

**The Mirroring Hypothesis:
Theory and Evidence on the Correspondence between the
Structure of Products and Organizations**

Lyra Colfer*

November 22, 2007

*Ph.D Candidate
Harvard Business School

Direct correspondence to:
lcolfer@hbs.edu

© Lyra Colfer, 2007

The Mirroring Hypothesis: Theory and Evidence on the Correspondence between the Structure of Products and Organizations

Lyra Colfer
Harvard University
DRAFT DATED NOVEMBER 22, 2007

Abstract

The relationship between product architecture and organizational structure is a topic of great interest in contemporary practice and research. Much of the literature that examines this relationship posits a “mirroring hypothesis”—that is, the claim that the structure of a product development organization must necessarily “mirror” the architecture of the product it develops. In this paper, I explain the theoretical underpinnings of this hypothesis and review the empirical evidence for and against it. The relevant literature includes works in engineering, organizational design, new product development, strategy, and economics. Notwithstanding this broad scope, the research falls into two broad classes: (1) a largely normative stream that aims to prescribe how best to organize and manage the internal development efforts of a single firm and (2) a more descriptive stream that seeks instead to explain when and why some firms choose to rely (or not to rely) on the development efforts of external suppliers. I review both streams and find support for the mirroring hypothesis in some settings but not others. This result indicates that conventional views on mirroring are too simplistic: rather than asking *whether* the mirroring hypothesis holds as a general proposition, a contingent theory is needed to explain when it holds, as well as when and why it may not hold.

1 Introduction

The relationship between product architecture and organizational structure is a topic of great interest in contemporary industrial practice and academic research. Ulrich (1995: p. 419) defined *product architecture* as “the scheme by which the function of a product is allocated to physical components.” I define *organizational structure* in parallel, as the scheme by which the function of an organization is allocated to distinct organizational entities (cf. Child, 1977).

Much of the literature on the relationship between product architecture and organizational structure posits a “mirroring hypothesis.” Loosely put, this hypothesis holds that the structure of a product development organization must “mirror” the architecture of the product it develops. In other words, because a specific architecture is thought to imply a certain partitioning of *product development tasks*, it is also thought to determine the feasible/optimal structure of the corresponding *product development organization* (Conway, 1968; Armour and Teece, 1980; von Hippel, 1990; Reichtin, 1991; Henderson and Clark, 1990; Sanchez and Mahoney, 1996; Teece, 1996; Chesbrough and Teece, 1996; Fine, 1998; Baldwin and Clark, 2000; MacCormack et al., 2006; Baldwin, 2007).

The concept of modularity permits a more precise statement of the mirroring hypothesis. *Modularity* aims to capture the degree to which the elements in a system are partitioned into functionally specialized units, or *modules*, that operate nearly independently and can be easily removed or replaced because they interact through simple, standardized interfaces (Schilling, 2000; Baldwin and Clark, 2000: pp. 63-64). Since modularity is a general systems’ concept, it applies equally to products and organizations. Thus, a *modular product* is one that is partitioned into distinct *components* that are (1) narrowly specialized, (2) nearly independent, and (3) easy to remove or replace because they interact through simple, standardized interfaces (Ulrich, 1995; Baldwin and Clark, 2000: Ch. 2). Likewise, a *modular organization* is one that is partitioned into distinct *organizational entities* (e.g., individuals, teams, firms) that are (1) narrowly specialized,

(2) nearly independent, and (3) easy to remove or replace because they interact through simple, standardized exchanges (Schilling and Steensma, 2001). By contrast, the opposite of a fully “modular” product or organization is a fully “integral” one—that is, a product or organization that either (1) lacks partitioning into specialized parts or (2) is partitioned, but whose parts are tightly integrated and difficult to remove or replace because they interact through rich, tacit interfaces.¹

Stated in terms of the concept of modularity, the mirroring hypothesis holds that *an integral organization is necessary for developing an integral product, while a modular organization is only capable of developing a modular product*. Most of the literature that invokes this claim falls into two broad classes: (1) a largely normative stream of research that aims to prescribe how best to organize and manage the internal development efforts of a single firm and (2) a more descriptive stream of research that seeks instead to explain when and why firms choose to rely (or not to rely) on the development efforts of external suppliers. In this paper, I outline the intellectual heritage and empirical findings of both of these streams of research.

2 The History and Applications of the Mirroring Hypothesis

The research that invokes the mirroring hypothesis commonly draws on two distinct sources for its motivation: (1) the management literature on organization design and organizations as complex systems (e.g., March and Simon, 1958; Simon, 1962; Thompson, 1967; Galbraith, 1974; Orton and Weick, 1990) and (2) the engineering literature on product design, product development, and products as complex systems (e.g., Simon, 1962; Alexander, 1964; Conway, 1968; Parnas, 1971; 1972; 1978; Brooks, 1975; Steward, 1981a; 1981b). To distinguish between

¹ In practice, few systems are fully modular or integral. A real-world system may fall somewhere along a spectrum between the two ideals, or it may be modular in some parts but integral in others. It may also be modular at one level of analysis (e.g., a set of plug-and-play components comprise a modular product) but integral at another (e.g., the internal architecture of each component is integral). Therefore, the outcome of any given effort to measure modularity is always a matter of relative, rather than absolute, degree; it depends on *which* systems or subsystems the analyst compares.

the two, I refer to the former as the “organization design tradition” and the latter as the “engineering tradition.” Both rely heavily on the concept of modularity, which I elaborate next.

2.1 Modularity

Scholars who favor the organization design tradition usually attribute the concept of modularity to Herbert Simon (1962; 1973). In his seminal discussion of the “architecture of complexity,” Simon (1962) used the parable of Hora and Tempus to illustrate the advantage of partitioning a complex problem into parsimoniously linked subproblems. The parable compares the work of Hora, a watchmaker who follows this strategy, with the work of Tempus, a watchmaker who doesn’t:

The watches that Hora made were no less complex than those of Tempus. But he had designed them so that he could put together subassemblies of about ten elements each. Ten of these subassemblies, again, could be put together into a larger subassembly; and a system of ten of the latter subassemblies constituted the whole watch. Hence, when Hora had to put down a partly assembled watch in order to answer the phone, he lost only a small part of his work, and he assembled his watches in only a fraction of the man-hours it took Tempus.

By partitioning the architecture of a watch into subassemblies, Hora made it easier to cope with the complexity of producing a watch. Moreover, although Simon did not discuss it, Hora also made it easier to split the process of a producing a watch across a group, since his architecture allows for the various subassemblies to be completed independently and in parallel.

In addition to crediting Simon, scholars who favor the engineering tradition also stress related works by the architect Christopher Alexander and the computer scientist David Parnas. Alexander (1964) focused on the same point as Simon (1962): it is easier for a person to cope with the complexity of a large-scale problem if the problem is decomposed into parsimoniously linked subproblems. Parnas (1971; 1972; 1978) stressed the benefits of parallelism: it is easier to split the work across a group if contributors can work independently and in parallel.

To support parallelism, Parnas encouraged developers to avoid sharing assumptions and data by conceiving of their assignments as “black boxes” with narrowly specialized functions, privately concealed implementations, and “lean” public interfaces. More formally, he (1972: p. 1056) contended that every work assignment or product module should be “characterized by its knowledge of a design decision that it hides from all others.” Today, this rule is known as the principle of *information hiding*.

Information hiding is the fundamental “design rule” (Baldwin and Clark, 2000) that guides the “modularization” of a product development process. In the “modularized” process, each contributing entity develops a different “module” of the product. By definition, each module performs a narrowly specialized function and operates largely independently of the other modules. As a result, there is a one-to-one mapping from both (1) contributing entities to product modules (Parnas, 1972) and (2) product modules to product functions (or features) (Ulrich, 1995).

Parnas’ conception of a module as “a responsibility assignment rather than [just] a subprogram” (Parnas, 1972: p. 1052) highlights the fact that a product architecture can simultaneously be a division of labor (Herbsleb and Grinter, 1999a: p. 85). From this perspective, the process of modularization serves two distinct goals: (1) it reduces the intrinsic complexity of the product design problem per se (Simon, 1962; Alexander, 1964) and (2) it reduces the extrinsic complexity of splitting the work across a group (Parnas, 1971; 1972; 1978; Simon, 1973; von Hippel, 1990; Brooks, 1995).

In later work, Simon explicitly stressed the significance of both of these goals. To motivate the first (e.g., reducing the complexity of the design problem per se), he wrote,

From the information processing point of view, division of labor means factoring the total system of decisions that need to be made into relatively independent subsystems, each one of which can be designed with only minimal concern for its interactions with the others.

The division is necessary because the processors that are available to organizations, whether humans or computers, are very limited in their processing capacity in comparison

with the magnitude of the decision problems that organizations face. The number of alternatives that can be considered; the intricacy of the chains of consequences that can be traced—all of these are severely restricted by the limited capacities of the available processors (Simon, 1973: p. 270; emphasis added).

To motivate the second goal (e.g., reducing the complexity of coordinating the group that solves the design problem), Simon proceeded,

Any division of labor among decisional subsystems creates externalities, which arise out of the interdependencies among the subsystems that are ignored. What is wanted is a factorization that minimizes these externalities and consequently permits a maximum degree of decentralization...and a maximum use of relatively simple and cheap coordinating devices like the market mechanism to relate each of the decisional subsystems with the others (Simon, 1973: p. 270; emphasis added).

These two contrasting motivations for modularization point to the fact that the *target for modularization* is not always the same. If one's goal is to reduce the intrinsic complexity of the product design problem per se, then the target is the product architecture. By contrast, if one's goal is to reduce the extrinsic complexity of coordinating the group that solves the problem, then the target is the division of labor.

In fact, there are at least four distinct “layers” of structure in the phenomenon of product development that are common targets for modularization: (1) the *product*, (2) the *product development organization*, (3) the *division of labor*, and (4) the *division of knowledge*.² The product is *what* is under development. In this case, the target is the network of technical dependencies among the elements comprising the good or service, itself (e.g., Ulrich, 1995; Baldwin and Clark, 2000: Chapter 2; Sharman and Yassine, 2003; Mikkola, 2006). The

² Baldwin and Clark (2000: p. 47) identified three distinct “‘layers’ of structure in the design of an artifact”: (1) the physical structure of the artifact per se, (2) the design structure that describes the elements of the artifact and the relationships among them, and (3) the task structure of the artifact design process. I use the terms “product architecture” and “design structure” interchangeably.

organization is *who* does the work. In this case, the target is the network of communication and authority relationships among the different organizational entities (e.g., individuals, teams, firms) who develop the good or service (e.g., Robertson and Langlois, 1995; Baldwin and Clark, 2000: Chapter 14; Schilling and Steensma, 2001; Brusoni and Prencipe, 2001a; 2001b; Sturgeon, 2002; Staudenmeyer et al., 2005). Finally, the divisions of labor and knowledge are *how* the work gets done. In the first case (e.g., the division of labor), the target is the network of technical dependencies among the *development tasks* of the different contributors (e.g., von Hippel, 1990; Eppinger et al., 1994; Baldwin and Clark, 2000: Chapter 14; Ernst, 2003; Baldwin, 2007). In the second (e.g., the division of knowledge), it is the network of technical dependencies among the *information/skill sets* required to perform those tasks (e.g., Brusoni and Prencipe, 2001a; 2001b).³

Once scholars began to recognize that the phenomenon of product development could be envisioned as a set of multiple distinct layers of structure, they also began to hypothesize about how the structure of one layer might relate to that of others. In particular, many authors have posited a “mirroring hypothesis”—that is, the claim that the structure of a product development organization must necessarily “mirror” the architecture of the product it develops. In the next section, I discuss this conventional wisdom and explain how it is understood in the literature that invokes it.

2.2 The Mirroring Hypothesis

Loosely speaking, the general idea of the mirroring hypothesis is that the structure of an organization “mirrors” the architecture of the product it develops. More formally, the mirroring hypothesis holds that an integral organization is necessary for developing an integral product, while a modular organization is only capable of developing a modular product. Most of the literature that invokes this claim falls into two broad classes: (1) a largely normative stream of

³ These are by no means the *only* layers of structure in the phenomenon of product development; there are also others, such as the division of property rights.

research that aims to prescribe how best to organize and manage the internal development efforts of a single firm and (2) a more descriptive stream of research that seeks instead to explain when and why some firms choose to rely (or not to rely), at least in part, on the complementary development efforts of *other* firms. To distinguish between the two, I refer to the former as the “within-firm literature” and the latter as the “across-firm literature.”

Though both classes of literature share a common definition of *product modularity* (e.g., a modular product is one that is partitioned into functionally specialized components with simple, standardized technical interfaces), they differ in their definitions of *organizational modularity*. Specifically, the within-firm literature focuses on communication patterns, while the across-firm literature also considers patterns of ownership and authority. Below, I review each class of literature in turn.

2.2.1 Invoking the Mirroring Hypothesis within the Firm

The first scholars to propose and apply the mirroring hypothesis sought mainly to provide managers and developers with (1) normative recommendations for how best to organize their efforts and (2) explicit warnings about common associated challenges and pitfalls. This stream of research assumes that developers are compatibly motivated but cognitively bounded. In other words, it stresses the fundamental problem of *coordination* over that of *cooperation*. From this perspective, the organization’s key problem is that its contributors will inevitably experience unintended information asymmetries as a consequence of their distributed cognition and bounded abilities as information processors (March and Simon, 1958; Conway, 1968; Brooks, 1975; Parnas, 1972; von Hippel, 1990; Weick and Roberts, 1993; Tsoukas 1996; Ocasio, 1997).

Communication then becomes a central concern. When there are many complex interdependencies among the components of a product, a change in one component will often necessitate many compensating changes in other components (von Hippel, 1990; Ulrich, 1995). If contributors fail to address these interdependencies, there are two ways to fix the problem: they

can redesign the *product* to reduce the need for communication (e.g., by simplifying and standardizing component interfaces) or they can redesign the *organization* to improve communication (e.g., by collocating developers who work on interdependent components). Either way, the underlying principle is the same: since developers must communicate to address the interdependencies between their efforts, there should be an isomorphic relationship between the organization's communication (and information-processing) structure and the architecture of the product under development (Conway, 1968; Henderson and Clark, 1990).⁴ In the words of Henderson and Clark (1990: p. 27), “[The organization's] knowledge and information processing structure... mirror the internal structure of the product they are designing.”

In light of this observation, the literature makes three related recommendations: (1) modularize the product (e.g., Simon, 1962; Alexander, 1964; Ulrich, 1995), (2) modularize the organization accordingly (e.g., Parnas, 1972; 1978; von Hippel, 1990; Herbsleb and Grinter, 1999a; 1999b), and (3) track communication needs over time in order to adjust the organization as the product evolves (e.g., Conway, 1968; Allen, 1977; Henderson and Clark, 1990; Herbsleb and Grinter, 1999a; 1999b; Sosa et al., 2004; Sosa et al., 2007).

While these recommendations may seem straightforward, the literature suggests that they are often quite difficult to implement. The first (e.g., to modularize the product) can be challenging, especially in the earlier phases of development, because the group rarely knows enough at the outset to achieve a viable product modularization. An iterative process of trial and error is usually necessary to arrive at a sound modular architecture, and this process generally requires that the structure of the organization remain flexible for some time (Baldwin and Clark, 2000: pp. 64-86). After studying the modularizations of the very-large-scale integrated (VLSI) microchip and the IBM System/360, Baldwin and Clark (2000: pp. 76-77) reported,

⁴ The basic effect of task interdependence is task uncertainty—and the basic effect of task uncertainty is to limit the organization's ability to preplan or make *ex ante* decisions about how the tasks will be performed (Galbraith, 1974: 28). Thus, greater task interdependence engenders greater task uncertainty and this, in turn, engenders greater requirements for inter-task communication and information-processing (cf. Thompson, 1967; Galbraith, 1974).

Perfectly modular designs do not spring fully formed from the minds of architects. When a design is first modularized, the designers' knowledge of interdependencies is usually imperfect, and as a result the initial set of design rules will be incomplete. Incomplete design rules give rise to unforeseen interdependencies, which in turn will require consultations and iterations between the hidden-module designers and the architects of the system. The integration and testing of these designs will be fraught with difficulty and the designs themselves will have high risks of failure.

The more complex the product, the more these challenges will be exacerbated by the natural limits on contributors' information-processing abilities. Empirical studies of product development in the automotive and aerospace industries (e.g., Allen, 1966; Loch and Terwiesch, 1999; Terwiesch et al. 2002; Clarkson et al., 2004) suggest that many design iterations are necessary because "engineers and managers cannot fully anticipate the impact that all their local design decisions have on other components" (Sosa et al. 2007: p. 7). So as long as a product is sufficiently complex, it is inevitable that *some* of the interdependencies among its components will remain unknown and unnoticed for much (or all) of its design and development (Sosa et al., 2004; see also Schrader et al., 1993; Pich et al., 2002).

Sosa et al. (2004; 2007) looked for evidence and correlates of these "unattended" component interdependencies in a pair of related studies. Investigating the development of a large commercial aircraft engine, Sosa et al. (2004) found evidence of a number of direct interdependencies between components that were *not* matched by direct communication relationships between the components' developers. In a second analysis of the same project, Sosa et al. (2007) provided further evidence that *indirect* links in the architecture and communication network also mattered in significant but distinct ways: whereas the former increased the risk of "unforeseeable" design iterations, the latter facilitated trust and knowledge sharing, and thereby enabled developers to attend to more interdependencies sooner (Sosa et al., 2007: p. 7).

Even if the group hits upon a viable product modularization in which most component interdependencies are well known and understood, the second recommendation in the within-firm literature (e.g., to modularize the organization accordingly) can still be problematic because many groups are not amenable to restructuring. In practice, it is not always feasible to consolidate all of the necessary resources for developing a component within a given location or organizational entity. In a study by Srikanth and Puranam (2007), the managers at two large software services firms reported a variety of constraints on their efforts to reorganize product development projects. One common source of constraints was the need for highly specialized skills and expertise. As one manager explained,

The developers with the required skills are not always available in a single location, and they do not always relocate. To have a large number of developers traveling will blow our budget. There is no other alternative to using distributed models for these projects (Srikanth and Puranam, 2007: p. 10).

In the same study, client objectives were another common source of constraints. For example, a client might require developers to collaborate with third-party vendors in order to take advantage of prior licensing and strategic agreements, or it might impose strict security provisions regarding which developers may access its data and systems, as well as where those developers must be in order to do so. In many of the projects that Srikanth and Puranam studied, constraints like these trivially precluded a clean mapping between the component boundaries of the product under development and the geographic and administrative boundaries of the project's development organization (p. 13).

The final recommendation in the within-firm literature (e.g., to track communication needs over time in order to adjust the organization as the product evolves) is often difficult to implement, not only because of the various constraints on reorganization noted above, but also because changing communication needs can be difficult to perceive. If the group begins without

an explicit idea of how to structure the product, it may pursue a product architecture that mirrors the structure of the extant organization, regardless of whether or not this structure yields a particularly sensible solution to the *product* design problem (Conway, 1968).⁵ Alternatively, if the group begins with a specific conception of how to organize the product (e.g., a “dominant design”), then it is much more likely “to organize itself around its conception of the product’s primary components, since these are the key subtasks of [its] design problem” (Henderson and Clark, 1990: p. 15; see also Mintzberg, 1979; von Hippel, 1990). *In either case*, there are two key obstacles to the group’s capacity to perceive its communication needs accurately—namely, contributors may not all share the same (correct) understanding of the architecture under development, and, even if they do, this architecture is likely to change over time.

Though there are many reasons why contributors may fail to share a common (and correct) understanding of their product’s architecture, the literature tends to focus on the effects of bounded cognition (e.g., Ocasio, 1997; Sosa et al., 2004; Sosa et al., 2007), functional specialization (e.g., Allen, 1977; Tushman and Katz, 1980; Van den Bulte and Moenaert, 1998; Bechky, 2003; Sosa et al., 2004; Kellogg et al., 2006) and spatiotemporal distance (e.g., Herbsleb and Grinter, 1999a; 1999b; Grinter et al., 1999; Sosa et al., 2002; Allen, 2007). When a product is complex, bounded cognition usually prevents *anyone* from understanding its architecture with perfect accuracy (Sosa et al., 2004; Sosa et al., 2007). Moreover, functional specialization can further inhibit the group’s understanding by fostering technically distinct modes of thinking and communicating that make it difficult for contributors to share their knowledge in a *mutually* meaningful way (Bechky, 2003: p. 312; see also Jelinek and Schoonhoven, 1990). Likewise, spatiotemporal distance can also compound the problem, both by localizing the scope of shared

⁵ Conway (1968) offered this simple anecdotal example to illustrate: A contract research organization assigned eight people to produce a COBOL compiler and an ALGOL compiler. After estimating the difficulty and time required to develop each, the group assigned five programmers to the COBOL project and three to the ALGOL project; the resulting COBOL compiler ran in five phases, while the resulting ALGOL compiler ran in three.

experience even further (Herbsleb and Grinter, 1999a; 1999b), as well as by necessitating the use of “lean,” or technology-mediated, communication (Daft and Lengel, 1984; 1986).

Even if the group does share the same (correct) understanding of the product’s architecture, this understanding will become outmoded as the architecture evolves. Whereas the group’s *needs* for communicating and processing information will naturally follow the evolving architecture, the group’s *routines* for doing so probably won’t (Henderson and Clark, 1990; Sosa et al., 2007).

After studying evidence from the photolithographic equipment industry, Henderson and Clark (1990: p. 28) concluded,

Given the way knowledge tends to be organized within the firm, learning about changes in the architecture of the product is unlikely to occur naturally. Learning about changes in architecture—about new interactions across components (and often across functional boundaries)—may therefore require explicit management and attention.

In other words, the contributors’ shared understanding of the old architecture, along with their existing patterns of communication and organization, will often *impede* their ability to perceive how changes to the product’s architecture have necessitated corollary changes to their communication and organization.

In sum, the within-firm literature provides both a normative prescription for how best to organize the internal development efforts of a single firm, as well as a substantial body of case-based evidence to illustrate common associated challenges and pitfalls. The across-firm literature is more descriptive in character. It invokes the mirroring hypothesis to explain why the development of some products is confined within firms, while the development of other products spans across multiple firms. Put another way, it seeks to explain the location of firm boundaries as a function of product architecture. I discuss this literature next.

2.2.2 *Invoking the Mirroring Hypothesis across Firms*

The across-firm literature invokes the mirroring hypothesis to explain when and why firms choose to rely (or not to rely), at least in part, on the complementary development efforts of other firms. Specifically, the across-firm version of the mirroring hypothesis contends that a *firm* is necessary for developing an integral product, while a *group of firms* is only capable of developing a modular product. Sanchez and Mahoney (1996) were among the first to reframe the within-firm version of the mirroring hypothesis in this new way. Pointing to the well-known challenges associated with interdependent development tasks (von Hippel, 1990; Henderson and Clark, 1990; Ulrich and Eppinger, 1995), they asserted,

...*processes* for developing tightly coupled component designs require intensive managerial coordination, since a change in the design of one component is likely to require extensive compensating changes in the designs of many interrelated components. Thus, product designs composed of tightly coupled components will generally require development processes carried out in a *tightly coupled organization structure* coordinated by a managerial *authority hierarchy*, [that is] an organization design typically achieved within a single firm (Sanchez and Mahoney, 1996: p. 65; emphasis in original).⁶

Naturally, they drew an analogous parallel at the other end of the spectrum:

⁶ Different authors use the term “coupling” differently. One common use has its roots in the engineering literature on product design. Here, authors use the term to describe *one* dimension of system structure: the degree of interconnectivity among system elements. In this case, system elements are said to be “tightly coupled” or “highly coupled” if they exhibit extensive interconnections or interdependencies; otherwise they are said to be “loosely coupled” or “not highly coupled.” The second use of the term “coupling” has its roots in the management literature on organizations, where Weick (1976; 1982) and Orton and Weick (1990) redefined it to capture *two* dimensions of system structure: *distinctiveness*, or the degree to which organizational entities retain evidence of separateness and independent identity (1976: p. 3), and *responsiveness*, or the degree to which organizational entities affect one another constantly (rather than occasionally), significantly (rather than negligibly), directly (rather than indirectly), and immediately (rather than eventually) (1982: p. 380). In this case, authors like Sanchez and Mahoney (1996) describe an organization as (1) “tightly coupled” if its elements exhibit “responsiveness without distinctiveness” (Orton and Weick, 1990: p. 205) or (2) “loosely coupled” if its elements exhibit “[at least some] distinctiveness and [limited] responsiveness” (Orton and Weick, 1990: p. 205).

In essence, the [loosely coupled] standardized component interfaces in a modular product architecture provide a form of *embedded coordination* that greatly reduces the need for overt exercise of managerial authority to achieve coordination of development processes, thereby making possible the concurrent and autonomous development of components by *loosely coupled organization structures* (Sanchez and Mahoney, 1996: p. 64; emphasis in original).

In other words, Sanchez and Mahoney (1996) viewed the overt exercise of managerial authority as a necessary mechanism for coordinating the development of an integral product, but not a modular one.

Teece (1986; 1988; 1992; 1996) proposed the same idea in different language. Specifically, he partitioned product innovation into two distinct classes: (1) *systemic innovations*, or those that require significant complementary adjustments throughout many components, and (2) *autonomous innovations*, or those that only require localized changes within a component. Stated in terms of the concept of modularity, a systemic innovation is one that is *integral*, or spans many modules, while an autonomous innovation is one that is *modular*, or confined within a module. Accordingly, the key difference between the two is the amount of explicit coordination that each requires: whereas a systemic innovation will require the developers of different components to coordinate their efforts explicitly, an autonomous innovation will not. For Teece (1996), just as for Sanchez and Mahoney (1996), this difference implies that the internal management processes of a firm are necessary for coordinating the development of a systemic (e.g., integral) innovation, but not an autonomous (e.g., modular) one. Writing with Henry Chesbrough, Teece elaborated,

The distinction between autonomous and systemic innovation is fundamental to the choice of organizational design. When innovation is autonomous the decentralized virtual organization can manage the development...tasks quite well...The information needed to

integrate an autonomous innovation with existing technologies is usually well understood and may even be codified in industry standards. Systemic innovations, on the other hand, pose a unique set of management challenges regarding information exchange. By their very nature, systemic innovations require information sharing and coordinated adjustment throughout an entire product system...In most cases, the open exchange of information that fuels systemic innovation will be *easier and safer* within a company than across company boundaries. The inevitable conflicts and choices that arise as a systemic innovation develops can best be resolved by an integrated company's internal management processes (Chesbrough and Teece, 1996: pp. 128-129; emphasis added).

In the literature more generally, many others have argued the same (e.g., Langlois and Robertson, 1992; Robertson and Langlois, 1995; Chesbrough and Kusunoki, 2001; Christensen et al., 2002; Chesbrough, 2003a; 2003b; Mikkola, 2003).

So why do Teece, Chesbrough, and others assume that the open exchange of information is “easier and safer” within a firm? Two well-known schools of thought serve to justify these claims. Specifically, Williamsonian transaction cost economics (TCE) supports the claim that open exchange is *safer* within a firm, while the knowledge-based theory of the firm (KBT) supports the claim that it is also *easier*.

The TCE perspective focuses on the hazards of opportunism. Here, the key problem is how to align the potentially conflicting interests of contributors so that their exchanges can take place safely—that is, without much risk of opportunistic behaviors like the withholding of valuable information and materials. Accordingly, the TCE approach stresses the role of managerial authority as a means “to infuse order in a relation where potential conflict threatens to undo or upset opportunities to realize mutual gains” (Williamson, 1999: p. 1090). Thus, from this perspective, the key benefits of collocating contributors within a firm are the firm's superior capacity (relative to the market) for (1) incentive alignment, (2) conflict resolution, and (3) performance monitoring (Williamson, 1971).

In contrast to the TCE approach, the KBT perspective stresses the hardships of bounded cognition over the hazards of opportunism. Here, the key benefits of collocating contributors within a firm are the firm's superior capacity (again, relative to the market) for (1) central planning (Alchian and Demsetz, 1972) and (2) rich, contextual, bilateral communication (Arrow, 1974; Nelson and Winter, 1982; Monteverde, 1995).⁷ In effect, the KBT approach focuses on the same basic coordination problem as the within-firm literature discussed above in §2.2.1.

Despite emphasizing distinct organizational problems and firm attributes, the TCE and KBT perspectives both imply the same general conclusion: it will be less costly to collocate integral product development tasks within a firm than to distribute them across firms. Because integral development tasks share extensive interdependencies, they necessarily require (1) extensive communication and exchange among the individuals who perform them, as well as (2) efficient resolution of disputes arising from the individuals' differing technical perspectives, product knowledge, and/or self-interests. The TCE and KBT perspectives both imply that it will be less costly to meet these requirements within a firm (rather than across firms) because both presuppose that a *within-firm organizational structure* has key advantages for managing interdependence—namely, the capacity for *open* communication, *informal* exchange, and *authority-based* dispute resolution. By contrast, there will be additional costs to managing interdependent development tasks across firms, at least so long as an *across-firm organizational structure* requires codified product standards (rather than open communication), formal transactions (rather than informal exchange), and self-organized discrepancy resolution (rather than authority-based).⁸ Thus, *at least under these assumptions*, it follows that collocating integral

⁷ Wheelwright and Clark (1992) labeled the pattern of continuously sharing information in a rich format and interactive manner as “integrated problem solving” and contrasted it explicitly with the traditional “serial mode” in which downstream contributors wait to begin their work until after upstream contributors are finished. Monteverde (1995) referred to this rich, continuous, bilateral exchange of information as “unstructured technical dialog” and proposed it to be the *raison d'être* of the firm.

⁸ Coase (1937) argued that firms exist because there are costs to using markets—specifically, the costs of negotiating and enacting formal market transactions. When the nature of development tasks dictates that transfers among contributors must be dense and complex, it can be exceptionally difficult for contributors

development tasks within a firm will necessarily be more efficient than distributing them across firms (Baldwin and Clark, 2000: Chapter 14; Baldwin, 2007).

Empirically, scholars have sought to substantiate the (across-firm) mirroring hypothesis by pointing to the organizational patterns observed in a range of industries, including stereo systems (Langlois and Robertson, 1992), personal computers (Langlois and Robertson, 1992; Baldwin and Clark, 2000), semiconductors (Monteverde, 1995), electronics equipment (Sturgeon, 2002), mortgage banking (Jacobides, 2005), building construction (Cacciatori and Jacobides, 2005), and bicycle drivetrains (Fixson and Park, 2006).

Langlois and Robertson (1992) examined the stereo component and personal computer industries. In each case, the focal industry initially consisted of generalist firms developing integral products, but later adopted a “dominant design” with standardized component interfaces. While some firms continued to develop integral products, the ones that used the new modular standards were more successful. In turn, the structure of the industry quickly came to mirror the structure of the “winning” modular architecture, with different firms developing the different components of the product. Accordingly, Langlois and Robertson advocated the mirroring hypothesis in an even stronger form, proposing that a modular product architecture does not merely *enable* the use of an across-firm organizational structure but, rather, that it actually *induces* the use of one (p. 300).

Monteverde (1995) focused on the flip side of the same coin—that is, the possibility that *integral* product engineering tasks induce the use of a *within-firm* organizational structure. Monteverde characterized a firm’s capacity for “unstructured technical dialog,” or communication that is “rich, bilateral and intense” (Wheelwright and Clark, 1992: p. 180), as a

to specify or value the content of their transfers ahead of time; the costs associated will often make it infeasible for them to conduct the transfers as formal, arms-length transactions (Baldwin and Clark, 2000: Chapter 14; Baldwin, 2007). The solution to this problem, within a firm, is that “one open-ended contract” substitutes for many transactional contracts (Sturgeon, 2002: p. 472). In other words, the legal and social conventions of a firm create a “transaction-free zone” in which contributors are willing to perform transfers freely—that is, without the extra costs of conducting the transfers as formal transactions (Baldwin, 2007).

key incentive for consolidating interdependent product engineering tasks within a firm. “When communication is imperative,” he argued, “it will be most efficient for all engineers involved...to communicate in one single, organization-specific dialect” (p. 1629). To test this claim, Monteverde examined chip design and fabrication in the US semiconductor industry. Specifically, he predicted,

[The] greater is the [perceived need] for unstructured, undocumented communication between these two functions, the greater is the likelihood of vertical integration...
Conversely, fablessness [or dis-integration] may be expected to be viable only where the required level of unstructured technical dialog is minimal (Monteverde, 1995: p. 1630).

The results of a formal probit analysis supported Monteverde’s hypothesis: the greater the perceived need for such dialog, according to a panel of six experts, the more likely it was that the two functions were, in fact, integrated within a single firm.

Based on a broader review of US electronics industries, Sturgeon (2002) advanced a more general construction of Monteverde’s hypothesis. Examining the rise and drivers of contract manufacturing in US electronics, Sturgeon argued that a number of different electronics industries had all come to rely on the same basic organizational form: a “modular production network” in which “distinct breaks in the value chain...form at points where information regarding product specifications can be highly formalized” (p. 455). Following the literature (e.g., Ulrich, 1995; Sanchez and Mahoney, 1996), he concluded that activities within “specialized value chain nodes” (e.g., firms) remained tightly integrated and tacitly managed, while coordination between them depended on “the transfer of codified information” (p. 455).

Jacobides (2005) drew a strikingly similar conclusion from the mortgage banking industry. Examining the transformations in that industry from 1970 to 1990, Jacobides found that intermediate markets only emerged after two precipitating changes. First, bankers reduced the degree of interdependence between the different steps of producing a mortgage loan (e.g.,

brokering, warehousing, securitizing, holding, and servicing). Second, they adopted standardized loan categories and credit scoring formulas. In retrospective interviews, bankers reported that these changes made it just as easy and safe to perform the different steps across firms as it was to perform them within a firm. Accordingly, Jacobides concluded that industrial vertical dis-integration can only occur when two “necessary conditions” are met: “On one hand, when...task interdependence is reduced; and on the other hand, when [the] information [required for inter-firm exchange] becomes standardized—that is, when it becomes simple, transmissible, and universally understood” (Jacobides, 2005: p. 479).

Cacciatori and Jacobides (2005) considered the opposite scenario—that is, *re-integration*—in the British building industry. Studying the changes in that industry from 1970 onwards, they found that integrated firms only reappeared after systemic failures in the dis-integrated industry structure became self-evident. Specifically, the various specialized professions involved in building design and construction (e.g., architects, builders, contractors, quantity surveyors) had grown too narrow and insular to understand the building process as a whole, and rigid contractual separation of their efforts had left customers unable to predict or control project costs and integration requirements. In response to a growing demand for better technological integration and managerial continuity, industry players proactively pursued re-integration. Accordingly, the authors concluded that re-integration “comes about as new capabilities, spanning the entire value chain, are needed” (p. 1874). In turn, they also characterized their findings as broadly consistent with the conventional wisdom that “[organizational] re-integration may be needed to ensure effectiveness and integrality in [product] design” (p. 1874).

Fixson and Park (2006) studied a different example of re-integration. In this case, they examined the industry for US bicycle drivetrains during the 1980s. In the first half of that decade, the dominant design for a bicycle drivetrain consisted of a “perfectly modular” set of six components (p. 12). However, in 1985, one firm introduced a fully integral drivetrain. This new alternative offered superior performance due to a systemic innovation involving many changes

across several components. In effect, the firm forced its competitors into a new era of “systemic competition” (p. 28). When competitors could not close the performance gap swiftly enough, the structure of the industry soon came to mirror the structure of the “winning” integral architecture, with the innovating firm holding a vast majority of the total market for all drivetrain components.

Despite these examples, there is also mounting evidence to suggest that conventional views on mirroring are too simplistic for broader application. Empirical studies indicating the need for a more nuanced theory of mirroring span a range of industries, including automobiles (Clark, 1989; Dyer, 1996; Takeishi, 2001; Sako, 2004), military aircrafts (Argyres, 1999), aircraft engines and chemical plants (Brusoni and Prencipe, 2001b), telecommunications, publishing, and computer software (Staudenmeyer et al., 2005) and hardware (Hoetker, 2006).

Contrary to the conventional wisdom that firms are only willing (or able) to rely on external suppliers for the production of predefined standardized parts, Clark (1989) found that Japanese automakers relied on their external suppliers to help them define and develop custom parts:

In the Japanese system, [external] suppliers are an integral part of the development process: they are involved early, assume significant responsibility, and communicate extensively and directly with [internal] product and process engineers (Clark, 1989: p. 1252).⁹

Following Clark (1989), Takeishi (2001) examined the relative quality of such jointly developed parts. A statistical analysis of coded interview and survey data indicated the positive impact of three variables: (1) the extent of joint problem-solving by automaker and supplier engineers, (2) the frequency of face-to-face communication between automaker and supplier engineers, and (3) the level of architectural knowledge held by automaker engineers. Takeishi concluded that all

⁹ In a subsequent study, Dyer (1996) reported similarly, finding that Toyota and Nissan had more frequent face-to-face contact with their suppliers, as well as more guest engineers at their sites, than did their US competitors; he concluded that these factors contributed to the Japanese automakers’ relatively shorter model development cycles.

three served to improve part quality because they helped engineers to identify and solve integration problems sooner and more efficiently (p. 413).¹⁰

Argyres (1999) studied another case that contradicts conventional views on mirroring: the development of the B-2 stealth bomber. Because no one firm held all of the technical capabilities or capital resources needed to develop the product independently, the project required an across-firm organizational structure. However, due to the exacting requirements associated with stealth, the firms could not modularize the product's architecture.¹¹ Instead, they coordinated their densely and complexly linked development tasks by creating and maintaining two key joint resources: (1) a set of standard rules for how to format and interpret product data and (2) a suite of semi-automated systems to help enforce the rules and share the rule-filtered data. In other words, the group did not simplify or standardize the product's architecture, but, rather, how it communicated *about* that architecture.

Whereas Argyres (1999) showed that it is possible for a group of firms to develop an integral product, Brusoni and Prencipe (2001b) found that it can be difficult for a group of firms to develop a modular product without a "systems integrator." Studying the development of aircraft engines and chemical plants, the authors found that the use of a modular product architecture failed to enable the firms involved to coordinate their efforts via arms-length transactions. Instead, the across-firm development process called for a "highly interactive organizational set-up" (p. 202) in which a systems integrator, or "lead firm" (Langlois and Robertson, 1992), proactively orchestrated the involvement of others. Accordingly, Brusoni and Prencipe concluded that *product modularization* neither derives from, nor brings about, *knowledge modularization*. "In a nutshell," they wrote, "systems integrators [must] 'know more than they do'" (p. 202).

¹⁰ Communication frequency correlated negatively with knowledge level and problem-solving pattern, indicating that communication is sometimes used to make up for a lack of knowledge or a failure to solve problems using a supplier-integrated approach. However, once Takeishi controlled for knowledge and problem-solving, he found that more communication did, indeed, predict better component design.

¹¹ Even if the firms *had* found a way to modularize the architecture, they still would have had difficulty modularizing the division of labor to match, due to the extant bounds on their capabilities and resources.

To see how this systemic knowledge crosses firm boundaries, Sako (2004) studied the “supplier development” activities of three major systems integrators: Honda, Nissan, and Toyota. Defining *supplier development* as “a firm’s attempt to transfer (or replicate) some aspects of its in-house organizational capability across firm boundaries” (p. 282), Sako found that all three automakers offered “multiple channels of supplier development...in order to transfer both explicit and tacit knowledge” (p. 299). While they began by providing suppliers with shopfloor assistance, their efforts eventually included product development as well, and they generally encountered “no barriers to suppliers opening their books for advice” (p. 301). According to Sako, the suppliers’ trust derived not just from the automakers’ competence as teachers, but also from their having devised “a clear set of rules for sharing specific gains from short-term intervention and...for letting suppliers appropriate wider gains from long-term capability enhancement” (p. 301). In effect, Sako found that longstanding relational contracts had sufficed, at least in these cases, to motivate the voluntary open exchange of knowledge, both codified and tacit, across firm boundaries.

Staudenmeyer et al. (2005) considered three additional industries in which the outputs of different firms must work together as a system: telecommunications, web publishing, and computer software. Within these industries, the authors studied product development in seven different firms. Based on manager interviews, the authors reported that inter-firm interdependencies were “ubiquitous” and “continually emerged throughout the product development process, despite efforts to limit them” (p. 303). Indeed, even firms without formal relationships experienced architectural interdependencies that necessitated product design changes (p. 313). The authors reported that focal firms benefited from working to manage these interdependencies as they came to light during the development process, rather than trying to eliminate them *ex ante*. To illustrate, Staudenmeyer et al. (pp. 313-314) described Lucent’s strategy for managing inter-firm software development:

If two different teams or individuals caused a system “crash” upon reintegration, they communicated directly with each other to negotiate a solution—all dependencies were to be pursued person to person and not via proxy or third party. Only if an interdependency cropped up several times was there an attempt to “sever” it by consolidating control in one team... This more flexible approach, while difficult at first in that it required more frequent interactions, also enabled the various teams to adjust to ongoing changes more effectively.

In another recent study, Hoetker (2006) asked whether a firm is more likely to use an external supplier for an autonomous (e.g., modular) innovation than a systemic (e.g., integral) one. Empirically, Hoetker tested this premise by comparing the sourcing decisions associated with two different types of laptop display innovation: (1) increased display size (e.g., a systemic innovation) and (2) increased display resolution (e.g., an autonomous innovation). Reviewing the sourcing decisions of all laptop manufacturers from 1992 to 1998, Hoetker found that “product modularity did *not* help firms move activities out of hierarchy” (p. 513; emphasis added). Though the autonomous innovation did permit firms to reconfigure their internal supply chains relatively more freely, the firms still preferred internal suppliers to external ones for both types of innovation. In other words, at least in this case, the autonomous innovation did *not* induce an across-firm organizational structure.

In sum, the extant literature provides support for the (across-firm) mirroring hypothesis in some settings but not others. Considered jointly, these studies indicate that conventional views on mirroring are too simplistic to apply generally. To align with the existing evidence, conventional mirroring theory must be refined.

3 Conclusion

The relationship between product architecture and organizational structure is a topic of great interest in contemporary practice and research. Much of the literature regarding this relationship

advocates what I have labeled the “mirroring hypothesis”—that is, the claim that an integral organization is necessary for developing an integral product, while a modular organization is only capable of developing a modular product. Most of the literature that invokes this claim falls into one of two broad classes: (1) a largely normative stream of research that aims to prescribe how best to organize and manage the internal development efforts of a single firm and (2) a more descriptive stream of research that seeks instead to explain when and why some firms choose to rely (or not to rely) on the development efforts of external suppliers.

In this paper, I reviewed the history and findings of both of these streams of research. In doing so, I found support for the mirroring hypothesis in some settings but not others. This result indicates that conventional views on mirroring are too simplistic to apply generally. I contend that the source of this problem lies in the framing of the question: rather than asking *whether* the mirroring hypothesis holds, I would argue that it is more appropriate to ask *when* it holds—and *when it does not*.

By definition, integral product development tasks require extensive coordination, communication, and exchange. But do these prerequisites actually prevent modular organizations from performing integral tasks? In future work, I intend to show that the answer to this question depends on the setting in which the group works: if contributors retain not only their autonomy in task selection and performance (as in a market), but also their willingness and capacity to engage in open communication and informal exchange (as in a firm), then the mirroring hypothesis may lose its force. To illustrate, I intend to focus on the theoretically interesting and empirically amenable case of *commons-based* modular organizations developing *open-source software* products.¹²

¹² The term “open source” denotes licensing that permit anyone to use, modify, and/or redistribute product source materials. This definition does not require that people collaborate voluntarily (e.g., without compensation or assigned tasks) to produce such goods; nonetheless, they often do. I follow Benkler (2002) in referring to this mode of organizing as a “commons-based” product development organization.

References

- Alchian, A. and H. Demsetz (1972). "Production, Information Costs, and Economic Organization." *The American Economic Review*. 62(5):777-795
- Alexander, C. (1964). *Notes on the Synthesis of Form*. Harvard University Press: Cambridge, Massachusetts
- Allen, T. (1966). "Studies of the Problem-solving Process in Engineering Design." *IEEE Transactions on Engineering Management*. EM-13(2):72-83
- (1977). *Managing the Flow of Technology: Technology Transfer and the Dissemination of Technological Information within the R&D Organization*. MIT Press: Cambridge, Massachusetts
- (2007). "Architecture and Communication among Product Development Engineers." *California Management Review*. 49(2):23-41
- Argyres, N. (1999). "The Impact of Information Technology on Coordination: Evidence from the B-2 'Stealth' Bomber." *Organization Science*. 10(2):162-213
- Armour, H. and D. Teece (1980). "Vertical Integration and Technological Innovation." *The Review of Economics and Statistics*. 62(3):470-474
- Arrow, K. (1974). *The Limits of Organization*. Norton: New York
- Baldwin, C. (2007). "Modularity, Transactions, and the Boundaries of Firms: A Synthesis." *HBS Working Paper 08-013*. Available at: <http://www.hbs.edu/research/pdf/08-013.pdf>
- Baldwin, C. and K. Clark (2000). *Design Rules: The Power of Modularity*. MIT Press: Cambridge, Massachusetts
- Bechky, B. (2003). "Object Lessons: Workplace Artifacts as Representations of Occupational Jurisdiction." *American Journal of Sociology*. 109(3):720-752
- Brooks, F. (1975). *The Mythical Man Month: Essays on Software Engineering*. Addison-Wesley: Reading, Massachusetts
- (1995). *The Mythical Man Month: Essays on Software Engineering, Anniversary ed.* Addison-Wesley: Reading, Massachusetts
- Brunoni, S. and A. Prencipe (2001a). "Managing Knowledge in Loosely Coupled Networks: Exploring the Links Between Product and Knowledge Dynamics." *Journal of Management Studies*. 38(7):1019-1035
- (2001b). "Unpacking the Black Box of Modularity: Technologies, Products and Organizations." *Industrial and Corporate Change*. 10(1):179-205
- Cacciatori, E. and M. Jacobides (2005). "The Dynamic Limits of Specialization: Vertical Integration Reconsidered." *Organization Studies*. 26(12):1851-1883
- Chesbrough, H. and D. Teece (1996). "When is Virtual Virtuous: Organizing for Innovation." *Harvard Business Review*. 74(1):65-74
- and K. Kusunoki (1999). "The Modularity Trap: Innovation, Technology phase-shifts, and the Resulting Limits of Virtual Organizations." In Nonaka, I. and D. Teece (Eds), *Knowledge and the Firm*. Oxford University Press: Oxford
- (2003a). *Open Innovation: The New Imperative for Creating and Profiting from Technology*. Harvard Business School Press: Boston, Massachusetts
- (2003b). "Towards a dynamics of modularity. A cyclical model of technical advance." In Prencipe, A., A. Davies, and M. Hobday (Eds), *The Business of Systems Integration*. Oxford University Press: Oxford
- Child, J. (1977). *Organizations: A Guide to Problems and Practice*. Harper & Row: New York
- Christensen, C., M. Verlinden and G. Westerman (2002). "Disruption, Disintegration and the Dissipation of Differentiability." *Industrial and Corporate Change*. 11(5):955-993
- Clark, K. (1989). "Project Scope and Project Performance: The Effect of Parts Strategy and Supplier Involvement on Product Development." *Management Science*. 35(10):1247-1262
- Clarkson, P., C. Simons, and C. Eckert (2004). "Predicting Change Propagation in Complex Design." *Journal of Mechanical Design*. 126(5):765-797
- Conway, M. (1968). "How Do Committees Invent?" *Datamation*. 14(4):28-31

- Daft, R. and R. Lengel (1984). "Information richness: A New Approach to Managerial Behavior and Organization Design." *Research in Organizational Behavior*. 6:191-233
- (1986). "Organizational Information Requirements, Media Richness and Structural Design." *Management Science*. 32(5):554-571
- Dyer, J. (1996). "Specialized Supplier Networks as a Source of Competitive Advantage: Evidence from the Automobile Industry." *Strategic Management Journal*. 17(4):271-291
- Eppinger, S., D. Whitney, R. Smith, and D. Gebala (1994). "A Model-Based Method for Organizing Tasks in Product Development." *Research in Engineering Design*. 6(1):1-13
- Ernst, D. (2005). "Limits to Modularity: Reflections on Recent Developments in Chip Design." *Industry and Innovation*. 12(3):303-335
- Fine, C. (1998). *Clockspeed: Winning Industry Control in the Age of Temporary Advantage*. Perseus Press: Reading, Massachusetts
- Fixson, S. and J. Park (2007). "The Power of Integrality: Linkages between Product Architecture, Innovation, and Industry Structure." *MIT Sloan Working Paper 4642-07*. Available at: <http://mitsloan.mit.edu/pdf/power-integrality.pdf>
- Galbraith, J. (1974). "Organization Design: An Information Processing View." *Interfaces*. 4(3):28-36
- Grinter, R., J. Herbsleb, and D. Perry (1999). "The Geography of Coordination: Dealing with Distance in R&D Work." In *Proceedings of the International ACM SIGGROUP Conference on Supporting Group Work*, Phoenix, Arizona, pp. 306-315
- Henderson, R. and K. Clark (1990). "Architectural Innovation: The Reconfiguration of Existing Product Technologies and the Failure of Established Firms." *Administrative Science Quarterly*. 35(1):9-30
- Herbsleb, J. and R. Grinter (1999a). "Splitting the Organization and Reintegrating the Code: Conway's Law Revisited." In *Proceedings, International Conference on Software Engineering*, Los Angeles, California, 1999, pp. 85-95
- (1999b). "Architectures, Coordination, and Distance: Conway's Law and Beyond." *IEEE Software*. 16(5):63-70
- Hoetker, G. (2006). "Do Modular Products Lead to Modular Organizations?" *Strategic Management Journal*. 27:501-518
- Jacobides, M. (2005). "Industry Change Through Vertical Disintegration: How and Why Markets Emerged in Mortgage Banking." *Academy of Management Journal*. 48(3):465-498
- Jelinek, M. and C. Schoonhoven (1990). *The Innovation Marathon*. Basil Blackwell: Oxford, UK
- Kellogg, K., W. Orlikowski, and J. Yates (2006). "Life in the Trading Zone: Structuring Coordination Across Boundaries in Postbureaucratic Organizations." *Organization Science*. 17(1):22-44
- Langlois, R. and P. Robertson (1992). "Networks and Innovation in a Modular System: Lessons from the Microcomputer and Stereo Component Industries." *Research Policy*. 21: 297-313
- Loch, C. and C. Terwiesch (1999). "Accelerating the Process of Engineering Change Orders: Capacity and Congestion Effects." *Journal of Product Innovation Management*. 16:145-159
- MacCormack, A., J. Rusnak, and C. Baldwin (2006). "Exploring the Structure of Complex Software Designs: An Empirical Study of Open Source and Proprietary Code." *Management Science*. 52(7):1015-1030
- March, J. and H. Simon (1958). *Organizations*. Wiley: New York, New York
- Mikkola, J. (2006). "Capturing the Degree of Modularity Embedded in Product Architectures." *Journal of Product Innovation Management*. 23:128-146
- Mintzberg, H. (1979). *The Structuring of Organizations*. Prentice-Hall: Englewood Cliffs, New Jersey
- Monteverde, K. (1995). "Technical Dialog As an Incentive for Vertical Integration in the Semiconductor Industry." *Management Science*. 41(10):1624-1638
- Nelson, R. and S. Winter (1982). *An Evolutionary Theory of Economic Change*. Harvard University Press: Cambridge, Massachusetts
- Orton, J. and K. Weick (1990). "Loosely Coupled Systems: A Reconceptualization." *Academy of Management Review*. 15(2):203-223

- Parnas, D. (1971). "Information Distribution Aspects of Design Methodology." In *Proceedings of IFIP Congress*, Ljubljana, Yugoslavia, 1971, Volume 1, pp. 339-344
- (1972). "On the Criteria to Be Used in Decomposing Systems into Modules." *Communications of the ACM*. 15(12):1053-1058
- (1978). "Some Software Engineering Principles." *Infotech State of the Art Report on Structured Analysis and Design, Infotech International*. In Hoffman, D. and D. Weiss (Eds), *Software Fundamentals: Collected Papers by David L. Parnas*. Addison-Wesley: Boston, Massachusetts
- Pich, M., C. Loch, and A. De Meyer (2002). "On Uncertainty, Ambiguity, and Complexity in Project Management." *Management Science*. 48(8):1008-1023
- Rechtin, E. (1991). *Systems Architecting: Creating and Building Complex Systems*. Prentice Hall: Englewood Cliffs, New Jersey
- Robertson, P. and R. Langlois (1995). "Innovation, Networks, and Vertical Integration." *Research Policy*. 24:543-562
- Sako, M. (2004). "Supplier Development at Honda, Nissan and Toyota: Comparative Case Studies of Organizational Capability Development." *Industrial and Corporate Change*. 13(2):281-308
- Sanchez, R. and J. Mahoney (1996). "Modularity, Flexibility, and Knowledge Management in Product and Organization Design." *Strategic Management Journal*. 17:63-76
- Schilling, M. (2000). "Toward a General Modular Systems Theory and its Application to Interfirm Product Modularity." *Academy of Management Review*. 25(2):312-334
- and K. Steensma (2001). "The Use of Modular Organizational Forms: An Industry-level Analysis." *Academy of Management Journal*. 44(6):1149-1168
- Schrader, S., W. Riggs, and R. Smith (1993). "Choice Over Uncertainty and Ambiguity in Technical Problem Solving." *Journal of Engineering and Technical Management*. 10:73-99
- Sharman, D. and A. Yassine (2003). "Characterizing Complex Product Architectures." *Systems Engineering*. 7(1):35-60
- Simon, H. (1962). "The Architecture of Complexity." *Proceedings of the American Philosophical Society*. 106(6):468-482
- (1973). "Applying Information Technology to Organization Design." *Public Administration Review*. 33:268-278
- Sosa, M., S. Eppinger, M. Pich, D. McKendrick, and S. Stout (2002). "Factors that Influence Technical Communication in Distributed Product Development: An Empirical Study in the Telecommunications Industry." *IEEE Transactions on Engineering Management*. 49(1):45-58
- Sosa, M., S. Eppinger and C. Rowles (2004). "The Misalignment of Product Architecture and Organizational Structure in Complex Product Development." *Management Science*. 50(12):1674-1689
- Sosa, M., M. Gargiulo, and C. Rowles (2007). "Component Modularity, Team Network Structure, and the Attention to Technical Interdependencies in Complex Product Development." *INSEAD Working Paper 2007/08/TOM/OB*. Available at: <http://knowledge.insead.edu/doc.cfm?cd=2656>
- Srikanth, K. and P. Puranam (2006). "Tacit Coordination in Distributed Organizations: The Case of Offshored Software Services." Available at: http://papers.ssrn.com/sol3/cf_dev/AbsByAuth.cfm?per_id=414615
- Staudenmayer, N., M. Tripsas and C. Tucci (2005). "Interfirm Modularity and its Implications for Product Development." *Journal of Product Innovation Management*. 22(4):303-321
- Steward, D. (1981a). "The Design Structure System: A Method for Managing the Design of Complex Systems." *IEEE Transactions in Engineering Management*. 28(3):71-84
- (1981b). *Systems Analysis and Management: Structure, Strategy, and Design*. Petrocelli Books: New York
- Sturgeon, T. (2002). "Modular Production Networks: A New American Model of Industrial Organization." *Industrial and Corporate Change*. 11(3):451-496
- Takeishi, A. (2001). "Bridging Inter- and Intra-firm Boundaries: Management of Supplier Involvement in Automobile Product Development." *Strategic Management Journal*. 22:403-433
- Teece, D. (1986). "Profiting from Technological Innovation: Implications for Integration, Collaboration, Licensing and Public Policy." *Research Policy*. 15:285-305

- (1988). “The Nature of the Firm and Technological Change.” In Dosi, G., C. Freeman, R. Nelson, G. Silverberg, and L. Soete (Eds), *Technical Change and Economic Theory*. Pinter: London
- (1992). “Competition, Cooperation, and Innovation: Organizational Arrangements for Regimes of Rapid Technological Progress.” *Journal of Economic Behavior and Organization*. 18:1-25
- (1996). “Firm Organization, Industrial Structure, and Technological Innovation.” *Journal of Economic Behavior and Organization*. 31:193-224
- Terwiesch, C., C. Loch, A. De Meyer (2002). “Exchanging Preliminary Information in Concurrent Engineering: Alternative Coordination Strategies.” *Organization Science*. 13(4):402-419
- Thompson, J. (1967). *Organizations in Action: Social Science Bases of Administrative Theory*. McGraw-Hill: New York
- Tsoukas, H. (1996). “The Firm as a Distributed Knowledge System: A Constructionist Approach.” *Strategic Management Journal*. 17:11-25
- Ulrich, K. (1995). “The Role of Product Architecture in the Manufacturing Firm.” *Research Policy*. 24:419-44
- Ulrich, K. and S. Eppinger (1995). *Product Design and Development*. McGraw-Hill: New York
- Van den Bulte, C. and R. Moenaert (1998). “The Effects of R&D Team Co-location on Communication Patterns Among R&D, Marketing, and Manufacturing.” *Management Science*. 44(11):S1-S18
- von Hippel, E. (1990). “Task Partitioning: An Innovation Process Variable.” *Research Policy*. 19:407-418
- Weick, K. and K. Roberts (1993). “Collective Mind in Organizations: Heedful Interrelating on Flight Decks.” *Administrative Science Quarterly*. 38:357-381
- Wheelright, S. and K. Clark (1992). *Revolutionizing Product Development*. The Free Press: New York
- Williamson, O. (1971). “The Vertical Integration of Production: Market Failure Considerations.” *American Economic Review*. 61(2):112-123
- (1999). “Strategy Research: Governance and Competence Perspectives.” *Strategic Management Journal*. 20:1087-1108