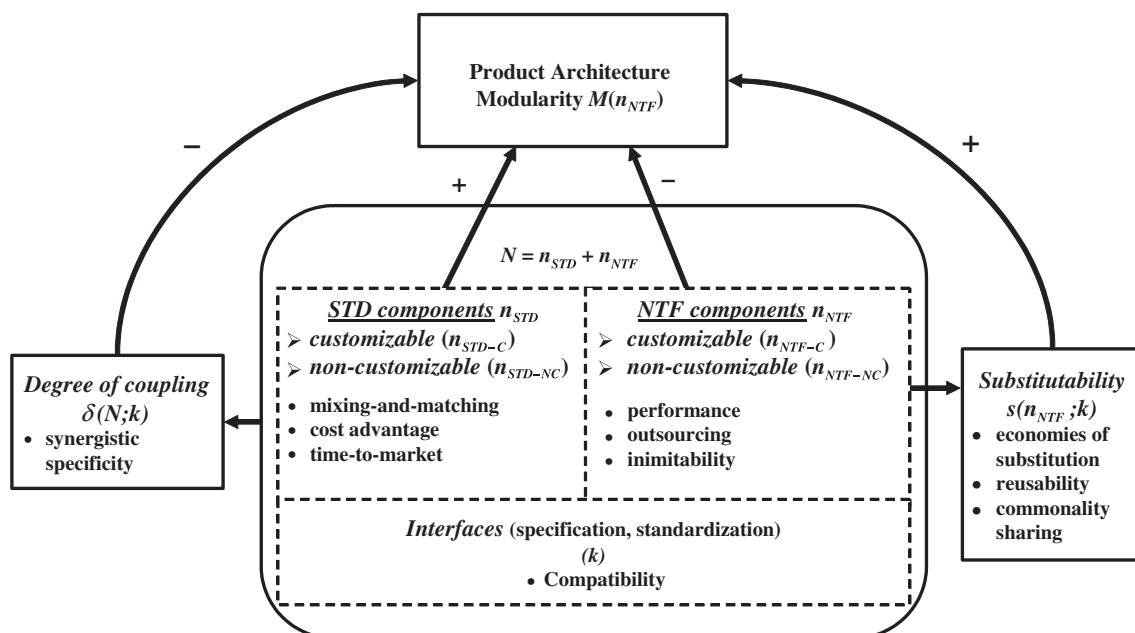


Figure 1. Theoretical Framework of Modularity in Product Architectures

model called the *modularization function*. In mathematical models, analyses often are confined to the limited number of variables allowed by the function. Hence, the variables should be measurable and representative of the phenomena under study. One obvious drawback in this approach is that qualitative aspects—such as the types of interfaces (e.g., mechanical, electrical, environmental), size, and the mass of the components—are not taken into account.

Components

A *component* is defined as a physically distinct portion of a product that embodies a core design concept (Clark, 1985) and performs a well-defined function (Henderson and Clark, 1990). The selection of components reflects a firm's strategic choices. In many firms, components are classified as either standard or new to the firm, depending on whether the firm has known and used these components in former or existing product architectures. Information about the components (e.g., total number of components, component description, component unit costs) is often listed in a bill of materials (BOM). For product architectures serving products in mature markets, such as the automotive and elevator industries, where product variety is a competitive market strategy, component customization adds another dimension to the complexity of product architectures as described in Figure 2. Customizable components allow a firm to create as many product variations as possible. However, product variety forces a manufacturing firm to confront a fundamental trade-off; the increased revenue that may result from more variety versus increased costs through the loss of scale economies (MacDuffie, Sethuraman, and Fisher, 1996). Hence, most firms want to control the amount of product variety offered.

Standard components, n_{STD-C} and n_{STD-NC} . Standard components, n_{STD} , refer to components available in a firm's library of qualified components or components used in a firm's previous or existing architectural designs. Component customization denotes whether the component can be modified to fit a firm's manufacturing processes. Customization of off-the-shelf or generic components can easily be carried out, as the interface specifications of such components are standardized across the industry. For example, a great deal of discrete components (e.g., capacitors, resis-

	Customizable	Non-Customizable
Standard	n_{STD-C} <ul style="list-style-type: none"> • off-the-shelf components • detail-controlled components 	n_{STD-NC} <ul style="list-style-type: none"> • carry-over components • supplier-proprietary components
New-to-the-Firm	n_{NTF-C} <ul style="list-style-type: none"> • new materials • new versions of upgradable components • modular innovations 	n_{NTF-NC} <ul style="list-style-type: none"> • unique components • product-specific components

Figure 2. Classification of Components

tors, transistors) are delivered to the production sites in standardized, uncut lead packages. The components are then cut according to the tolerances allowed by the placement machines and design specifications. Customization of detail-controlled components—that is, parts where the design is controlled by the firm but is manufactured by a supplier—also may take place without incurring significant risks. Some standard components cannot be customized, n_{STD-NC} , such as carried-over components (Ulrich and Ellison, 1999) and supplier-proprietary components. For instance, an engine or a motor used in successive generations of an automobile is generally noncustomizable. Due to previous experience with standard components, interface compatibility issues can be assessed quickly without incurring expensive testing costs. Product architectures comprising standard components are often considered modular product architectures with low synergistic specificity and a high degree of recombability, especially when the number of noncustomizable components is low. Schilling (2000, p. 316) defined synergistic specificity as “the degree to which a system achieves greater functionality by its components being specific to one another” and recombability as the degree to which components can be separated and recombined. According to Ulrich and Ellison (1999), the benefits from selecting an existing component include (1) minimizing investment, or the reuse of existing components avoids significant additional investment in product development and tooling; (2) exploiting economies of scale from production volume; and (3) preserving organizational focus leading to specialization and the development of capabilities.

New-to-the-firm components, n_{NTF-C} and n_{NTF-NC} . NTF components, on the other hand, have recently been introduced to the firm. Because prior knowledge of how NTF components interact with other components is limited, NTF components are assumed to involve higher technological risks than standard components. Interface compatibility issues with other components within the product architecture have to be tested and reevaluated regularly, and sometimes this process can be costly and time consuming. Often, the risks are well justified because normally NTF components improve the overall performance of the system. The components are also difficult to imitate by competitors, thus creating competitive advantages for the firm, at least in the short run. But too many NTF components may delay product development lead time and increase the technological complexity of the product architecture, since a system achieves a greater functionality by the strong interdependence between components (Schilling, 2000). NTF components may be customizable (n_{NTF-C}) or noncustomizable (n_{NTF-NC}). Customizable NTF components (n_{NTF-C}) are the new components that have to be customized for particular applications such as new materials, new versions of upgradeable components, and modular innovations. For instance, molded plastic or stamped sheet-metal parts in coffee makers are custom fabricated by or for the manufacturer (Ulrich and Pearson, 1998). Valves, too, are considered common components in nuclear plant piping systems, and many of them are custom built to respond to specific design and accident scenarios (Farrell and Simpson, 2003). To stay ahead of competition in terms of technological performance, firms often have to design product-specific components from scratch, which cannot be customized (i.e., n_{NTF-NC}). Such components typically challenge the performance of existing technology, either by integrating different technologies into a new component or by significantly improving the performance of the existing component. For instance, the silent relay used in the first generation of Chrysler Grand Cherokee Jeep's windshield-wiper controller was a new-to-the-world component designed specifically for Jeeps.

An NTF component can be designed and manufactured in house, outsourced to suppliers, or code-developed with another firm. The goal is to create a technological performance lead and a certain amount of inimitability or imperfect substitutability (Dierickx and Cool, 1989) with a view to competitors. A gradual introduction of NTF components will keep the product architecture at the desired level of performance

over time. The continuous improvement reflects the firm's ability to renew its competencies and to respond to the rate of technological change anticipated by the industry. A firm's choice about product variety also requires that manufacturing plants can cope with a certain level of product mix complexity (MacDuffie, Sethuraman, and Fisher, 1996). When it is necessary to adopt new sets of core design concepts at the system level driven by the technological, market, and regulatory forces, radical innovations may emerge (Abernathy and Utterback, 1988; Henderson and Clark, 1990; Utterback, 1994).

The distinction between STD and NTF components should be based on the system (e.g., mobile phones, automobiles, elevators) and the firm's manufacturing processes and policies. Whereas some NTF components require, for example, component qualification, manufacturing tooling, and new testing equipment for the production, others do not. For instance, a microprocessor may be interpreted as an STD or an NTF component depending on the changes required in the manufacturing process. If the basic functionality of a new microprocessor simply replaces the old one—that is, the number of leads, lead form, and technical specifications, for example, remain unchanged—the microprocessor is considered to be an STD component. However, if the new microprocessor requires for example a new printed circuit board (PCB) design, new software codes, additional burning, or new tooling, it should be seen as an NTF component.

Interfaces, k

Interfaces are linkages shared among components, modules, and subsystems of product architectures. Interface specifications define the protocol for the fundamental interactions across all components comprising a technological system. The degree to which interfaces become standardized and specified defines the compatibility between components—hence, the degree of modularity. With standardization of interfaces, market-entry barriers for new suppliers are lowered, and component performance can be attained through technological specialization. According to Langlois and Robertson (1995, p. 5), “standardization of interfaces creates ‘external economies of scope’ that substitute in large part for centralized coordination among the wielders of complementary capabilities. This allows the makers of components to concentrate their capabilities narrowly and deeply and thus to improve their piece of the system independently of others.”

STD components have well-specified and standardized interfaces. Conversely, interface specification of NTF components is normally not standardized, because it takes time to determine the components' compatibility and functionality in relation to the rest of the product architecture. Introduction of NTF components also might require the alteration of interface specifications of other components and may even create a ripple effect (Meyer and Lehnerd, 1997) on other subsystems with a devastating result. Consequently, introduction of NTF components into product architectures reduces modularity freedom. Over time, when the technological operation of the NTF component with the rest of the product architecture has become standardized (i.e., the component is qualified and listed in the component library as a standard component), one might expect the product architecture to become more modular. This means that contract arrangements with suppliers and customers are in place; that is, purchasing volumes and prices are set. Production processes are frozen in the sense that alterations to design and assembly processes, such as changing automation technology and tooling, cannot be made without going through official engineering change request procedures.

Degree of Coupling, δ

Degree of coupling can be treated as a proxy for estimating the degree of tightness shared among the components. The relative criticalness of components in the architecture also is identified. The way in which components are linked with one another creates a certain degree of coupling. Critical components, which depend on many other components—like many interfaces—for functionality, imply a high degree of coupling. Microprocessors, for instance, are critical components since they have to interface directly with a number of components, easily ranging from 56 to over 200 interfaces. It is not a stretch, then, to imagine that product architectures with a great percentage of critical components may not easily be decomposed. In other words, product architectures with a high degree of coupling among the components exhibit a high synergistic specificity, because the strong interdependence among components inhibits recombination, separability, and substitution of components, hence preventing the architecture from becoming more modular (Schilling, 2000).

Depending on the product architecture configuration, often decided by the engineers, the combined ef-

fect of components and interfaces dictates the degree of coupling of the product architecture. In tightly coupled product architectures, it should be expected that many critical components will share complex interface relationships with other components. On the other hand, product architectures with a low degree of coupling include components that are relatively independent of each other. It might be possible to encapsulate the functions of particular components and to employ a standard interface between them that enables contact with little or no loss of performance (Garud and Kumaraswamy, 1995; Sanchez and Mahoney, 1996; Schilling and Steensma, 2001). If an analysis were conducted of similar systems produced by different companies—for example, Parnasonic versus Sony televisions—in terms of their components and respective interfaces, it would most likely reveal that the product architectures have their own configurations and degrees of coupling. Some product architectures have a large number of components but few interfaces, whereas others have fewer components but require more interfaces to meet functionality requirements.

Substitutability, s

Substitutability is another crucial element of product architecture. According to Dogramaci (1979), substitutability in product design decisions has been studied extensively by industrial engineering scholars under terms such as the *modular production concept* (Starr, 1965), the *commonality problem* (Baker, Magazine, and Nuttle, 1986; Collier, 1981, 1982; Dogramaci, 1979; Moscato, 1976; Rutenberg, 1971; Rutenberg and Shaftel, 1971); and the *assortment problem* (Sadowski, 1959; Wolfson, 1965). The strategy literature, on the other hand, sees substitutability in a broader perspective, refraining from sophisticated mathematical techniques. For instance, Garud and Kumaraswamy (1995) used the term *substitution* to suggest that technological progress may be achieved by substituting certain components of a technological system while reusing others, hence taking advantages of economies of substitution. This has great implications for technological systems that are modularly upgradeable. With economies of substitution, firms may reduce product development time, may leverage past investment, and may provide customers with continuity. To capture the essence of economies of substitution, they identified three system level attributes: integrity, modularity, and upgradeability. Economies of substitution exist when the cost of designing a high-performance system through the

partial retention of existing components is lower than the cost of designing the system afresh (Garud and Kumaraswamy, 1993). Furthermore, Sanchez (1999) suggested that reusability of common components within and across product lines reduces costs by reusing existing component designs, by lowering costs through learning curve effects, by increasing scale of component production, by increasing buyer power for common components, by reducing component variety and inventories, and by reducing costs of product support.

Another aspect of substitutability is component sharing, or using the same component version across multiple products, which is a product-based strategy based on the fact that families of similar products have similar components (Fisher, Ramdas, and Ulrich, 1999). Many firms view component sharing as a way to offer a high variety in the market place while retaining a low variety in their operations. Component sharing of NTF components is especially critical. Fisher, Ramdas, and Ulrich (1999, p. 299) argued that “because each new and unique component must be designed and tested, component sharing can reduce the cost of product development. Each new and unique component generally also requires an investment in tooling or other fixed costs of production. Therefore component sharing may also reduce the required production investment associated with a new product.” The managerial challenge is how to provide the high degree of uniqueness that seems necessary for competitive success while retaining the scale economies required by low cost. Firms generally do not introduce radical product designs to the market all the time. Incremental product designs are more often observed. It is easy to see how a firm would save costs by using STD components in product architecture designs compared to NTF components. If a firm plans to invest time and effort to incorporate NTF components into the product design, the components should provide superior performance and value for the firm, especially if they can be shared across product families. There is a focus on substitutability of NTF components from a product architecture modularity perspective, especially if such components are designed for reusability and commonality sharing across product families.

Modularization Function

Modularization function (equation 1) is used to investigate the degree of modularization embedded in product architectures and trade-offs imposed by

components, N ; n_{STD} ; n_{NTF} , interfaces, k , degree of coupling, δ , and the substitutability factor of NTF components, s . See Appendix A for its formulation.

$$M(n_{NTF}) = e^{-n_{NTF}^2/2Ns\delta}, \quad (1)$$

where $M(n_{NTF})$ is the modularization function, n_{NTF} is the number of NTF components, N is the total number of components, s is the substitutability factor, and δ is the degree of coupling.

The modularization function is interpreted as follows. A given product architecture has N components that are the sum of STD and NTF components ($N = n_{STD-C} + n_{STD-NC} + n_{NTF-C} + n_{NTF-NC}$). The specific ways components are linked through interfaces, k , create a certain degree of coupling, δ , which is approximated as the average number of interfaces per component. The impact of the substitutability of NTF components in the product architecture modularity is captured by the substitutability factor, s , estimated as the total number of families in which the NTF components are used divided by the average number of interfaces required of functionality, k_{NTF} . Due to interface compatibility uncertainties imposed by NTF components, the lower the number of NTF components the higher the degree of modularization. Hence, a perfect-modular product architecture, $M(n_{NTF}) = 1.0$, does not have any NTF components. NTF components used across product families have a higher substitutability factor—thus benefiting from economies of substitution, reusability, and commonality sharing—than NTF components dedicated to one specific product family. This increases the degree of modularity. The modularization function shows that the combined effect of variables varies exponentially according to the set of NTF components. Every time the composition of NTF components is altered—such as with modular innovations—the degree of modularity also changes. In many cases, the introduction of NTF components requires changes in other parts of the product architecture, leading to the changing the values of N and δ . If the degree of modularity was simply assessed based on the number of components, regardless of whether they are STD or NTF, and ignored the effects of interfaces captured in δ and s , it may not be evident as to why some systems are more modular than others.

Empirically, the modularization function has been tested in Chrysler Jeep’s windshield-wiper controllers (WIPER) and Schindler elevators for comparative analysis. The analysis of the WIPER case was possible because the present author, as the main design engineer, had full access to the engineering data.

Because the data needed for validating the modularization function includes a certain amount of proprietary data—such as schematics, BOMs, NPD, and procurement strategies for NTF components—many firms were reluctant to participate in the study; Schindler Lifts of Switzerland was an exception. In the following section, the two studies are described. Although automobiles and elevators are different systems, it is possible to make a systematic analysis of dissimilar systems when the complexity of the modularity imposed by components and respective interfaces is expressed mathematically. The assessment of both systems involved the following steps:

- (1) Define product architecture and its boundaries.
- (2) Decompose the product architecture into desired levels of analyses, such as subsystem or module, to enable each level of decomposition to be assessed independently. For instance, a system might be decomposed into two subsystems where each subsystem again is decomposed into five and three modules, respectively.
- (3) Count the total number of components included in the product architecture, N . This can be accomplished by looking at the product's BOM.
- (4) Count the number of NTF components, $n_{NTF} = n_{NTF-C} + n_{NTF-NC}$.
- (5) Compute the degree of coupling, δ . There is a δ associated with each level of analysis: for example, δ_{system} , $\delta_{sub-system}$, δ_{module} . Determine the number of subcircuits comprising each level of analysis. For each subcircuit, count the number of interfaces, k , and components, n . Compute the degree of coupling for each subcircuit, $\delta_{sub-circuit}$: divide the sum of k into the sum of n , $\Sigma k / \Sigma n$. Compute the degree of coupling of the subcircuit: the average of the $\delta_{sub-circuit}$. For instance, a subsystem Y might have three subcircuits A, B, and C; hence, $\delta_Y = (\delta_A + \delta_B + \delta_C) / 3$. Repeat the algorithm for other levels of analyses. To calculate δ of the product architecture, add the corresponding δ values for each level of analysis, $\delta = \delta_{system} + \delta_{sub-system} + \delta_{module}$.
- (6) Compute the substitutability factor of the NTF components, s . Count total number of product families in which the NTF components are used; then, divide this number by the average number of interfaces required for functionality, k_{NTF} .
- (7) Plug these values into the modularization function (equation 1) to compute the degree of modularization embedded in the product architecture.

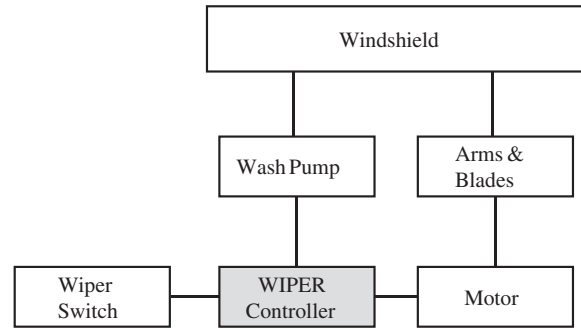


Figure 3. Block Diagram of Jeep's Windshield Wiper System

Case A: Chrysler Jeep's Wiper Controller

Jeep Grand Cherokee was introduced in 1993 as a high-end utility vehicle. A great deal of innovation and design concepts was incorporated into the new vehicle, including the windshield-wiper controller module (WIPER), which is a black-box module outsourced—for both NPD and manufacturing—to a Fortune 100 original equipment manufacturer (OEM) supplier. WIPER's block diagram is illustrated in Figure 3. Most researchers would exclude the windshield from being part of the wiper system. However, in the first generation of Jeep Grand Cherokees, the windshield was a crucial component of the system because the WIPER's functionality and performance could not have been determined without it. Primary data—the author's personal involvement in NPD, manufacturing, and sourcing—as well as secondary data—schematic drawings, BOMs, engineering log books, and other proprietary engineering data—were examined. The data collection took place between 1991 and 1993, from the start of the development date to the full production date. There were two different potential technological solutions to the design of the module: solid state and silent relay. The WIPER module used by former Jeep families applied a standard relay-based technology that made clicking noises when switching from on to off, a feature Chrysler wanted to eliminate. During the first design attempt, a solid-state approach was applied using only transistors and electrical components.

After almost a year of development, the solid-state concept failed due to insufficient knowledge about the interface constraints between the WIPER and the rest of the windshield wiper system. Since Jeep Grand Cherokee was a new family of vehicles incorporating many new technologies, not all interface compatibility issues among the components were well understood. For instance, the slight change in

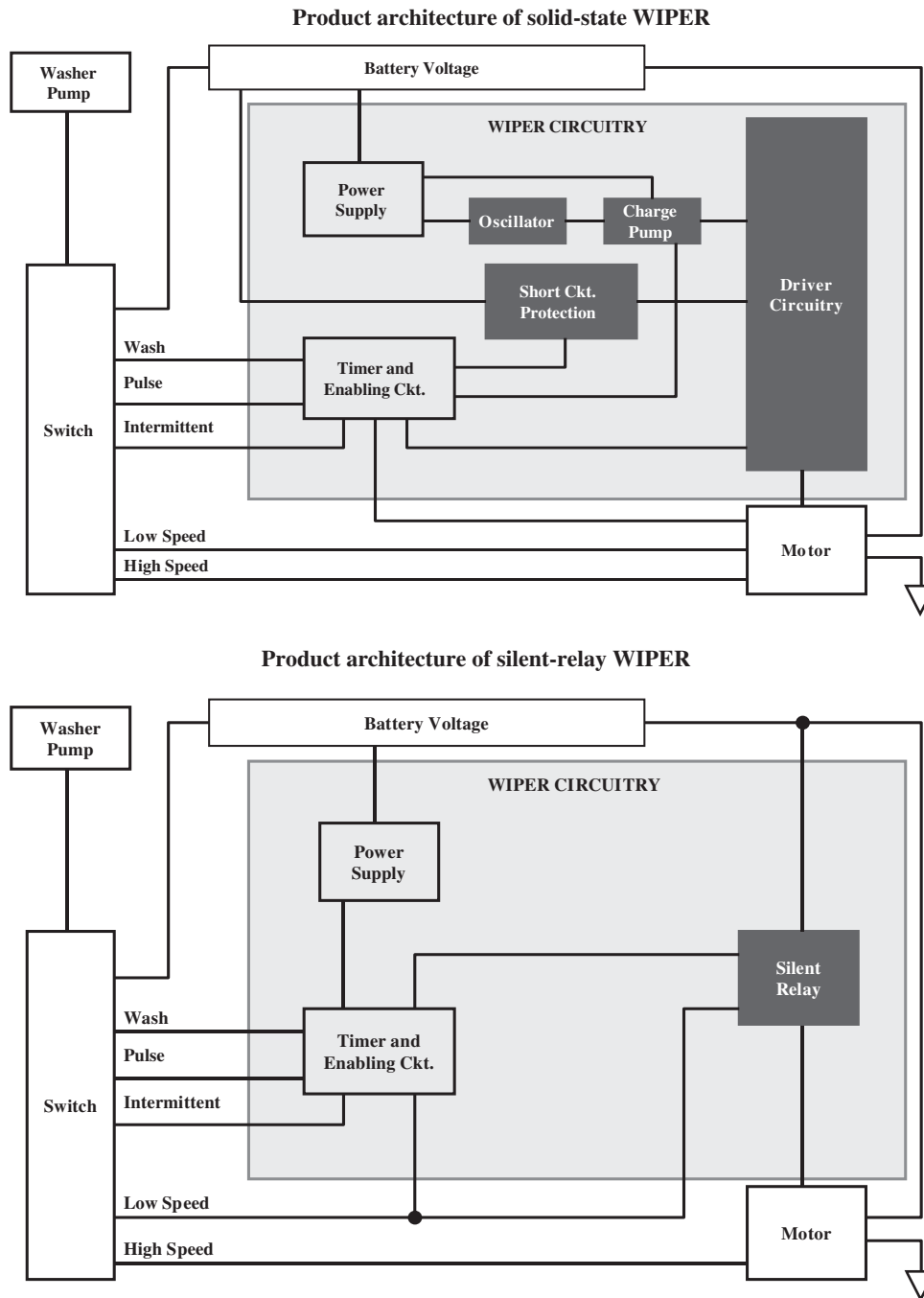


Figure 4. Product Architectures of the WIPERs

the windshield angle posed critical constraints on the entire windshield wiper system. Under special conditions, the WIPER would overheat and lose functionality. During the second attempt, a totally new innovation was developed to create the silent-relay WIPER. In an effort to minimize design and manufacturing changes, the silent-relay and peripheral circuits replaced a portion of the solid-state WIPER. Although the changes were not drastic, the relation-

ships between the components and the respective sub-circuits and interfaces were altered. A comparison between the silent-relay and the solid-state WIPERs is shown in Figure 4.

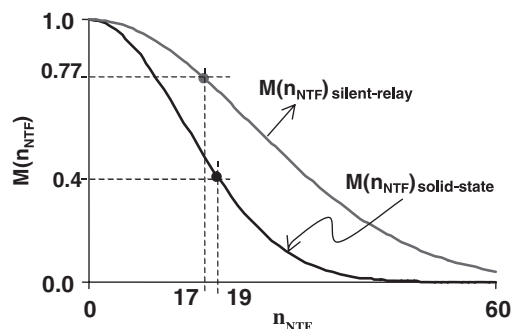
The WIPER module requires three immediate linkages for functionality: wiper switch, wash pump, and motor. Whereas the solid-state WIPER is only compatible with Grand Cherokee Jeeps (substitutability factor, $s = 1/3 = 0.33$), all three families of Jeeps—Grand

Table 2. Values for Modularization Functions for the Solid-State and Silent-Relay WIPERs

Solid-State WIPER	Silent-Relay WIPER
$n_{NTF} = 19$ components ($n_{NTF-C} = 18$; $n_{NTF-NC} = 1$) $N = 60$ components	$n_{NTF} = 17$ components ($n_{NTF-C} = 15$; $n_{NTF-NC} = 2$) $N = 57$ components
$s = 0.33$ components/interface	$s = 1.00$ components/interface
$\delta = 9.85$ interfaces/component	$\delta = 9.94$ interfaces/component
$b = 31.7\%$	$b = 29.8\%$
$M_{solid-state} = 0.40$	$M_{silent-relay} = 0.77$

Cherokee, Cherokee, and Wrangler—can use the silent-relay WIPER ($s = 3/3 = 1$). The solid-state WIPER has 60 components ($N = 60$), out of which 19 are NTF components ($n_{NTF} = 19$), yielding a NTF component ratio b of 0.317 ($b = 19/60 = 0.317$). Similarly, the silent-relay WIPER has 57 components, out of which 17 are NTF components translating to a value of 0.298 for b . The values for the modularization functions are shown in Table 2. Graphically, the modularization functions of both WIPERs are shown in Figure 5.

The silent-relay WIPER has a higher degree of modularization ($M_{silent-relay} = 0.77$) than the solid-state WIPER ($M_{solid-state} = 0.4$). Given the relatively similar values of interface constraints ($\delta_{solid-state} = 9.85$; $\delta_{silent-relay} = 9.94$), the silent-relay WIPER was more modular due to its higher substitutability factor and lower NTF component composition. Notice how the modularization gap increases as the number of NTF components increases. This suggests that product architectures can achieve higher levels of modularity by reducing the number of NTF components. Similarly, modularity can be improved by designing product architectures with a higher substitutability factor, if the NTF component composition remains constant.

**Figure 5. Modularization Functions for Solid-State and Silent-Relay WIPERs**

Case B: Schindler Elevators

According to elevator experts, the product architecture of elevators has been stable over a long period due to regulations and few innovations. The number of competitors has decreased dramatically during the last 15 years. Over 80% of the world market share belongs to seven global players. Modularization through standardization of interface specifications has enabled small elevator companies to source from standard component manufacturers, therefore benefiting from economies of scale despite their small market share. Based on the transmission principle, dominant elevator designs, which account for over 90% of market, are distinguished between (1) the traction-pull (TR) design with drive machine, ropes, and counterweight; and (2) the hydraulic (HY) design with a hydraulic jack.

The project, carried out in three phases, took place between 1997 and 2000. In phase 1, the details of TR and HY elevators were mapped and recorded by Schindler Lifts. Based on the unified modeling language (UML) model, an object modeling technique originally developed for supporting object oriented software development, several hundred components with respective interfaces were documented for TR and HY elevators. Phase 2 involved interviews with elevator experts from Schindler's research and development (R&D), system management, purchasing, and marketing organizations. In phase 3, the modularization function was applied to analyze the degree of modularization embedded in TR and HY elevators.

Figure 6 shows a partial product architecture of TR elevators. The classification of components into *unique*, *neutral*, and *standard* was carried out by an interdisciplinary group consisting of R&D, purchasing, and marketing experts from Schindler. *Unique* represents n_{NTF-NC} components; *standard* represents n_{STD-NC} components; and *neutral* are customizable components that may be n_{STD-C} or n_{NTF-C} depending on the required customization and application. The linkage between the components is characterized as *fundamental* and *optional*. Though fundamental linkages exist for all elevator variants, optional linkages are only relevant for certain variants. To illustrate how the modularization function can be applied, the transmission subsystems of both HY and TR elevators were selected for comparative analysis. The analysis of each elevator system was carried out at two levels: the subsystem level, or the transmission, and the system level, or the elevator, as shown in Figure 7.

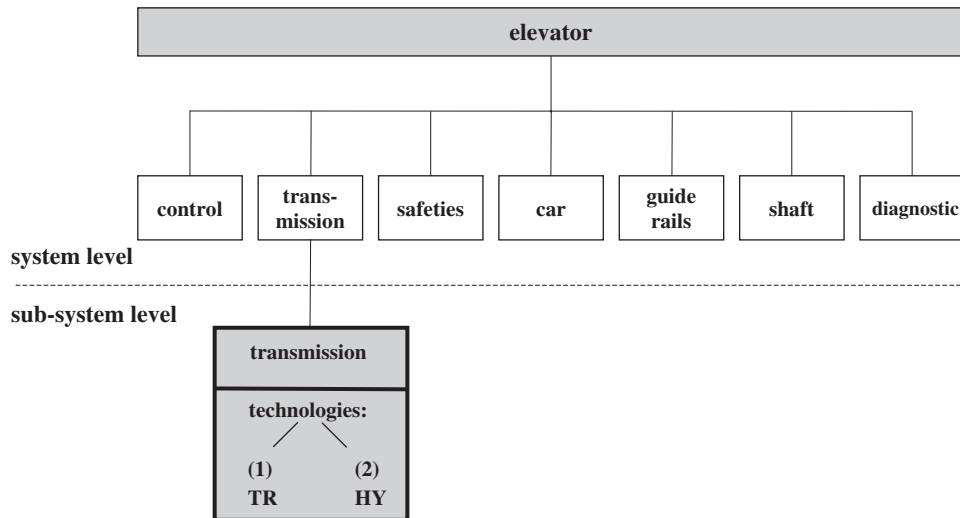


Figure 7. The Elevator and Its Subsystems

(3) Neutral components represent a set of customizable components that can be either standard (n_{STD-C}) or NTF (n_{NTF-C}). The total number of customizable components is the sum of n_{STD-C} and n_{NTF-C} , neutral = $n_{STD-C} + n_{NTF-C}$. Since elevators have to be built on site with components provided by various suppliers—especially components procured under the multisourcing strategy—it is assumed that customization of components varies from one geographical area to another; hence, the component mix of n_{NTF-C} and n_{STD-C} varies, too. This assumption allows for an analysis of the worst-case scenario imposed by NTF components: when $n_{STD-C} \rightarrow n_{NTF-C}$. In this case, all the components require extensive customization, which may increase the technological complexity of the elevator and may delay product development and manufacturing lead time. Such a situation may complicate the implementation of the modularization considerably.

Since both elevators have fundamental and optional linkages as well as three classifications of components, the basic evaluation only focuses on components linked by fundamental interfaces. The maximum relationship between the components and respective linkages is achieved when the remaining components with optional linkages are added to the product architecture. This generates different sets of degrees of coupling, δ , substitutability factors, s , unique component compositions, b , and the total number of components, N , in the analysis. Therefore,

there may be a range of modularity levels for the two elevators. $M_{fundamental}(n_{NTF})$ and $M(n_{NTF})$ represent the basic and the maximum degree of modularity, respectively. A comparative analysis of HY and TR elevators is summarized in Table 3. The modularization functions for HY and TR elevators are illustrated in Figure 8.

In the following, some findings on HY and TR elevators are presented.

- (1) Both elevators are highly modular from a n_{NTF-NC} component composition perspective, $M_{HY}(3) = 0.98$ and $M_{TR}(6) = 0.87$.
- (2) HY elevators are more modular than TR elevators due to a higher substitutability factor ($s = 1.2$), a lower unique component composition ($b = 7\%$), and a smaller average number of interfaces shared per component ($\delta = 4.59$). Graphically, the higher modularity of HY elevators is indicated by the relative slopes where $M_{TR}(n_{NTF})$ is much steeper than $M_{HY}(n_{NTF})$.
- (3) When all neutral components are treated as NTF components (neutral = n_{NTF-C}), TR elevators provide more opportunities for improving the degree of modularization. For instance, the degree of modularization of TR elevators, $M_{TR}(n_{NTF})$, may range from 0.08 to 0.87 ($n_{NTF-NC} = 6$; $n_{NTF-C} = 19$) compared to the degree of modularization of HY elevators, $M_{HY}(n_{NTF})$, which ranges from 0.47 to 0.98 ($n_{NTF-NC} = 3$; $n_{NTF-C} = 16$).
- (4) The modularity of both TR and HY elevators can be increased by increasing the number of families or models of elevators that can use the NTF

Table 3. A Comparison of HY and TR Elevators

HY ELEVATORS		TR ELEVATORS	
2 families (low-rise, mid-rise) $n_{NTF-NC} = 3$ components $n_{STD-C} \leq 16$ neutral components $\leq n_{NTF-C}$		3 families (low-rise, mid-rise, high-rise) $n_{NTF-NC} = 6$ components $n_{STD-C} \leq 19$ neutral components $\leq n_{NTF-C}$	
Fundamental Linkages N = 37 components $b = n_{NTF}/N = 8\%$ $s = 1.2$ components/interface $\delta = 4.02$ interfaces/component $M_{fundamental}(n_{NTF}) = 0.98$ $M(n_{NTF})_{unique + neutral} = 0.36$		Fundamental Linkages N = 38 components $b = n_{NTF}/N = 16\%$ $s = 0.64$ components/interface $\delta = 4.83$ interfaces/component $M_{fundamental}(n_{NTF}) = 0.86$ $M(n_{NTF})_{unique + neutral} = 0.07$	
			All Linkages N = 43 components $b = n_{NTF}/N = 7\%$ $s = 1.2$ components/interface $\delta = 4.59$ interfaces/component $M(n_{NTF}) = 0.98$ $M(n_{NTF})_{unique + neutral} = 0.47$
			All Linkages N = 42 components $b = n_{NTF}/N = 14\%$ $s = 0.60$ components/interface $\delta = 5.01$ interfaces/component $M(n_{NTF}) = 0.87$ $M(n_{NTF})_{unique + neutral} = 0.08$

components—that is, by increasing the substitutability factor, s .

- (5) From the interface perspective, optional linkages also influence the degree of modularization. When all of the interfaces are taken into consideration—both fundamental and optional—the opportunities for modularization are better in HY elevators than in TR elevators. This is indicated by the

larger differences between the modularization functions $M(n_{NTF})$ and $M_{fundamental}(n_{NTF})$.

Theoretical and Managerial Implications

Although the application of the modularization function to two sets of product architectures (WIPERs and elevators) has provided very preliminary findings on how product architectures' degree of modularity can be assessed, it can nevertheless be used as a powerful managerial tool. Similar systems produced by different companies will undoubtedly have different product architecture designs due to the companies' different design and technology choices. This suggests that the composition of components is idiosyncratic to a particular product architecture design. Product configurations and their related variations are rooted in the product architecture designs, whereas the way in which components can be disaggregated and recombined into new configurations without losing functionality and performance is based on the level of modularization in product architectures. The constituent components—which may be either customizable or noncustomizable NTF or standard—as well as how they are linked to one another determine the performance and cost benefits of product architectures. Component selection embedded in product architecture designs reflects companies' different strategic choices. Some general observations can be drawn from the case studies.

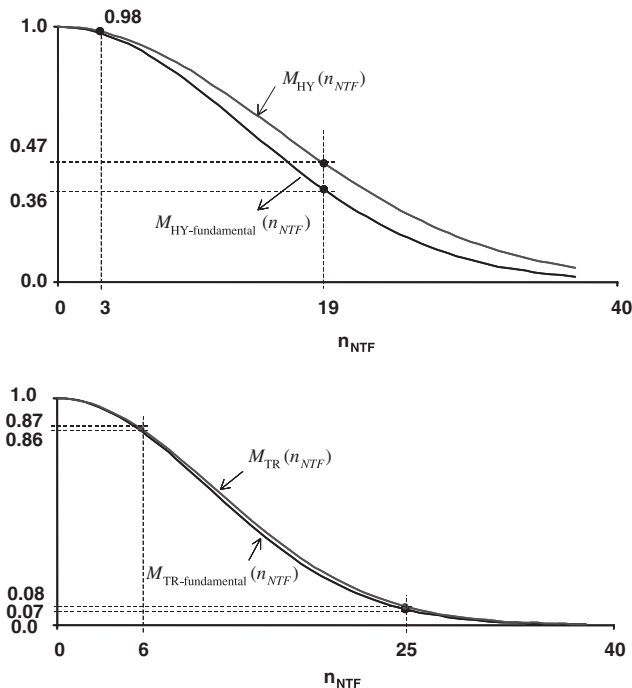


Figure 8. Modularization Functions of TR and HY Elevators

- (1) Movement along the curve captures the dynamics of NTF components. Over time, when interfaces of NTF components have become well specified and standardized—assuming no changes are made to the product architecture—NTF components will turn into STD components. Under such circumstances, an increased degree of modularity should be observed.
- (2) A shift to a new curve indicates changes made in the product architecture in terms of NTF components, either through introduction of new innovations, such as modular innovations, or through major architectural changes.
- (3) Mature product architectures, such as elevators, have a good grip on interrelations among components; thus, fewer opportunities exist for modularization. This is indicated by the gap between the modularization function curves compared to those of the WIPER functions.

For practitioners, a measurement model of modularization is valuable, as it highlights various managerial and strategic implications of architecture design decisions. These decisions are often based on the firm's vision influenced by strategic managers' knowledge and expertise on the technological development in the industry. When the fundamental relationship between components and the respective interfaces of a given product architecture is understood in a systematic manner through some sort of framework or model, it facilitates decision making with regard to potential cost and benefit implications for the future generations of product architectures. It also enhances knowledge sharing and facilitates consensus making between engineering and management.

From a strategic perspective, a measurement model can help managers handle complexity embedded in product architecture designs better and understand and foresee what the impact of system decomposition into simpler portions or integration of standardized components into a new innovation will be on the degree of modularity in future generations of product architectures. According to Clark and Fujimoto (1991), decisions about innovation and variety affect product complexity, and the degree of supplier involvement in addition to the use of off-the-shelf parts affects the volume of engineering work performed in house. Together, these choices determine the complexity of the project, which in turn influences productivity, lead time, and total product quality. The systematic analy-

sis of product architecture modularity in terms of components and interfaces and translating the analysis into a graphical format provide a common language for both engineers and other members of the firm. It also may be used as a tool for performing scenario analysis and for analyzing competitors' product architectures through reverse engineering. Often, such a process reveals the potential future technological innovations pursued by the competitor in addition to the competitor's current product architecture strategies and manufacturing capabilities, such as miniaturization and component integration or decomposition.

Discussion and Future Research

The modularization function may provide a good theory for studying complexity embedded in product architectures, although its formulation has not statistically been tested or proven. But other methods and approaches may aid in strengthening the robustness of the model. Statistical methodology is one complementary method for unraveling the correlations of variables in the modularization function to the degree of modularity in the product architecture, provided there are sufficient empirical observations. This would further validate the general assumptions on product architecture modularity and the robustness of the model. Consequently, one way to study the modularization associated with product architectures statistically is to test propositions based on the variables of the modularization function with a sufficient number of product architectures.

Depending on the product architecture configuration, the combined effect of components and interfaces dictates the degree of coupling, δ , of the product architecture. This explains to some extent why virtually no two assembled products in the market are exactly the same, even when they compete in the same product category (e.g., Sony Walkman versus Aiwa personal portable stereos; Panasonic versus Philips high-definition televisions). A component with tight coupling indicates a component requiring many interfaces per component for functionality. A given product architecture comprised of many tightly coupled components will exhibit a high degree of coupling. For instance, the product architectures of solid-state and silent-relay WIPERS portray very similar values of δ : 9.85 and 9.94, respectively. The degree of coupling for Schindler elevator's HY and TR product architectures, on the other hand, are more interesting. In general, the degree of coupling values for HY elevators is lower than for TR elevators. This indicates

that if the effects of NTF components and the substitutability factor are ignored, the components of TR elevators are more tightly coupled than the components of HY elevators. In other words, if both elevators had the same component composition, b , and the same substitutability factor, s , and only differed in the value of δ , TR elevators would be less modular than the HY elevators. Hence, the following is suggested.

Proposition 1: The degree of coupling has a negative effect on the degree of modularization embedded in product architectures.

Architectural development and renewal take time and money, and upgrades generally change incrementally from one generation to the next. It is possible to make upgrades by recombining standard components and by adding to and substituting NTF components in the existing product architecture. To obtain economies of scale from NTF components, the components should be developed with cross-family substitutability, since one of the design criterion is to gain from economies of substitution (Garud and Kumaraswamy, 1995). Whereas standard components facilitate component reusability, NTF components improve the technological performance of the upgraded product architecture. Component sharing of NTF components is especially critical as the development in terms of time and money of NTF components might be overwhelming if not managed properly. Theoretically speaking, to improve the value of NTF components they should be shared as much as possible across product families, which will increase the value of the substitutability factor and therefore will increase the degree of modularity of the product architecture. As the WIPER product architectures show, one of the main factors that made the silent-relay WIPER more modular [$M(n_{NTF})_{solid\ state} = 0.4$ and $M(n_{NTF})_{silent\ relay} = 0.77$] is attributed to its higher substitutability factor [$s_{solid\ state} = 0.33$ and $s_{silent\ relay} = 1.0$]. The effect of the substitutability factor on the overall product architecture modularity can also be observed in Schindler elevators. In the worst case, when all components and interfaces are considered, the HY elevators have a higher substitutability factor ($s = 1.2$) compared with TR elevators ($s = 0.6$), which can be translated into a higher degree of modularization [$M(n_{NTF})_{HY} = 0.98$ and $M(n_{NTF})_{TR} = 0.87$]. Hence, the following is proposed.

Proposition 2: The substitutability factor of new-to-the-firm components has a positive effect on the degree of modularization embedded in product architectures.

So far, the present article has shown how the modularization function can be used as a tool to analyze in theory how components, interfaces, the degree of coupling, and the substitutability factor influence product architecture modularity. The modularization function also has its limitations. For instance, it only considers the basic elements of product architecture: components and respective interfaces. Because all the data on components and interfaces are gathered from schematic drawings and BOMs, the modularization function does not differentiate between types of interfaces and related types of modules. Also, the function does not consider the trade-offs between cost and performance of components. It simply assumes that performance advantage is gained from NTF components and that cost savings are gained from standard components. In reality, there are expensive firm-specific standard components (e.g., microprocessors, engine controllers) and inexpensive industry-specific components (e.g., transistors, DRAMS, plugs). But because they are not incorporated into the firm's database, they are treated as NTF components by the modularization function. In many cases, the qualification process of these components is pretty straightforward, especially with off-the-shelf parts.

In addition to cost and benefit implications of product architecture designs, organizational design and supplier–buyer interdependence issues are equally crucial. It has been argued that outsourcing of noncore technical activities is enabled by the standardization of the noncore components with respect to the core technology (Mikkola, 2003). Can decisions on product architecture designs provide insight into strategic decisions on outsourcing, manufacturing, and supply chain management? If so, how should a firm design its organization to match such strategies with respect to its suppliers and customers? Other areas of research interest that may be examined through the lenses of the modularization function include the impact of product architecture design choices—such as multiplexing and deintegration of component—on postponement and mass customization strategies (Farrell and Simpson, 2003; Pine, 1993).

References

- Abernathy, William J. and Utterback, James M. (1988). Patterns of Industrial Innovation. In: *Readings in the Management of Innovation*. Michael L. Tushman and William L. Moore (eds.). Ballinger Publishing Company, 25–36.

- Baker, Kenneth R., Magazine, Michael J. and Nuttle, Henry L.W. (1986). The Effect of Commonality on Safety Stock in a Simple Inventory Model. *Management Science* 32(8):982–8.
- Baldwin, Carliss Y. and Clark, Kim B. (1997). Managing in an Age of Modularity. *Harvard Business Review* 75(5):84–93 (September–October).
- Chesbrough, Henry W. and Kusunoki, Ken (2001). The Modularity Trap: Innovation, Technology Phase Shifts, and the Resulting Limits of Virtual Organizations. In: *Managing Industrial Knowledge: Creation, Transfer, and Utilization*. Ikujiro Nonaka and David J. Teece (eds.). London, UK: Sage, 202–30.
- Christensen, Clayton M. and Rosenbloom, Richard S. (1995). Explaining the Attacker's Advantage: Technological Paradigms, Organizational Dynamics, and the Value Network. *Research Policy* 24(2): 233–57.
- Clark, Kim B. (1985). The Interaction of Design Hierarchies and Market Concepts in Technological Evolution. *Research Policy* 14(5):235–51.
- Clark, Kim B. and Fujimoto, Takashiro (1991). *Product Development Performance*. Boston, MA: Harvard Business School Press.
- Collier, David A. (1981). The Measurement and Operating Benefits of Component Part Commonality. *Decision Sciences* 12(1):85.
- Collier, David A. (1982). Aggregate Safety Stock Levels and Component Part Commonality. *Management Science* 28(11):1296–303.
- Cusumano, Michael A. and Nobeoka, Kentaro (1992). Strategy, Structure, and Performance in Product Development: Observations from the Auto Industry. *Research Policy* 21(3):265–93.
- Dierickx, Ingemar and Cool, Karel (1989). Asset Stock Accumulation and Sustainability of Competitive Advantage. *Management Science* 35(12):1504–11.
- Dogramaci, A. (1979). Design of Common Components Considering Implications of Inventory Costs and Forecasting. *AIIE Transactions* 11(2):129–35.
- Emmons, Hamilton and Tedesco, Alan R. (1971). The Modular Growth Design Problem. *AIIE Transactions* 3(2):104–14.
- Evans, David H. (1963). Modular Design—A Special Case in Nonlinear Programming. *Operations Research* 11(4):637–47.
- Farrell, Ronald S. and Simpson, Timothy W. (2003). Product Platform Design to Improve Commonality in Custom Products. *Journal of Intelligent Manufacturing* 14(6):541–56.
- Feitzinger, Edward and Lee, Hau L. (1997). Mass Customization at Hewlett-Packard: The Power of Postponement. *Harvard Business Review* 75(1):116–21 (January–February).
- Fine, Charles H. (1998). *Clockspeed—Winning Industry Control in the Age of Temporary Advantage*. Reading, MA: Perseus Books.
- Fisher, Marshall, Ramdas, Kamalini and Ulrich, Karl (1999). Component Sharing in the Management of Product Variety: A Study of Automotive Braking Systems. *Management Science* 45(3):297–315.
- Friedland, Jonathan (1994). Car Industry: Mini Miracle. *Far Eastern Economic Review* 157(17):76.
- Garud, Raghu and Kumaraswamy, Arun (1993). Changing Competitive Dynamics in Network Industries: An Exploration of Sun Microsystems's Open Systems Strategy. *Strategic Management Journal* 14(5):351–69.
- Garud, Raghu and Kumaraswamy, Arun (1995). Technological and Organizational Designs for Realizing Economies of Substitution. *Strategic Management Journal* 16(special issue):93–109.
- Henderson, Rebecca M. and Clark, Kim B. (1990). Architectural Innovation: The Reconfiguration of Existing Product Technologies and the Failure of Established Firms. *Administrative Science Quarterly* 35(1):9–30.
- Hsuan, Juliana (1999). Impacts of Supplier–Buyer Relationships on Modularization in New Product Development. *European Journal of Purchasing and Supply Management* 5(3–4):197–209.
- Langlois, Richard N. and Robertson, Paul L. (1992). Networks and Innovation in a Modular System: Lessons from the Micro-computer and Stereo Component Industries. *Research Policy* 21(4): 297–313.
- Langlois, Richard N. and Robertson, Paul L. (1995). *Firms, Markets and Economic Change*. London: Routledge.
- Link, Albert N. and Tasse, Gregory (1987). *Strategies for Technology-Based Competition*. Lexington, MA: Lexington Books.
- MacDuffie, John P., Sethuraman, Kannan and Fisher, Marshall L. (1996). Product Variety and Manufacturing Performance: Evidence from the International Automotive Assembly Plant Study. *Management Science* 42(3):350–69.
- Meyer, Marc H. and Lehnerd, Alvin P. (1997). *The Power of Product Platform*. New York: Free Press.
- Mikkola, Juliana H. (2003). Modularity, Component Outsourcing, and Inter-Firm Learning. *R&D Management* 33(4):439–54.
- Mikkola, Juliana H. and Gassmann, Oliver (2003). Managing Modularity of Product Architectures: Toward an Integrated Theory. *IEEE Transactions on Engineering Management* 50(2):204–18.
- Milgrom, Paul and Roberts, John (1990). The Economics of Modern Manufacturing: Technology, Strategy, and Organization. *American Economic Review* 80(3):511–28.
- Moscato, Donald R. (1976). The Application of the Entropy Measure to the Analysis of Part Commonality in a Product Line. *International Journal of Production Research* 14(3):401–6.
- Muffatto, Moreno (1999). Introducing a Platform Strategy in Product Development. *International Journal of Production Economics* 60–61:145–53.
- Orton, J. Douglas and Weick, Karl E. (1990). Loosely Coupled Systems: A Re-Conceptualization. *Academy of Management Review* 15(2):203–23.
- Passy, U. (1970). Modular Design: An Application of Structured Geometric Programming. *Operations Research* 18(3):441–53.
- Pine, Joseph (1993). *Mass Customization—The New Frontier in Business Competition*. Boston: Harvard Business School Press.
- Robertson, David and Ulrich, Karl (1998). Planning for Product Platforms. *Sloan Management Review* 39(4):19–31 (Summer).
- Rutenberg, David P. (1971). Design Commonality to Reduce Multi-Item Inventory: Optimal Depth of a Product Line. *Operations Research* 19:491–509 (March–April).
- Rutenberg, David P. and Shaftel, Timothy L. (1971). Product Design: Subassemblies for Multiple Markets. *Management Science* 18(4): B220–31.
- Sadowski, Wieslaw (1959). A Few Remarks on the Assortment Problem. *Management Science* 6(1):13–24.
- Sanchez, Ron (1999). Modular Architectures in the Marketing Process. *Journal of Marketing* 63:92–111 (Special Issue).
- Sanchez, Ron and Mahoney, Joseph T. (1996). Modularity, Flexibility, and Knowledge Management in Product and Organisation Design. *Strategic Management Journal* 17:63–76 (Winter Special Issue).
- Sanderson, Susan W. and Uzumeri, Mustafa (1997). *Managing Product Families*. Singapore: Irwin/McGraw-Hill.
- Schilling, Melissa A. (2000). Toward a General Modular Systems Theory and Its Application to Interfirm Product Modularity. *Academy of Management Review* 25(2):312–34.
- Schilling, Melissa A. and Steensma, H. Kevin (2001). The Use of Modular Organizational Forms: An Industry-Level Analysis. *Academy of Management Journal* 44(6):1149–68.
- Shaftel, Tim L. (1971). An Integer Approach to Modular Design. *Operations Research* 19(1):130–4.
- Simon, Herbert (1962). The Architecture of Complexity. *Proceedings of the American Philosophical Society* 106:467–82 (December).
- Starr, Martin K. (1965). Modular Production—A New Concept. *Harvard Business Review* 43(6):131–42 (November–December).
- Taylor, Frederick W. (1967). *Principles of Scientific Management*. New York: Norton.

- Tassey, Gregory (2000). Standardization in Technology-Based Markets. *Research Policy* 29(4–5):587–602.
- Teece, David J. (1986). Profiting from Technological Innovation: Implications for Integration, Collaboration, Licensing and Public Policy. *Research Policy* 15(6):285–305.
- Ulrich, Karl T. (1995). The Role of Product Architecture in the Manufacturing Firm. *Research Policy* 24(3):419–40.
- Ulrich, Karl T. and Ellison, David (1999). Holistic Customer Requirements and the Design-Select Decision. *Management Science* 45(5):641–58.
- Ulrich, Karl T. and Eppinger, Steven D. (1995). *Product Design and Development*. New York: McGraw-Hill.
- Ulrich, Karl T. and Pearson, Scott (1998). Assessing the Importance of Design through Product Archeology. *Management Science* 44(3): 352–69.
- Ulrich, Karl, Sartorius, David, Pearson, Scott and Jakiela, Mark (1993). Including the Value of Time in Design-for-Manufacturing Decision Making. *Management Science* 39(4):429–47.
- Utterback, James M. (1994). *Mastering the Dynamics of Innovation*. Boston: Harvard Business School Press.
- van Hoek, Remko I. (2001). The Rediscovery of Postponement: A Literature Review and Directions for Future Research. *Journal of Operations Management* 19(2):161–84.
- Wolfson, M.L. (1965). Selecting the Best Length to Stock. *Operations Research* 13(4):570–85.

Appendix A. The Derivation of the Modularization Function, $M(n_{NTF})$

A given product architecture is comprised of four types of components: (1) standard-customizable (n_{STD-C}); (2) standard-noncustomizable (n_{STD-NC}); (3) NTF-customizable (n_{NTF-C}); and (4) NTF-noncustomizable (n_{NTF-NC}). Component customization denotes whether the component can be modified to fit a firm's manufacturing processes. It is assumed that (1) NTF components do not have standardized interface specifications, whereas STD components do; and (2) customization of STD components is easier than customization of NTF components. Hence, product architectures tend to have more STD components than NTF components. The total number of components, N , and the proportion of NTF components, b , present in a given product architecture are

$$N = n_{STD} + n_{NTF} = (n_{STD-C} + n_{STD-NC}) + (n_{NTF-C} + n_{NTF-NC}); n_{NTF-NC} \leq n_{NTF \setminus C} < n_{STD-NC} < n_{STD-C}$$

$$b = \frac{n_{NTF}}{N} = \frac{n_{NTF-NC} + n_{NTF-C}}{N}; 0 \leq b \leq 1.0.$$

It is assumed that there is a relationship between degree of modularization, M , and the number of NTF components, n_{NTF} , $M = f(n_{NTF})$. The lower the number of NTF components—that is, the lower the composition of NTF components, b —the higher the degree of modularization. b is similar to project scope (Cusumano and Nobeoka, 1992)—the percentage of unique components a manufacturer designs from scratch in house. Here, b also includes the NTF components designed by the suppliers. Hence, a perfect modular product architecture has no NTF components ($b = 0$).

The degree of modularization, M , decreases at a rate, r , proportional to the amount of modularization present in each set of NTF components, n_{NTF} . If M is the amount of modularization present in a given product architecture in any set of NTF components, n_{NTF} , then the amount of modularization will change by the amount of $\Delta M = rM$ as the number of NTF components vary. In other words, for any unit change of NTF components ($\Delta n_{NTF} = 1$), the corresponding amount of modularization change ΔM is proportional to the initial level of modularization.

$$\Delta M = (-rM)\Delta n_{NTF} \quad \text{and} \quad r = \frac{b}{s\delta} = \frac{n_{NTF}/N}{s\delta}.$$

The degree of coupling, δ , measures the tightness of coupling of a given product architecture. The higher the value of δ , the lower the degree of modularization embedded in product architectures. Conversely, the substitutability factor, s , provides the opposite effect, as it measures the number of product families using the NTF components. Hence, the higher the value of s , the greater is the degree of modularization. Both s and δ take the number of interfaces, k , as a variable. $s\delta$ can be interpreted as the cumulative interface constraint effect of subsystems across product families; it captures the amount of modularization imposed by interfaces. the rate at which the composition of NTF components is averaged out across $s\delta$, then, is r . Hence,

$$\Delta M = (-rM)\Delta n_{NTF} = \left(-\frac{n_{NTF}/N}{s\delta}\right)M\Delta n_{NTF}.$$

Appendix A (Cont'd)

In differential equation form,

$$\frac{dM}{dn_{NTF}} = -\frac{n_{NTF}}{Ns\delta} M \quad \text{or} \quad \frac{dM}{M} = -\frac{n_{NTF}}{Ns\delta} dn_{NTF}.$$

For any constant r , the solutions to the previous differential equation are of the form

$$M(n_{NTF}) = M_0 e^{-n_{NTF}^2/2Ns\delta}.$$

Solution for the initial condition: $M(0) = M_0 = 1.0 \Rightarrow M(n_{NTF}) = e^{-n_{NTF}^2/2Ns\delta}$. The degree of coupling, δ , measures the tightness of coupling of the product architecture. For each level of decomposition, there is a value of δ associated with the decomposition (e.g., δ_{system} , $\delta_{subsystem}$, δ_{module}). For instance, δ_i of a particular subsystem, I , is calculated as the ratio of the number of interfaces, k_c , and the number of components, n_c , and n_{NTF} of the subsystem:

$$\delta_i = \frac{\text{no. of interfaces of subsystem } i}{\text{no. of components of subsystem } i} = \frac{\sum k_c}{\sum n_c}. \quad (2a)$$

For product architectures with multiple subsystems, the degree of coupling for the subsystem level of decomposition, $\delta_{subsystem}$, is approximated as the average of all δ_i :

$$\delta_{subsystem} = \delta_{average} = \frac{\sum_{i=1}^I \delta_i}{I}, \quad (2b)$$

where I = number of subsystems.

The overall degree of coupling is the sum of the δ values for each level of decomposition:

$$\delta = \delta_{system} + \delta_{subsystem} + \delta_{module} + \dots + \delta_{lowest \text{ level of decomposition}}. \quad (2c)$$

The substitutability factor, s , equals the total number of product families in which the NTF components are used, divided by the average number of interfaces required for functionality, k_{NTF} :

$$s = \frac{\text{no. of product families}}{k_{NTF}(\text{avg})} = \frac{\sum_{j=1}^L PF_j}{\frac{\sum_{i=1}^K k_{NTF}}{K}}, \quad (3)$$

where L = number of product families and K = total number of interfaces of NTF components.
