View metadata, citation and similar papers at core.ac.uk

brought to you by CORE

provided by Illinois Digital Environment for Access to Learning and Scholarship Repository

PRODUCTS AND ORGANIZATIONS, MIRRORS IN A FUNHOUSE: THREE ESSAYS ON THE MIRRORING HYPOTHESIS

BY

CHIH I. LIU

DISSERTATION

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Business Administration
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2015

Urbana, Illinois

Doctoral Committee:

Professor Joseph Mahoney, Co-Chair
Associate Professor Douglas Miller, Co-Chair
Associate Professor Janet Bercovitz
Associate Professor Deepak Somaya

ABSTRACT

Much of the contemporary management literature on modularity implicitly assumes that increased product modularity is associated with advantageous increases in organizational modularity. Known as the “mirroring hypothesis,” this postulated relationship is the basis of prescriptions in favor of increasingly modular product design, despite inconclusive empirical evidence. The three essays in this dissertation seek to advance the extant literature on the mirroring hypothesis in the following ways: Essay one presents a systematic review of this fragmented literature. Specifically, this review finds that the extant literature currently contains inconsistent interpretations of modularity, as well as inconsistent conceptualizations of mirroring. Thus, debates on the mirroring hypothesis often amount to unproductive arguments over different things. Essay two puts forth a theoretical framework to explore mirroring between product and organization at the within-firm level. The proposed theory maintains that mirroring is contingent on the level of architectural knowledge, which correlates with a set of observable constructs. Essay three empirically tests the proposition that mirroring between product and organization is also contingent on the demand characteristics of the target customers. Using a sample constructed from the computer systems integration industry, I found empirical support for the proposed demand-side contingencies on mirroring.

Research on the mirroring hypothesis has made important contributions to our understanding of the interactions between technology and organization. However, the extant narratives of mirroring in fact encompass distinct causal mechanisms interacting

across multiple units of analysis. As this dissertation research shows, the structural correspondence between product and organization is like mirrors in a funhouse – there are many mirrors; and many of the mirrors are distorted.

To the memory of my late grandmother, and grandfathers.

ACKNOWLEDGEMENTS

This dissertation would not have been written without the advice and support of my dissertation co-chairs, Joseph Mahoney and Douglas Miller. Joe and Doug show me what it means to be a true scholar, and above all, kind and good person. I also want to acknowledge Janet Bercovitz and Deepak Somaya for serving on my committee and providing constant feedback and encouragements. I also thank Eric Neuman for the support and introduction to the organization theory literature. Diana Gonzalez provided great administrative support, especially during the last year of my enrollment in absentia.

Faculty from other institutions provided invaluable feedback along the way. I would like to thank Sebastian Fixson from Babson College, Alan MacCormack from Harvard Business School, Jane Zhao from University of Kansan, and Jason Woodard from Singapore Management University.

I am also indebted to many fellow graduate students. The friendship and support from Prasanna Karhade and Wooje Cho, especially during the difficult first year, was invaluable. Shawn Riley, Lihong Qian, and Kim Wang helped me transition into a new group. Young-Chul Jeong, Jaegoo Lim, Jun Ho Lee, Jin Uk Kim, Radek Nowak, Yuanyuan Sun, and Gregory Fisher helped me stay connected beyond my focus area. Particularly, the experience with fellow graduate students in the extended strategy group, Joshua Sears, Minyoung Kim, Seth Carnahan, Shweta Gaonkar, Mahka Moeen, Marleen McCormick, John Mawdsley, Kurt Desender, Sandra Corredor, Grace Su, Wenxin Guo, Ji Yoon Chung, and Wen Zheng is an integral part of my time in Illinois.

Graduate students from other institutions also contributed greatly to this endeavor. I would like to thank Doug Hannah from Stanford University, Nahoko Kameo from UCLA, and Kenji Klein from UC Irvine for their friendship and support.

I also want to thank my colleagues at Rochester Institute of Technology for their support in the last phase of the dissertation process. The support from my department chair, Robert Barbato, helped me manage between a full-time appointment and finishing dissertation; my faculty mentor, Clyde Hull, is the best mentor a junior faculty can ever hope for. I look forward to all the great things we will do together.

Finally, I would especially like to acknowledge my family. My father Ming-Chuan and my mother Hui-Ping devoted themselves to everything my brother Chih-Chung and I want to pursue. We are now both Illinois Ph.D. because of their selfless devotion. My extended family also contributes a great deal to our accomplishments. During my father’s recent illness, we are once again reminded how fortunate we are to be a part of this amazing family. I hope this dissertation is just the beginning of what I can accomplish for them.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

The interaction between technology and organization has been a well-established research area in management scholarship (Burns and Stalker, 1961; Woodward, 1965). More recently, management scholars elaborate on this interaction and advance the idea of modularity as a solution to growing complexity in technology and organization (Baldwin and Clark, 1997, 2000; Sanchez, 1995; Sanchez and Mahoney, 1996). Specifically, these scholars put forth the so-called "mirroring hypothesis" between product and organization, which postulates that increased product modularity is associated with advantageous increases in organizational modularity (Colfer and Baldwin, 2010; Hoetker, 2006; Sanchez and Mahoney, 1996). Despite inconclusive empirical evidence (Brusoni, Prencipe, and Pavitt, 2001; Colfer and Baldwin, 2010; Ernst, 2005), much of the subsequent management research on modularity seems to have implicitly accepted the mirroring hypothesis (Hoetker, 2006).

This dissertation seeks to advance our understanding of mirroring between product and organization. In particular, the three studies answer the recent call for a more contingent model that goes beyond asking whether the mirroring hypothesis holds or not, but instead explains and predicts when the mirroring relationship will hold and when it will not (Cabigiosu and Camuffo, 2012; Colfer and Baldwin, 2010; Sanchez and Mahoney, 2013). The following section summarizes the three essays.

1. Systematic Review. Chapter 2 reports the findings of a systematic review of the relevant literatures on the mirroring hypothesis, using methods developed in the evidence-based management literature (Denyer and Neely, 2004; Tranfield, Denyer, and Smart, 2003), which require systematic, transparent, and reproducible criteria to select and synthesize relevant studies. Given that our understanding of mirroring is still incomplete and that extant research has found both great benefits (Baldwin and Clark, 2000; Sanchez and Mahoney, 1996) and risks (Chesbrough and Kusunoki, 2001; Henderson and Clark, 1990) of mirroring, there is a pressing need for a more nuanced synthesis of past research to map out the conceptual landscape of a heterogeneous body of research drawn from multiple literatures.

Specifically, the objectives of the systematic review are to: (1) identify the ways in which definitions of modularity differ, and what these differences mean in terms of mirroring; and (2) contextualize observations of different types of mirroring so that they can be systematically compared and contrasted with one another.

This review identifies four clusters of definitions of modularity in the extant literature on the mirroring. Despite a general consensus on the foundational properties (loose coupling and near decomposability), substantial variations and ambiguity remain in the ways in which modularity is defined and interpreted. In particular, analysis of the findings reveals that many attributes commonly associated with modularity (e.g., reusability, substitutability, component commonality, etc.) in the extant literature actually describe additional dimensions of characteristics beyond the foundational properties of modularity. Thus, one key insight of this review is that not all interpretations of modularity characterize the structural patterns associated with the observations of mirroring between product and organization.

This study also finds that despite a general level of support for the mirroring hypothesis in the extant literature, there is less support for mirroring at higher levels of analysis. Moreover, distinct causal mechanisms (coordination, incentive, and competitive) have been invoked to explain mirroring, suggesting that modularity conveys different meanings or conditions at different levels. Failure to make these distinctions might have contributed to the inconsistent empirical findings. In addition, this review finds significant differences in the support for the mirroring hypothesis across industries. Comparative analysis across industries suggests that mirroring between product and organization depends not only on *how modular* the product is, but also on how the product is *modularized* in the specific technological regimes, i.e., the specific way in which the product is partitioned into stand-alone components.

2. Contingent Model. Chapter 3 puts forth a theoretical framework to explore mirroring between product and organization *at the within-firm level,* which is the level of analysis in the original formulations of the mirroring hypothesis. The proposed theory is based on the premise that implicit assumptions underpinning the prediction of mirroring have been taken for granted and are in fact not always tenable. Specifically, I maintain that the extant literature on the mirroring hypothesis has come to implicitly posit that organizations adopting modular product development possess architectural knowledge of *all possible* component interactions, such that the overall system architecture can be *completely specified* a priori. That is, these organizations are assumed to be capable of specifying component interfaces that *completely* partition product systems into stand-alone modules that only interact through the standardized interfaces.

I submit that these implicit assumptions of completeness should be better understood as theoretical ideals; the extent to which these assumptions are borne out in practice places the boundary conditions on the prediction of mirroring. That is, consistent with Sosa, Eppinger and Rowles (2004), I maintain that no product development organization is omniscient with regard to component interactions and their implications to system behaviors. Even with the best efforts at ex-ante standardization, some component interactions can only be identified during the design process, or even after the product system has been put to use. Architectural knowledge is inevitably incomplete; mirroring is therefore contingent on the level of architectural knowledge. Accordingly, a set of propositions is systematically derived from this baseline contingency to complete the proposed theory framework.

3. Empirical Study. The fourth chapter is an empirical study of the mirroring hypothesis in the computer systems integration industry, which is well-known for its modular product architecture (Baldwin and Clark, 2000; Langlois and Robertson, 1992; Sturgeon, 2002). The dominant computer architecture is the Intel x86 standard[1], with public standards defining all major component interfaces and a well-established

[1] Apple's Macintosh was the remaining major personal computer product line not using the Intel x86 architecture (based on the competing PowerPC architecture) until 2006. Intel x86 architecture has since become the only PC architecture by major PC manufacturers until Samsung's introduction of its Google Chromebook in 2012, which is based on the ARM architecture primarily used in mobile devices. Despite these important competing architectures and their strategic implications, Intel x86 has been the dominant standard throughout the history of personal computing.

commercial-off-the-shelf (COTS) market supplying a large selection of interchangeable parts. In addition, systems integration capability has become so mature and diffused that anyone with some basic skills can easily obtain the needed parts and put together a functioning computer system, which is evidenced by the robust do-it-yourself community. Simply put, not only are all PC components commoditized, systems integration is also characterized by low entry barrier[2].

Based on these industry characteristics, mainstream interpretations of the mirroring hypothesis would predict that systems building firms rely on the COTS market to take advantage of the mature modular architecture. However, many systems building firms deviate from this prediction and continue to adopt more integrated approaches to build computer systems that are fully Intel x86 compatible. The rationales behind this deviation thus shed light on the important contingencies on the mirroring hypothesis. Accordingly, I hypothesize that mirroring between product and organization is contingent on demand characteristics. Specifically, I predict that the extent of mirroring is reduced for companies targeting performance-intensive market tiers, or companies targeting customers who would suffer great economic loss in the event of system failures.

---

[2] My focus here is on the systems building stage only. Systems integration in this stage does not include the design and fabrication of integrated circuits or other upstream activities, which can be highly technical and have high entry barriers.

CHAPTER 2

SYSTEMATIC REVIEW OF THE EXTANT LITERATURE ON THE MIRRORING HYPOTHESIS

2.1 Introduction

Research from a wide range of disciplines posits that a correspondence exists between product and organizational architectures (Argyres and Bigelow, 2010; Baldwin and Clark, 2000; Cabigiosu and Camuffo, 2012; Colfer and Baldwin, 2010; Conway, 1968; Herbsleb and Grinter, 1999; Hoetker, 2006; MacCormack, Baldwin, and Rusnak, 2012; Sanchez and Mahoney, 1996; among others). In particular, much of the contemporary management literature on modularity adopts the assumption that increased product modularity is associated with advantageous increases in organizational modularity (e.g., Baldwin and Clark, 2000; Colfer and Baldwin, 2010; Jacobides, 2005; Langlois, 2002; MacCormack *et al.*, 2012; Sanchez and Mahoney, 1996; Sanchez, 1995, 1999; Sturgeon, 2002).

First articulated in the management literature by Henderson and Clark (1990), Sanchez (1995), and Sanchez and Mahoney (1996), and later named the "***mirroring hypothesis***" by Colfer (2007), this proposed relationship postulates that the structure of a product development organization and the architecture of the product being developed would come to "mirror" each other. However, subsequent empirical research has not consistently found support for this proposed relationship. Specifically, Colfer and Baldwin's (2010) review of 102 empirical studies on the mirroring hypothesis finds that 69% of the studies either provide support or at least find partial or mixed support.

However, their review also shows that the extant literature has yet to converge on a consistent or systematic explanation for the remaining 31% of the studies that find no support or produce contradictory evidence.

Moreover, there remain critical inconsistencies in this literature. Inconsistencies impede cumulative knowledge development and prolong the ongoing debate. Accordingly, the purpose of this essay is to conduct a comprehensive review of the extant literature, so that fundamental inconsistencies can be systematically revealed. More importantly, this essay aims to examine what these inconsistencies mean in terms of our current understanding of mirroring between product and organization.

This essay proceeds as follows. The next section briefly reviews these critical inconsistencies and discusses how the extant narratives of mirroring in fact encompass distinct phenomena. As a result, debates on the empirical status of the mirroring hypothesis often amount to arguments over different things. The next section presents the review method, which attempts to rectify the deficiencies of the review methods traditionally adopted in management research. This is followed by a detailed discussion of the review findings and the conclusion.

2.1.1 Critique of the Extant Literature on the Mirroring Hypothesis

The concept of modularity is central to the research on the mirroring hypothesis. As a general systems concept (Bertalanffy, 1968; Schilling, 2000), modularity has received significant research interests across a wide range of disciplines that deal with complex systems (Baldwin and Clark, 2000; Fixson, 2007b; Schilling, 2002). With its diverse theoretical and practical applications, modularity has a broad-based, intuitive appeal to both researchers and practitioners. However, this widely held intuition belies the underlying complex, multidimensional meanings that have eluded a consistent definition.

Inconsistent meanings of modularity. Scholars have noted that even within the management literature, substantial variations and ambiguities exist in the ways in which the concept of modularity is defined and interpreted (Campagnolo and Camuffo, 2010; Fixson, 2002, 2003, 2007a). Despite an implicit consensus on a general idea, modularity as it is being used really encompasses a bundle of characteristics rather than a single condition (Campagnolo and Camuffo, 2010; Fixson, 2007b; Gershenson, Prasad, and Zhang, 2003, 2004; Sako, 2004). In addition, different viewpoints tend to emphasize different elements (Fixson, 2002, 2003, 2007a), which is reflected in the different definitions and operationalizations. For instance, widely cited definitions for modularity include “the degree to which a system’s components can be separated and recombined” (Schilling, 2000: 312), “high degree of independence or ‘loose coupling’ between component designs by standardizing component interface specifications” (Sanchez and Mahoney, 1996: 65), “interdependence within and independence across modules” through “abstraction, information hiding, and interface” (Baldwin and Clark, 2000: 63), and an architectural type characterized by “a one-to-one mapping from functional elements in the function structure to the physical components of the product” (Ulrich, 1995: 422). Whereas some of these definitions describe the properties and conditions that constitute the concept of modularity, others give more emphasis on the antecedents or design processes that create the desirable properties associated with modularity.

However, it remains unclear whether these definitions are logically equivalent, essentially restatements of the same underpinning ideas, or they reflect real conceptual differences.

Furthermore, the extant literature has yet to clearly delineate modularity with related concepts like commonality (Fixson, 2007b), reusability (Biggerstaff and Richter, 1987; Jones, 1984; Prieto-Diaz, 1993), and substitutability (Garud and Kumaraswamy, 1995). These concepts are often used interchangeably with modularity, even though they actually convey distinct ideas or emphases. Thus, modularity as a theoretical construct still suffers from insufficient construct clarity. More importantly, as a core construct for much of the research literature on the mirroring hypothesis[3], insufficient construct clarity impedes rigorous empirical testing and the development of cumulative knowledge (Suddaby, 2010).

Inconsistent conceptualizations of mirroring. Even though the mirroring hypothesis appears to be a relatively straightforward statement of structural correspondence, closer reading of the relevant literatures suggests that the proposed relationship is conceptualized differently across research streams. In particular, mirroring has been proposed at different levels of analysis.

---

[3] Note that not all of the conceptualizations of the mirroring hypothesis make explicit use of modularity. For instance, following Conway's (1968) conceptualization, Colfer and Baldwin (2010) formally define the mirroring hypothesis in terms of the structural correspondence between two networks - one technical and one organizational, and suggest an ideal test of the hypothesis that examines the correlation between these two networks. The direct examination of the structural correlation thus does not require researchers to assess either product or organizational modularity. This procedure is a stronger test on mirroring since two networks with little structural correspondence can still exhibit similar degree of modularity, resulting in false support for the hypothesis. However, because of the difficulty to directly measure technical and organizational networks except in the context of software development, much of the literature on the mirroring hypothesis relies on the concept of modularity as a system attribute to indirectly assess the extent of mirroring. Inconsistent definitions and operationalizations of modularity would thus further weaken an indirect test.

One stream of studies conceptualizes the mirroring hypothesis as a ***within-firm*** structural correspondence between product and organization (e.g., Cataldo *et al.*, 2006, 2009; Conway, 1968; Herbsleb and Grinter, 1999; MacCormack *et al.*, 2012; Sosa, Eppinger, and Rowles, 2004). The proposed correspondence is thus an isomorphic relationship between the focal firm's organizational structure and the technical architectures of the products being developed (Colfer and Baldwin, 2010). Another stream of studies conceptualizes the mirroring hypothesis as the correspondence between product architecture and the way ***inter-firm*** relationships are organized (e.g., Argyres and Bigelow, 2010; Cabigiosu and Camuffo, 2012; Furlan, Cabigiosu, and Camuffo, 2014; Hoetker, Swaminathan, and Mitchell, 2007; Hoetker, 2006; Mikkola, 2003; Monteverde, 1995). That is, the mirroring hypothesis at the inter-firm level is between the governance of product development tasks *across firm boundaries* and the product system's underlying pattern of technical dependencies. A third stream of studies extends the analysis of inter-firm organization to the ***industry-level***. These studies (e.g., Fixson and Park, 2008; Galvin and Morkel, 2001; Langlois and Robertson, 1992; Lee and Berente, 2012; Linden and Somaya, 2003; Sahaym, Steensma, and Schilling, 2007; Shibata, Yano, and Kodama, 2005) focus on how the overall industry structure co-evolves with the trajectory of product architectures and the underpinning technologies. In this context, the mirroring hypothesis is conceptualized as the correspondence between product architecture and industry organization.

Importantly, while similar narratives of mirroring have been applied to the proposed relationships at different levels, the constructs being mirrored are in fact different across these research streams, which have been developed more or less independently until recently. Furthermore, these research streams invoke different causal mechanisms to explain mirroring.

Specifically, the within-firm studies typically draw on contingency theory (Burns and Stalker, 1961; Chandler, 1962; Drazin and Van de Ven, 1985: 198; Lawrence and Lorsch, 1967; Thompson, 1967; Woodward, 1965) and the information-processing view of organization (Brown and Eisenhardt, 1995; Galbraith, 1974; Tushman and Nadler, 1978) to explain mirroring. These researchers maintain that product modularity economizes on organizational information processing, which is the causal mechanism that brings about mirroring (Conway, 1968; Sanchez and Mahoney, 1996). The inter-firm studies place the analytical focus on product modularity's implications on a firm's boundary choice and the corresponding governance mode. These researchers maintain that product modularity reduces the risk of opportunism by making it easier to switch component suppliers (e.g., Hoetker, 2006) and retain proprietary information (e.g., Argyres and Bigelow, 2010) through information hiding (Baldwin and Clark, 2000; Parnas, 1972). Minimizing transaction costs (Williamson, 1975, 1985) is thus an important mirroring mechanism at the inter-firm level. In the industry-level studies, researchers have begun to recognize the existence of multiple linkages between product architecture and industry structure (e.g., Chesbrough, 2003; Fixson and Park, 2008; Garud and Kumaraswamy, 1995). Technological change and industry organization are connected through the interactions among multiple competitive and institutional mechanisms. This discussion suggests that distinct causal forces have been conflated under the banner of mirroring. There are in fact multiple distinct albeit interrelated mirroring mechanisms. Moreover, how these mechanisms interact remains largely unexplored. The extant literature on mirroring hypothesis is thus more fragmented than it appears.

Leading scholars have called for a more contingent model of mirroring that goes beyond asking whether the mirroring hypothesis holds or not, but instead explains and predicts when the mirroring relationship will hold and when it will not (Cabigiosu and Camuffo, 2012; Colfer and Baldwin, 2010; Sanchez and Mahoney, 2013). However, as an emerging literature that combines ideas from distant disciplines, the extant literature on the mirroring hypothesis still contains numerous critical inconsistencies. These inconsistencies deserve more research attention before useful refinement can be attained.

2.1.2 Review Method

One possible reason that these inconsistencies persist is the way literature reviews have been conducted. Traditionally, management literature reviews are *ad hoc* and narrative-based, which can be lacking in thoroughness and rigor (Tranfield *et al.*, 2003). Particularly, narrative reviews often select articles for inclusion on the implicit biases of the researchers (Fink, 2010; Hart, 1998; Tranfield *et al.*, 2003). Rousseau *et al.* (2008) maintain that traditional management reviews are often position papers, cherry-picking studies to advocate a point of view. As a result, inconsistencies are often left unresolved and the literature remains fragmented. Some review methodologists even suggest that traditional review approach is often biased by factors that would be unforgivable in primary research (Evans and Pearson, 2001; Glass, McGaw, and Smith, 1981) and is therefore not undertaken as a genuine investigatory science (Tranfield *et al.*, 2003).

Advocates of the evidence-based management movement note that their counterparts in medical sciences have steadily improved the quality of review process in the past decades through the development of systematic review methods (Denyer and Neely, 2004; Tranfield *et al.*, 2003; Whittemore and Knafl, 2005). These methods attempt to address the 'deficiencies of traditional review methods by applying the same standards to secondary research (where the unit of analysis is other research studies) as should be applied to primary research' (Davies and Crombie, 1998: 2). The aim is to achieve reliable research synthesis, which is the "systematic accumulation, analysis and reflective interpretation of the full body of relevant empirical evidence related to a question" (Rousseau *et al.*, 2008: 475).

Even though fundamental differences in the ontological and epistemological assumptions between medical sciences and management prevent the direct transplant of all systematic review methods (e.g., methods designed to synthesize findings from randomized controlled experiments might not apply easily to most management research[4]), Tranfield *et al.* (2003) propose that specific principles can still be usefully applied to management research to counteract bias by making explicit the values and assumptions underpinning a review. Specifically, they state that systematic review in management should adhere to the following guidelines: (1) include explicit development of clear and precise review objectives, (2) follow review protocol clearly defined at the outset, (3) conduct comprehensive search for all potentially relevant articles, (4) use

[4] Meta-analysis is considered a particular type of systematic review method that combines data from multiple primary studies by employing statistical methods (Glass, 1976). However, this method requires the research design and hypotheses of primary studies to be very similar, if not identical (Cooper, 1998), which is not always the case for management research. This method is not applicable when studies to be reviewed include both quantitative and qualitative studies. It is therefore not a suitable review method for the literature on the mirroring hypothesis.

explicit and reproducible criteria for the inclusion and exclusion of articles, and (5) apply an explicit analytic framework to synthesize individual studies. These guidelines help ensure the review process is carried out in a systematic, transparent, and reproducible manner.

Following the review guidelines suggested by Tranfield *et al.* (2003), Denyer and Neely (2004), and Whittemore and Knafl (2005), and the examples provided by a number of exemplary systematic reviews in the management literature (e.g., Keupp, Palmié, and Gassmann, 2012; Nijmeijer, Fabbricotti, and Huijsman, 2013; Pittaway *et al.*, 2004; Ravasi and Stigliani, 2012), I formulate a multi-stage review protocol for this essay. The following subsections describe the review protocol in detail.

Review objectives. One central thesis of an emerging revisionist perspective is that the mainstream modularity literature has the tendency to put forth generalized prescriptions of mirroring between product and organization based on empirical observations that are more or less context-specific (Ernst, 2005). Given that there remain substantial inconsistencies in how mirroring is conceptualized and that there are in fact multiple mirroring mechanisms, simple prescriptions in favor of a generalized notion of mirroring are unlikely to be specific enough to inform practice. Moreover, the danger of strong mirroring is also illustrated by Henderson and Clark's (1990) well known observation that incumbents often suffer sharp decline in the face of architectural innovations because their architectural knowledge tends to become embedded in organizational structure, i.e., some instances of mirroring can lead to 'modularity trap' (Chesbrough and Kusunoki, 2001) and have disastrous consequences. Furthermore, the substantial variations and ambiguities in the ways in which modularity is interpreted suggest that we might not yet fully know what mirroring means exactly. Given that our understanding of mirroring is still incomplete and that extant research has found both great benefits and danger of mirroring, there is a pressing need for a more nuanced synthesis of past research to map out the conceptual landscape of a heterogeneous body of research. Specifically, the objectives of the review are to:

1. Identify the ways in which definitions of modularity differ, and what these differences mean in terms of mirroring; and
2. Contextualize observations of different types of mirroring so that they can be systematically compared and contrasted with one another.

Search strategy. One key principle of systematic review is the construction of an unbiased evidence base, which requires using comprehensive and protocol-driven search strategy to identify potentially relevant studies. Ideally, all articles relevant to the review objectives should be included; however, identifying all relevant studies can be challenging. The fact that the mirroring hypothesis literature combines independently developed research streams from distant disciplines further complicates this search challenge. In order to use an electronic citation indexing service, this review is limited to journal articles, omitting books, book chapters, and working papers. The search strategy follows a multi-stage process:

1. Starting from Colfer and Baldwin's (2010)[5] selection of 19 journals, and informed by the prior knowledge that the extant literature draws on research from three disciplines (i.e., management, software engineering, and product

[5] To the best of my knowledge, Colfer and Baldwin (2010) is the only review article that considers all three relevant literature streams. The expanded list of 32 journals is also reviewed by two senior management scholars.

development), I select 32 high-impact journals to represent these three literatures. These journals are listed in Table 2.1.

2. To increase comprehensiveness and ensure unbiased search, the keyword search string was developed iteratively in the following manner:
   First, I identify all the relevant articles referenced[6] by Colfer and Baldwin's (2010) with more than 100 citations as reported by Google Scholar. This step produces 16 seminal articles representative of the literature as the search "target."

   Second, using Thomson Reuters' *Web of Science* indexing service, I iteratively expand Colfer and Baldwin's (2010) search keywords until all 16 seminal articles are among the articles in the search result. The iterative expansion of the search string is also informed by prior knowledge of the literature. For instance, in the software engineering literature, the mirroring hypothesis is known as "Conway's Law" (Conway, 1968) or "socio-technical congruence" (Cataldo, Herbsleb, and Carley, 2008). These terms are added to the search string to increase comprehensiveness.

   I also compare the search results produced by each incremental search string expansion to make sure the additional search terms actually identify more relevant articles. This extra step ensures that all the search terms are theoretically meaningful. Overall, 797 potentially relevant articles are identified using the expanded keyword search string. The fact that all 16 seminal articles are identified by a theoretically meaningful keyword search string gives high confidence of the comprehensiveness of this search procedure. The final keyword search string is presented in Table 2.2.

3. I read the abstracts of the 797 articles and retain only those articles meeting the inclusion criteria discussed below. This step produces 113 potentially relevant articles. I then read the 113 articles in detail. 9 articles are removed at this step for

[6] Colfer and Baldwin's (2010) review only considers empirical studies. Since this current review intends to cover both empirical and non-empirical studies, I reviewed and identified all the relevant non-empirical articles referenced by Colfer and Baldwin (2010) and added them to the citation count list as well. Articles referenced that do not pertain to the mirroring hypothesis were excluded. The aim is to construct of list of seminal articles to represent the literature.

failing to meet the inclusion criteria. This search procedure thus identifies 104 relevant articles to review. The full list of the articles is reported in the Appendix. I should also note that of the 113 articles, 63 are articles I was unaware of before this review effort. The identification of a large number of unfamiliar articles gives confidence that the search procedure is indeed unbiased.

Inclusion criteria. This essay reviews both empirical and conceptual articles since the review objectives include identifying definitional differences in the literature. Specifically, to be included in this review, the article should (1) investigate ***both*** product architecture and organization structure at within-firm, inter-firm, or industry-level; (2) report empirical analysis or present conceptual elaboration that either directly addresses the mechanism(s) linking product and organization or addresses consequences of the linkage(s) between product and organization from which mirroring can be indirectly inferred. These conditions are intentionally inclusive to serve the review objective of mapping out the conceptual landscape of a fragmented literature. For the same reason, this review does not perform stringent quality evaluation to exclude studies the way meta-analytic studies typically do.

Analytic framework. The review objectives stated at the outset of a systematic review directly informs the construction of the analytic framework. The analytic framework is an integral element of a review protocol since it describes and structures the coding procedure and the analysis of the included articles. In order to achieve the review objectives stated above, the included articles in this essay are coded and analyzed along the following four dimensions:

1. *Context of Mirroring:*

- At what level(s) of organization is the mirroring relationship examined or theorized?
- What is the industry context of the study?
- What is the level of dynamism of the prevailing product architecture - stable, evolving, or changing radically?

2. *Conceptualization of Modularity:*
    - What is the conceptual definition for modularity cited in the article?
    - How is product modularity operationalized?
    - How is organizational modularity operationalized?
    - What life-cycle dimension is product modularity observed in (i.e., modularity in design, manufacturing, or in use)?
3. *Mechanism(s) of Mirroring:*
    - What is the theoretical explanation given in support of (or against) the mirroring hypothesis?
    - What is the direction of causality?
4. *Findings:*
    - (For empirical studies) What type of evidence is provided in support of (or against) the mirroring hypothesis?
    - What are the findings of the study (i.e., direct support, indirect support, contingent support, or contradict the mirroring hypothesis[7])?

Note that some of the coding categories do not apply to all included articles. For instance, studies from the software engineering literature typically do not measure product and organizational modularity to study mirroring; instead, structural correlation between software product and organization structure is directly measured. It is also

[7] An article is coded as providing direct support if the empirical evidence directly demonstrates mirroring between product and organization or if the conceptual discussion concludes in direct support for the hypothesis. An article is coded as providing indirect support if the empirical evidence or the theoretical claims can be used to indirectly infer support for the mirroring hypothesis, e.g., observing performance advantage for mirroring even though the sample contains a roughly equal mix of observations of mirroring and lack of mirroring. An article is coded as providing contingent support if the findings clearly identify condition(s) under which mirroring is supported.

important to note that the coding procedure is not intended to produce statistical synthesis of past research.

2.2 The Evidence Base

Applying the analytic framework to the 104 included articles produces the evidence base for this systematic review. The following section presents an overview of the results. Detailed analysis and discussion follow.

2.2.1 Overview of Results

Among the 104 included studies, 43 studies examine within-firm mirroring, 39 studies examine inter-firm mirroring, and 32 studies examine industry-level mirroring[8]. In total, 34 within-firm studies (79.07%), 31 inter-firm studies (79.49%), and 23 industry-level studies (71.88%) conclude with direct, indirect, or contingent support for the mirroring hypothesis. This finding is consistent with the current consensus that there is a general level of support for the mirroring hypothesis in the extant literature. This finding also shows that there is less support in the current literature for mirroring at the industry-level. In addition, counting only those studies that clearly identify contingencies under which the mirroring hypothesis is supported results in 7 within-firm studies (16.28%), 8 inter-firm studies (20.1%), and 8 industry-level studies (25.00%). 3 within-firm studies (6.98%), 7 inter-firm studies (17.95%), and 7 industry-level studies (21.88%) produce findings that contradict the mirroring hypothesis. Thus, there is more evidence that

[8] These counts add up to 114, which is greater than the number of included articles. This is because some articles examine mirroring at more than one level.

mirroring at higher level involves more factors that impinge on the mirroring process. These findings are summarized in Figure 2.1.

Importantly, these numbers should not be interpreted as results of aggregated statistical tests on the mirroring hypothesis. No attempt is made to combine statistical observations across individual studies. These numbers only represent a simple count of the various judgments found in the current literature. Overall, the evidence base indicates a general level of support for the mirroring hypothesis, although it also suggests that mirroring involves more than one simple linkage between product and organization across all levels. A more nuanced understanding of mirroring is thus needed. Moreover, the evidence base also reveals that cross-over across disciplines is limited. Except the few seminal articles that are sometimes cited across disciplinary boundary, e.g., Baldwin and Clark (2000) and Sanchez and Mahoney (1996), researchers seldom make reference to studies outside of their disciplines. Consequently, research on the mirroring hypothesis is on divergent paths across disciplines, with different literatures adopting distinct definitions and focusing primarily on different units of analysis. As a first step toward synthesis, the next section addresses the first review objective by identifying the ways in which definitions of modularity in the extant literature differ, and what these differences mean in terms of mirroring. The section that follows addresses the second review objective by contextualizing observations of mirroring across different literatures so that the evidence base can be better synthesized.

2.2.2 Meanings of Modularity

Modularity as it is being used encompasses a plurality of meanings. To assess the ways in which interpretations of modularity differ in the extant literature, the included articles are carefully analyzed to extract the contents and sources of the definitions provided. For those articles that do not provide explicit definitions of modularity, empirical operationalizations are extracted instead. Definitional elements are then reconstructed to capture the intended meanings reflected in the operationalizations.

The extracted contents appear to be similar or even logically equivalent at first, but subtle differences emerge upon more reflection. To make sense of these differences, the extracted contents are broken down into their elements and categorized. These definitional elements can be categorized into three types[9] - (1) description of the defining attributes of modularity (e.g., loose coupling, one-to-one function-component mapping), (2) description of the process through which the properties of modularity are created (e.g., information hiding, interface specification), and (3) heuristics that can be applied to assess the degree of modularity[10] (e.g., the extent of interface standardization). Along with citation information, the extracted definitions are thematically analyzed and clustered into four groups of similar meanings and origins.

1. Modularity as loose coupling or near decomposability. The first group of definitions focuses on the defining attributes of modularity. Drawing on seminal works on complex systems, these definitions describe modularity as a special form of design which exhibits *loose coupling* (Orton and Weick, 1990; Weick, 1976), *near*

---

[9] Many extracted definitions include more than one type of definitional elements.

[10] These elements are therefore more operational than conceptual. Viswanathan (2005) points out that the level of abstraction is an important consideration in the development of theoretical constructs. Constructs that are too concrete might not be useful for theoretical generalization, even though their measurement can be more direct. Therefore, these heuristic definitional elements might only reveal specific aspects of the multidimensional meanings of modularity.

decomposability (Simon, 1962), or creates system decomposition into *parsimoniously linked subsystems* (Alexander, 1964). By and large, these definitions consider modularity as synonymous with loose coupling and near decomposability, which constitute the foundational properties of modularity[11]. However, these foundational properties might be too abstract to tell us much about the tangible structural characteristics associated with modular systems. In fact, Simon, (1962) comments on the observed ubiquity of modularity and suggests that perhaps the perceptual and analytical limits of human cognition prevent us from ever observing the inner structures of non-modular complex systems[12]. In this sense, modularity might be *perceptually* ubiquitous as it is a reflection of how human cognition works (Callebaut, 2005). This philosophical puzzle notwithstanding, clearly there are observable hence definable attributes that could characterize different degrees of modularity.

To bring the conceptualization down to a more concrete level, some definitions in this group describe more measurable characteristics directly derived from the foundational properties. For instance, MacCormack and coauthors (2006, 2012) conceptualize and measure modularity in product design by the potential for a design change in one component to propagate to other components. Modularity is thus defined

---

[11] Much of the modularity literature treats loose coupling and near decomposability as synonyms. Notably, Orton and Weick (1990) make a distinction between a dialectical interpretation and a unidimensional interpretation of loose coupling. According to this distinction, near decomposability is a unidimensional interpretation of loose coupling. Except when this distinction is explicitly invoked, this review follows the rest of the literature and uses these two terms interchangeably.

[12] Simon (1962: 478) concedes that this is a chicken and egg problem that he shall not try to settle: "whether we are able to understand the world because it is hierarchic or whether it appears hierarchic because those aspects of it which are not elude our understanding and observation. I have already given some reasons for supposing that the former is at least half the truth - that evolving complexity would tend to be hierarchic - but it may not be the whole truth."

as the reverse of the potential for design change propagation. This characterization of product modularity is logically consistent with loose coupling and near decomposability, but grounded in the more tangible context of product design.

Other definitions in this group also describe various desirable properties that are frequently associated (and often used interchangeably) with modularity. For instance, a common survey item for product modularity asks the respondents to assess the extent of component reusability, i.e., using the same component design *across* different product systems (e.g., Bush, Tiwana, and Rai, 2010; Worren, Moore, and Cardona, 2002). However, component reusability is not logically equivalent to the foundational properties. In addition to the foundational properties, component reusability requires compatibility across product systems, which imposes additional design constraints on all components intended to be reused[13]. Therefore, near decomposability is a necessary but not sufficient condition for component reusability. Despite the common association in the literature, not all modular components are reusable; but all reusable components are modular. Attributes that are similarly associated with modularity include component substitutability, i.e., the ability to use a variety of qualified components to satisfy a system function (e.g., Garud and Kumaraswamy, 1995), and "mixing and matching" of components, i.e., the ability to combine qualified components in various ways to compose different system architectures (e.g., Bush *et al.*, 2010; Schilling, 2000). These characterizations are often implicitly and erroneously assumed to be equivalent to loose coupling or near decomposability. These findings can be summarized as follows:

---

[13] Essentially, component reusability requires standardization. However, standardization has its own issues as a defining attribute of modularity as discussed in later section.

Finding 1: Loose coupling / near decomposability is a necessary but not sufficient condition for component reusability, component substitutability, and "mixing and matching" of components[14].

These nuanced distinctions have important implications in terms of the reliability of measurement. For instance, the survey instrument in Worren *et al.* (2002) has four items for product modularity, with $\alpha = 0.64$. The item *component reusability* has a factor loading of 0.82; the item *degree of component carry-over* has a factor loading of 0.74; however, the item *we can make changes in key components without redesigning others,* arguably the more essential attribute of modularity in terms of loose coupling and mirroring, only has a factor loading of 0.39. The remaining item, *our products have been decomposed into separate modules,* has a factor loading of 0.51. These factor loadings indicate that what practitioners commonly understand as modular in specific contexts is not always consistent with the specific interpretations of modularity associated with mirroring between product and organization.

In sum, this discussion shows that there are nuanced differences among different interpretations of modularity. Beyond the broad consensus on loose coupling and near decomposability, we do not (and perhaps will not) have complete agreement on the precise defining attributes of modularity. Modularity as it is being used is not just a multidimensional concept encapsulating a bundle of characteristics; the exact meanings also vary, in subtle but consequential ways, across the extant literature. This plurality of

[14] Reusability, substitutability, and mixing and matching are highly interrelated but nonetheless distinct properties. Reusability refers to reusing the same component design across multiple product systems, whereas substitutability refers to substituting different components for a given product system. Mixing and matching requires both reusability and substitutability. The underpinning condition is the varying ***scope*** of compatibility of the standardized interface specifications.

meanings, along with the conflation of distinct mirroring mechanisms, reflects our current incomplete knowledge on mirroring between product and organization.

2. Modularity as interface standardization. The second group of definitions is an influential variant of the first group. Building on the same foundational properties of loose coupling and near decomposability, Sanchez and Mahoney (1996) expand the definition of modularity to include standardization of component interface specification. Standardized component interfaces, they maintain, help create loosely coupled component designs because these components can be developed relatively independently so long as they conform to the standardized component interface specifications. Standardized interfaces thus play an important coordinating role for mirroring between product and organization. Similarly, Baldwin and Clark (2000) state that modularization requires product designers to partition design parameters into two categories: visible information that specifies how potential interactions across modules will be handled, and hidden information that only affects the functioning of the local modules. This segregation enables loose coupling and must be maintained throughout the development process, i.e., it must be standardized. Importantly, this conception of modularity begins to bring attention to the process that creates the defining properties of modularity, thus rendering more concrete and operational definitions.

Subsequently, many researchers adopt this emphasis on interface standardization as a defining attribute of modularity (e.g., Argyres and Bigelow, 2010; Jacobs *et al.*, 2011; Tiwana, 2008a, 2008b). As a result, the extent of interface standardization becomes a common measure of modularity. However, a growing body of empirical findings shows that interface standardization alone does not create loosely coupled components

(Cabigiosu, Zirpoli, and Camuffo, 2013; Chuma, 2006; Kotabe, Parente, and Murray, 2007; Staudenmayer, Tripsas, and Tucci, 2005; Takeishi, 2002; Tidd, 1995). In particular, Staudenmayer *et al.* report:

> While an environment of interfirm modularity should in theory eliminate interdependencies among firms since interfaces between products are defined ex-ante, the present study found, ironically, that interdependencies were ubiquitous. Interdependencies continually emerged throughout the product development process, despite efforts to limit them (2005: 303).

In fact, a more careful reading of Sanchez and Mahoney (1996) and Baldwin and Clark (2000) would reveal that these works do not suggest that interface standardization alone creates loosely coupled component design and organization. Specifically, Sanchez and Mahoney assert that:

> To fully specify component interfaces in a modular product architecture, a firm must have, or have access to, advanced *architectural* knowledge about relevant components and their interactions (1996: 70, emphasis in the original).

Baldwin and Clark similarly state:

> As the properties of the system and the modules become better understood, the design rules will tend to become more complete. Then, as more of the innate interdependencies come to be addressed in the design rules, integration and testing of the system will become more cut-and-dried. Eventually, this part of the process may become so standardized and so simple that users themselves can take over the tasks of integrating and testing their own systems (2000: 77).

These passages clearly suggest that interface standardization not informed by adequate knowledge on component interactions would not reduce component coupling. To illustrate this idea, consider the two examples of task partitioning provided by von Hippel

(1990: 440). The first partitioning has firm X responsible for the design of the aircraft body, and firm Y responsible for the design of the engine. The second partitioning has firm X responsible for designing the front half of the aircraft body and engine, and firm Y responsible for the back half of each. Clearly, standardizing the task interface in the second partitioning would not reduce the needed interactions between firm X and Y.

Importantly, standardization of component interfaces creates the necessary information stability needed to decouple component designs and reduce organizational interdependencies (Sanchez and Mahoney, 1996); however, subsequent research has since forgotten the important premise of having advanced architectural knowledge about relevant component interactions. Not all interface standardizations achieve loosely coupled product and organization designs. Therefore, this review also finds:

> *Finding 2: Standardization of component interface specifications is a necessary but not sufficient condition for loosely coupled product and organization designs. Standardized interfaces must be informed by adequate architectural knowledge about relevant components and their interactions in order to achieve loose coupling.*

This finding can be stated alternatively as

> *Corollary: Incomplete knowledge of component interactions reduces the extent of mirroring between product and organization.*

Moreover, Chuma (2006) points out that there is a conceptual distinction between "perfect modularity"[15] à la Sanchez and Mahoney (1996) and Baldwin and Clark (2000) vis-à-vis what he labels "interim modularity," which is the interface specification adopted during trial-and-error development or prototyping phases as a tool to orchestrate

[15] Chuma (2006) uses “perfect modularity” and “ex ante modularity” interchangeably.

dispersed specialists to collectively learn about the "innate interdependencies" and emerging system architecture. Chuma (2006) states that the primary purpose of interim modularity is the facilitation of collective sensemaking, which is considered as "another power of modularity." Thus, the presence of codified interface specifications does not necessarily indicate the presence of adequate knowledge about component interactions to achieve loose coupling. In fact, interim modularity is frequently associated with reduced organizational modularity since collective learning would require more (not less) coordination (Ernst, 2005). Chuma's (2006) conceptual distinction thus helps resolve an important contradiction in the literature. Accordingly, measuring modularity by assessing the extent of interface standardization alone is likely to introduce considerable measurement error in terms of testing the mirroring relationship between product and organization.

3. Modularity as one-to-one function-component mapping. The third group of definitions of modularity comes from research that focuses on product design and development, which imports a strong *functional* emphasis and much of the meanings of modularity from the engineering design literature (e.g., Cross, 2008; Pahl, Beitz, and Wallace, 1996: 1; Suh, 1990, 2001, 2005). Functions rightly play a central role in this literature since they are the intended reasons behind the existence of all engineering products (Hirtz *et al.*, 2002; Taura and Yoshikawa, 1994). These scholars conceptualize product design as an iterative mapping process that “zigzags” between the functional and physical domains (Clark, 1985; Fixson, 2002; Suh, 1990). Accordingly, this group of definitions conceptualizes product architecture as the scheme by which the functions of a product system are allocated to physical components; and modular architecture as characterized by a *one-to-one mapping* from functional elements to physical components (Ulrich, 1995)[16]. In similar terms, Ulrich (1994: 220) posits that modularity encapsulates two characteristics: (1) similarity between the *physical* and *functional* architectures of the design; and (2) minimization of *incidental interactions* among physical components.

This functional emphasis is distinct from the typical definitional emphases in the mainstream management literature on modularity. Management scholars are primarily concerned with the organizing principles that effectively govern organizational activities. They find product functions to be "inherently manifold and nonstationary" (Baldwin and Clark, 2000: 63) in the contexts of organizational phenomena. Since coordinating interdependent activities stemming from all relevant sources is arguably the most fundamental task of management, management scholars prefer definitions of modularity based on a *generalized* notion of interdependencies (i.e., loose coupling, near decomposability, and their various derivatives). Generalizing interdependencies means considering interdependencies of different kinds as instances of a generic notion of interdependencies, which provides a conceptually stationary unit of analysis to study the structural features and organizing principles that help manage interdependent organizational activities (e.g., Thompson, 1967).

However, the conceptual relationships between definitions with these distinct emphases remain unestablished. Specifically, we need to establish whether Ulrich's (1994) characteristics of modularity are logically compatible with the foundational

---

[16] Note that Ulrich (1995) also states that modular architecture needs to specify decoupled interfaces between components. It is therefore a two-part definition in terms of both elements and interfaces.

properties of loose coupling and near decomposability, or whether they actually reflect distinct interpretations.

Physical and functional similarity. Functional decomposition is the process of successively breaking down a complex problem into its constituent parts by resolving the functional relationships among them[17]. It is indispensable to product development (Pahl *et al.*, 1996; Taura and Yoshikawa, 1994) as a "divide and conquer" method that reduces the scope and complexity of individual design tasks so that they can be feasibly assigned to boundedly rational design teams. Moreover, functional decomposition also depicts the necessary component interactions (i.e., the functional relationships) that contribute to the overall system functionality. Therefore, a product system can be described parsimoniously by a collection of functional elements and the way these elements are linked together (Ulrich 1994). In Sanchez and Mahoney's (1996) language, functional decomposition generates the component interface specifications. If the physical architecture of a product system is designed to mirror functional decomposition, we would know that at least the *intended* interactions among the physical components consist of only those interactions critical to the system functionality. Thus, we can say that such a product design is *intended* to be loosely coupled or nearly decomposable. In other words, functional decomposition is an important first step in the process of modularization.

Minimization of incidental interactions. However, even if the physical architecture of a product system corresponds to its functional architecture in a one-to-one

---

[17] Note that a product system can be functionally decomposed in many different ways, which potentially can produce different system qualities. Functional decomposition is thus a part of *conceptual design* process and is often informed by extant architectural knowledge. Product design is not confined to the physical domain only.

manner, in reality, there might still be many insidious, *incidental* interactions among the physical components. These interactions could remain unknown and unnoticed for much (or all) of the product development (Sosa *et al.*, 2004). Ulrich (1994) gives the example of common automobile design in which the heat produced by the engine will influence the seal on the water pump in a way entirely incidental to the functions of either the engine or the water pump. Elimination of these incidental interactions thus makes the product system more loosely coupled or nearly decomposable. It also helps ensure the integrated product system actually performs as intended. Therefore, finding and eliminating or otherwise preventing incidental interactions is another important step in the process of modularization.

This discussion illustrates that the definition of modularity as one-to-one function-component mapping is logically consistent with the foundational properties. Moreover, this definition is also compatible with the definition of modularity as interface standardization. In addition, adopting the functional perspective provides a number of nuanced insights to our understanding of modularity and mirroring.

First, adopting the functional perspective provides a framework that explicitly incorporates design intentions into the analysis of product development. Such a framework also helps make clear the distinction between the intended and the unintended, incidental component interactions. This distinction suggests that our conception of architectural knowledge should be expanded to include not only knowledge about relevant components and their designed interactions, but also about the undesirable and unintended component interactions to be eliminated or prevented. Therefore, in practice, architectural knowledge actually contains more contents than our literature currently recognizes. In addition, unintended component interactions are difficult to uncover by loosely coupled organizational units focused on their own component development. Some level of organizational integration is required in order to carry out the joint discovery of unintended component interactions. As a result, how a product development organization uncovers and eliminates incidental component interactions would have significant implications on the extent of mirroring between product and organization.

Second, defining modularity from the functional perspective also brings more clarity to the complex process of modularization. Particularly, Ulrich's (1994, 1995) two-part definition suggests that distinct design activities are carried out to achieve specific attributes of modularity. Importantly, achieving one modular attribute does not necessarily achieve others. Sako (2004) therefore asserts that modularity is a multidimensional bundle of attributes that is difficult to rank along a unidimensional modular-integral spectrum. To illustrate this assertion, Sako asks:

> [I]s a product architecture with a one-to-one mapping from functions to components but without decoupled interfaces more or less modular than a product architecture with a many-to-one mapping from functions to components with decoupled interfaces? (ibid., p. 232)

This line of reasoning reveals that defining modularity based on a generalized notion of interdependencies might have flattened and obscured the multidimensionality of the underpinning constitutive processes. As it turns out, this issue on dimensionality, which will be discussed in detail in the next subsection, has important implications on our understanding of modularity and mirroring.

4. Modularity as reduction in post-assembly system fine-tuning. The fourth group of definitions identified in this review is a small but interesting set of interpretations worthy of more attention, since it reveals important but little-known distinctions. The idea, originated in (Fujimoto, 2001) and elaborated further by Chuma (2006), is that the need to perform post-assembly system fine-tuning is a defining characteristic of an integral architecture because system fine-tuning entails *mutual adjustments* of component configurations, which therefore requires knowledge beyond component boundary. According to this interpretation, the defining characteristic of a modular architecture is that the total performance of the assembled product system can be automatically guaranteed if the specified performance of the constituent components is assured ex ante (ibid.). Therefore, assembly of modular systems should be a simple matter of putting the modules together. Modular components are supposed to just "plug and play." In other words, modular systems are those that have found ways of evading the nonlinearities of complexity that could arise out of unpredictable or reciprocal component interactions (Langlois, 2006).

This interpretation of modularity is a significant departure from the mainstream view. In particular, this interpretation explicitly assumes the perspective of a systems integrator. Modularity is conceptualized as the extent to which a component can be treated as a "black box" in the context of systems integration. For example, PC assembly does not involve much more than physically connecting the commodity components. The more modular a component, the less component knowledge is required of the systems integrators to incorporate such a component into the overall system.

Notably, this alternative interpretation of modularity would consider aircraft, automobile, and computer operating systems integral systems (Chuma, 2006: 405), whereas mainstream interpretations would consider these products modular systems (Sanchez and Mahoney, 1996: 67). The point of departure between these interpretations is that while many components in these product systems can be developed more or less independently following standardized component interface specifications (i.e., modular component design), substantial post-assembly system fine-tuning is still required in order to achieve the desired system-level performance. Systems integrators thus play a critical role in the production of these product systems.

Importantly, the distinction between system fine-tuning and interdependent component designs is a subtle but critical one. Specifically, *system fine-tuning does not change the designs of the components involved;* it only changes the *configurations* of these components within the range of component variation that the standardized interface specifications allow. For instance, fine-tuning an automobile might involve mutual adjustments of the carburetor and the ignition system to produce the desired performance profiles. However, it does not involve a redesign of either the carburetor or the ignition system. This subtle distinction has an important implication - namely, even for the very same product system, *product modularity as experienced by component developers and product modularity as experienced by systems integrators need not always be the same.*

This claim might appear to be somewhat counter-intuitive. On the surface, both system fine-tuning and interdependent component design reflect interdependencies across components. However, not all interdependencies are of the same kind. Interdependencies exist in all dimensions of a complex system. The dimensions of interdependencies most relevant to component developers might not always be the same as those most relevant to systems integrators.

Specifically, in addition to functional interdependencies, systems integration also depends critically on knowledge about *performance interdependencies* across components. Certainly, functional and performance dimensions are not orthogonal. Critical performance interdependencies are often captured as part of the functional requirements. However, there is no reason to presume that the patterns of interdependencies in these dimensions should be identical. This observation is consistent with the main finding of the systems integration literature that integrators often possess knowledge in excess of what they need for what they make because they need to account for additional dimensions of interdependencies (Brusoni *et al.*, 2001; Brusoni and Prencipe, 2001a). It is also consistent with the life-cycle perspective of modularity (Campagnolo and Camuffo, 2010; Fixson, 2003; Gershenson, Prasad, and Allamneni, 1999), which emphasizes the changing modularization objectives throughout a product's development life-cycle phases (design for manufacturing, design for logistics, design for time-to-market, etc.). In fact, a complex system can simultaneously exhibit different patterns of interdependencies in different dimensions. (For an illustration of this claim, consider figure 2.2.) Since modularity is a concept about the pattern of cross-component interdependencies, I find

> *Finding 3: A complex system can simultaneously exhibit different degrees of modularity in different dimensions. Therefore, a unidimensional conception of modularity based on a generalized notion of interdependencies underspecifies the underlying multidimensional structure of interdependencies.*

Recognizing the multidimensionality of interdependencies requires us to rethink how we formulate the mirroring hypothesis. If a complex system can simultaneously exhibit different degrees of modularity in different dimensions, what really constitutes mirroring or lack of mirroring between product and organization? For instance, consider Henry Ford's introduction of assembly line manufacturing to automobile production. Product architecture was more or less the same after the adoption of assembly line manufacturing; however, organizational structure was radically transformed. In this case, the structure of functional interdependencies remained stable; the structure of production process interdependencies became much more modular. Mirroring is thus between production process and organizational structure. This example illustrates that we need a more nuanced statement of mirroring beyond structural correspondence between product and organization since product architecture only embodies a subset of interdependencies that impinge on organizational design.

Summary. This review identifies four clusters of definitions of modularity in the extant literature. Despite a general consensus on the foundational properties of loose coupling and near decomposability, substantial variations and ambiguity remain in the ways in which modularity is defined and interpreted. In particular, this analysis reveals that many attributes commonly associated with modularity in the literature actually describe additional dimensions of characteristics beyond the foundational properties (e.g., reusability, substitutability, etc.). Moreover, these attributes call for distinct design considerations, which hold different implications on the needed organizational coordination to carry out the distinct design tasks. Therefore, one key finding of this review is that *not all interpretations of modularity characterize the structural patterns* *associated with the observations of mirroring between product and organization.* Mirroring thus needs a more precise specification.

One important reason for the imprecision in the current specification is the literature's tendency to drift toward unidimensional interpretations of complex constructs (Orton and Weick, 1990). Such interpretations of modularity generalize interdependencies and collapse their dimensionality, which obscures the salient structural features upon which various mirroring forces operate and consequently weakens the explanatory value of modularity. To address this deficiency, the next section attends to the second review objective by contextualizing observations of mirroring (or lack of mirroring) so that the forces operating on multiple dimensions of interdependencies can be systematically compared and contrasted with one another.

2.2.3 Contexts of Mirroring

1. Mirroring mechanisms. An emerging consensus about mirroring is that there is no simple deterministic link between product and organization architectures (Cabigiosu *et al.*, 2013; Ernst, 2005; Fixson and Park, 2008; Hoetker, 2006; Sako, 2004). Multiple causal explanations have been invoked to explain mirroring. The evidence base provides an opportunity to systematically review the literature's current understanding of the mirroring mechanisms. Accordingly, three clusters of theoretical explanations for mirroring can be identified:

Coordination explanations. The first cluster explains mirroring in terms of the coordination or communication structures of organizations as a result of information processing or knowledge management. For instance, Conway (1968) states that organizations are constrained to produce products whose structures reflect the existing communication structures of the organizations (i.e., product mirrors organization). On the other hand, Sanchez and Mahoney (Sanchez and Mahoney, 1996) maintain that coordination tasks implicit in specific product designs largely determine the feasible organization designs for developing those products (i.e., organization mirrors product). In addition, Cataldo *et al.* (2006) demonstrate that achieving congruence between coordination requirements, which is determined by technical dependencies, and coordination activities, which determine organizational dependencies, reduces product development time (i.e., product and organization mirror each other). Therefore, whether it is a one-way constraint from either direction or a two-way constraint between product and organization, these researchers point out that the need to efficiently coordinate technical and organizational dependencies is an important mirroring mechanism.

It is worth noting that many studies that produce findings against the mirroring hypothesis (e.g., Brusoni and Prencipe, 2006; Brusoni, 2005; Cabigiosu *et al.*, 2013; Chuma, 2006; Ernst, 2005) also explain the contradictory findings through coordination explanations. These studies maintain that modularization requires more (not less) coordination if the underpinning technologies keep changing fast and unpredictably (Ernst, 2005) or if the inherent complexity of the product systems exceeds certain threshold (e.g., Cabigiosu *et al.*, 2013). Therefore, we still need to establish the boundary conditions on product modularity's role as a functional equivalent of overt managerial coordination.

Incentive explanations. The second cluster draws on a variety of theories from the organizational economics literature to explain mirroring. For instance, Cabigiosu and

Camuffo (2012) and Hoetker (2006) maintain that buyers of modular components are less likely to suffer from hold-up problems because the presence of other modular component suppliers helps curb opportunism, which in turn promotes the use of arm's length transaction. Similarly, Garud and Kumaraswamy (1995) submit that the presence of open standards, which indicates high inter-firm product modularity, reduces transaction costs and promotes market governance. These studies can be considered a part of the empirical transaction cost economics literature. Product modularity serves as an empirical apparatus to assess the risk of opportunism. Accordingly, theoretical extensions to transaction cost economics can point to unexplored contingencies on mirroring at the interfirm level for future research. For example, governance inseparability (Argyres and Liebeskind, 1999), the condition in which a firm's past governance choices constrain the range and types of future governance choices, implies that that there will be variance in observed governance mechanisms among firms for a given transaction. Kang, Mahoney, and Tan's (2009) real-option extension suggests that positive inter-project spillovers can explain deviations from standard transaction cost predictions. Thus, some observations of "broken mirrors" at the interfirm level might be explained by dynamic adjustment costs or real-option values.

In addition, Susarla, Barua, and Whinston (2010) draw upon agency theory and contend that modularity improves the verifiability of inter-firm tasks and therefore increases the desirability of high-powered incentives in such transactions. Similarly, Baiman, Fischer, and Rajan (2001) suggest that modular product architecture improves a firm's ability to trace the responsibility for its product's failures to the component suppliers. Moreover, Novak and Eppinger (2001) and Henkel and Baldwin (2011) argue that modularity helps better define property rights so firms can capture the benefits of their investment in specialized components. In sum, the second cluster explains mirroring as the result of the various ways in which product modularity improves incentive alignment between transacting partners. Future research can extend theses incentive explanations by connecting with the sociology literature on cooperation, given the increasing interests in the implications of mirroring in the context of open source software development (e.g., MacCormack, Baldwin, and Rusnak, 2012).

Competitive explanations. The third cluster explains mirroring as the results of various competitive dynamics of technological change. For instance, Langlois and Robertson (1992) contend that modular systems enable firms to take advantage of external economies, which provide the competitive advantage that explains the increasing prevalence of modular industry structure. Sturgeon (2002) similarly maintains that the key advantage of modular production networks is the build-up of external economies of scale. These accounts suggest that product modularity provides important cost advantage. On the other hand, Fixson and Park (2008) observe that new and superior integral product architectures can provide performance advantage that pushes the entire industry to become more integrated. To reconcile these opposing forces, Christensen, Verlinden, and Westerman (2002) maintain that whether performance or cost advantage would dominate depends on whether the customers are under-served or over-served by the functionality or performance available from products in the markets. In summary, change in product architecture can result in change in the performance and cost profiles of the product, which has competitive implications that alter the locus of competition and hence the structure of an industry.

Among the 36 studies[18] that examine mirroring at the within-firm level *only,* all of them rely on coordination explanations to explain mirroring. 34 studies appeal to coordination explanations exclusively, with the remaining 2 studies also discussing incentive mechanisms, but only tangentially. It is thus safe to say that the current literature considers coordination the predominant mirroring mechanism at the within-firm level.

Among the 39 studies that examine mirroring at the inter-firm level, 36 make reference to coordination as a mirroring mechanism. Coordination thus remains an important explanation for mirroring at the inter-firm level. However, unlike the within-firm studies, only 15 studies rely on coordination explanations as the exclusive mirroring mechanism. 23 studies rely on incentive alignment as the primary explanation for mirroring.

Among the 32 industry-level studies, 27 make reference to coordination as an explanatory mechanism (8 of them exclusively), 18 make reference to incentive alignment as an explanatory mechanism (no exclusive use of incentive alignment as explanation), and 5 make reference to various competitive dynamics as the explanatory mechanism (no exclusive use of competitive dynamics as explanation). Coordination remains an important mirroring mechanism at the industry level, though the type of explanations provided is observably more diverse.

[18] As reported earlier, 43 studies examine within-firm mirroring; however, only 36 of them examine within-firm level mirroring ***exclusively.*** I remove multi-level studies for this count in order to see if incentive explanations are also used at within-firm level.

Therefore, the evidence base suggests that although coordination explains the majority of mirroring, at higher levels, mirroring between product and organization results from the interaction of a variety of forces. This finding is consistent with Chesbrough's (2003) assertion that modular product design is a necessary but not sufficient condition for market modularity[19]. Therefore, modularity actually conveys different meanings or conditions at different levels. Failure to make these distinctions contributes to the inconsistent empirical findings in the current literature. Consequently, the mirroring hypothesis begins to fall apart when conceptualized as a simple deterministic relationship between concrete structural features of product and organization, rather than an outcome of dynamic interactions among multiple dimensions of interdependencies.

2. Industry contexts. A revisionist critique of the literature on the mirroring hypothesis is that it has the tendency to generalize empirical observations that are context-specific (Ernst, 2005). In particular, the computer industry has become the paradigmatic example of the mirroring hypothesis (Baldwin and Clark, 2000; Langlois, 1992; Langlois and Robertson, 1992). This industry emphasis has to some extent biased the rest of the research (Campagnolo and Camuffo, 2010), which therefore calls into question the generalizability of the proposed relationship beyond the computer industry.

---

[19] Specifically, Chesbrough (2003) draws on the institutional conditions that market needs to be able to function and posits that four criteria must be met for internal and market modularity to converge: (1) diffusion of architectural knowledge, (2) shared concepts and codes to specify component attributes, (3) existence of tools and equipment to verify component attributes, and (4) existence of capable supplier base. These criteria indicate that many institutional arrangements need to be in place before an internal modular design can become widely available commodity components.

The extent to which market or industry specific factors affect the mirroring relationship thus remains an important unanswered question (Cabigiosu *et al.*, 2013).

Categorizing findings into specific industry contexts, this review finds significant differences in the support for the mirroring hypothesis across industries. Specifically, among 48 empirical studies[20] with clearly defined industry contexts, 23 studies examine mirroring in the computer industry. 19 of the 23 studies find direct, indirect, or contingent support for the mirroring hypothesis; the remaining 4 studies produce contradictory evidence (17.39%). Among the 25 studies that examine industries other than the computer industry, 12 studies find direct, indirect, or contingent support for the mirroring hypothesis; 13 studies produce contradictory findings (52.00%). Clearly, the extant body of empirical evidence lends significantly more support to the mirroring hypothesis in the computer industry. Furthermore, breaking down the counts into industry subcategories reveals even sharper contrast. Among 16 studies that examine mirroring in software development, 15 studies find support; and only 1 study produces contradictory finding (6.25%). On the other end of the extreme, among 12 studies that examine mirroring in the automotive industry, 5 studies produce supportive conclusions, whereas 7 studies produce contradictory findings (58.33%). These numbers suggest that industry specific factors might play a significant role in the mirroring mechanisms.

To uncover industry specific factors that impinge on the underpinning mirroring processes, the studies that produce contradictory evidence are content analyzed to extract the rationales given to support the findings. The extracted rationales are then compared with the mainstream narratives of mirroring, as well as observations of mirroring in the

[20] This count only includes empirical studies with clearly defined industry contexts, which therefore reports different numbers from those reported in section 2.2.1.

software industry. This comparative analysis reveals the following critical distinctions across industries.

Distinct approaches to mitigate incidental interactions. As Ulrich (1994) and Sosa *et al.* (2004) point out, incidental interactions among modular components are inevitable and could remain unnoticed for much of the product development. These unknown interactions could result in costly "glitches," which create reciprocal task dependencies that require organizational integration to resolve (Hoopes and Postrel, 1999). Therefore, as discussed earlier, incomplete knowledge of component interactions (i.e., increased likelihood of incidental interactions) reduces mirroring between product and organization.

Many studies that provide evidence against the mirroring hypothesis report significant presence of incidental interactions. For instance, Staudenmayer *et al.* (2005) report that incidental interactions continually emerge despite efforts to limit them. Ro, Liker, and Fixson's (2007, 2008) case studies of the U.S. auto industry reveal that product modularity does not eliminate reciprocal task dependencies among firms. Similarly, Zirpoli and Camuffo (2009) show that intense information sharing between firms is a non-substitutable means to achieve coordination. Altogether, these studies suggest that, in settings similar to the auto industry, modular product architectures do not necessarily lead to the organizational benefits promised by the mainstream modularity literature.

Some researchers suggest that the inherent complexity of automobiles is the reason why product modularity is not an effective functional equivalent of overt managerial coordination (Cabigiosu *et al.*, 2013; Ro *et al.*, 2007, 2008). These researchers maintain that the complexity of automobiles entails high likelihood of incidental interactions. However, this line of reasoning fails to explain why the inherent complexity of software systems does not seem to generate the same level of coordination difficulty. Modern software systems can grow to millions of lines of code and require thousands of software developers. It is therefore hard to make a convincing argument that automobile development is necessarily more complex than software development.

Other researchers suggest that there might be industry specific factors that restrict the degree to which product architectures can be modularized (Zirpoli and Becker, 2011). In fact, the product architecture of automobile has been characterized as persistently integral (MacDuffie, 2008). Comparatively, computer and other electronics seem particularly suitable to modularization (Baldwin and Clark, 2000; Sturgeon, 2002; Zirpoli and Becker, 2011). However, the industry specific factors have yet to be explicitly identified.

Comparing between studies in the software and automotive industries reveals a critical distinction that might explain industry specificities in modularization. Importantly, I maintain that *software development relies heavily on a fundamentally different approach to mitigate unknown component interactions*. Instead of uncovering incidental interactions and managing them with mutual design adjustments, which is the typical approach in automotive or other non-computer industries, software development relies heavily on “thick module insulation” that guarantees any potential unwanted component interactions are effectively severed, even with no prior knowledge of their potential occurrence. With unknown interactions preempted, software development organizations could remain loosely coupled despite incomplete architectural knowledge.

Specifically, modern computer programming languages feature built-in syntactic supports[21] for information hiding (Parnas, 1972). That is, attempts to access or manipulate the internal design of other software components are deemed syntactically incorrect and would not be allowed. Therefore, software component interfaces are not just coordination devices that merely express how components should interact; software interfaces are in fact also *abstraction barriers* (Abelson and Sussman, 1996) that *enforce* access restriction to component internals. If the same approach were to be applied to automotive design, every component would be wrapped with thick thermal insulation and motion damper to prevent unknown interactions across components. Clearly, this approach would not be feasible in automotive design.

Importantly, using insulation to preempt interactions incurs performance or cost penalty. In the software engineering literature, this trade-off is known as *abstraction penalty* (Müller, 2000; Veldhuizen, 2004), which states that while high-level programming features make complex programming tasks simpler, they also produce less efficient software. Schilling's general theory depicts this trade-off in terms of *synergistic specificity,* which is "the degree to which a system achieves greater functionality by its

[21] Two well-known examples of these syntactic supports for information hiding are scoping and encapsulation (Friedman, Wand, and Haynes, 2001; Pierce, 2002; Scott, 2000). Scoping refers to the rules concerning the visibility of entities in a program, thereby hiding information from program segments that are outside of the scope. Encapsulation is an object-oriented programming technique that restricts access to data and functions bundled inside an object. Violations of scoping or encapsulation rules trigger "compile-time errors," which prevent the software systems from being built. In other words, these syntactic supports provide strong enforcement of modular design principles.

components being specific to one another" (2000: 316). Therefore, this analysis suggests that software development has a tendency to trade synergistic specificity for development simplicity in order to modularize development organization. In fact, much of the advancement in software engineering can be characterized as trading more and more computing resources for development simplicity so that developers can design increasingly complex software systems.

Furthermore, there is reason to believe that this approach to mitigate unknown interactions is unique to the computer industry. Specifically, semiconductor technology has achieved million-fold cost reductions and performance improvements since its inception, an observation known as the Moore's Law (Moore, 1965). The uniqueness of this performance trajectory can best be seen by projecting the growth pattern onto other technologies. Moore himself has jokingly mused:

> [I]f similar progress were made in transportation technologies such as air travel, a modern-day commercial aircraft would cost $500, circle the earth in 20 minutes, and use only five gallons of fuel. The catch is that it might be only the size of a shoe box. (as cited in Schaller, 1997: 58)

This unique exponential performance trajectory is therefore an important industry specific factor that enables the distinct modularization process in the computer industry. With the million-fold cost reductions and performance improvements in the underpinning semiconductor technology, computer engineers afford to indiscriminately rely on thick module insulation to preempt unknown component interactions. Engineers in other industries are much more constrained by performance and cost limitations of the underpinning technologies. Therefore, product development in the computer industry could proceed while remaining relatively ignorant of potential incidental component interactions vis-à-vis product development in other industries. Mirroring between product and organization is thus particularly strong in this industry. In summary, I suggest

> *Finding 4: The ways in which incidental component interactions are mitigated strongly influence the extent of mirroring between product and organization. Specifically, mitigating incidental interactions preemptively by insulating components increases the extent of mirroring, but reduces synergistic specificity. Mitigating incidental interactions by collective learning and mutual design adjustments reduces the extent of mirroring, but improves synergistic specificity.*

Thus, product and organization are more likely to mirror each other when there is room to trade performance and cost for development simplicity. A corollary thus follows

> *Corollary: The extent of mirroring between product and organization is inversely related to the extent to which the underpinning technologies are constrained by performance and cost limitations.*

Separation of design and manufacturing. This comparative analysis reveals another distinction across industries. Particularly, many studies outside of the software industry report that modularity initiatives are driven by objectives other than simplifying product development. For instance, Ro *et al.* (2007) report that manufacturing cost reduction is the primary driver for product modularity in the U.S. auto industry. Kotabe *et al.* (2007) report that modularity in the Brazilian auto industry is driven by the desires to seek lower labor cost, avoid strong labor unions, or reduce operational risks, among others. In addition, Cabigiosu and Camuffo (2012) observe strategic heterogeneity among air conditioning manufacturers pursuing product modularity, which can be associated with either more or less inter-firm organizational modularity depending on specific strategic intents. These findings are consistent with the life-cycle perspective

(Campagnolo and Camuffo, 2010; Fixson, 2003; Gershenson *et al.*, 1999), which maintains that the meanings and implications of modularity are relative to specific objectives throughout the development life-cycle. Together, these studies suggest that product modularity outside of the software industry is not always intended to bring about organizational modularity and therefore is not always associated with it.

In comparison, studies in the software industry more consistently report that product modularity is primarily intended to simplify product development. I maintain the reason for this relative consistency in modularization objective is that *software development entails no separation of design and manufacturing in its life-cycle.* Unlike manufacturing of physical products, software systems are information-based and do not require physical materials and construction. Therefore, software firms do not need to manage material coordinating tasks, such as logistics of raw materials and manufacturing, which also influence organizational design. Unlike product development in other industries, complexity of software development arises primarily out of interdependencies in the design dimension. Consequently, software products as a design artifact embodies a larger subset of the relevant interdependencies that impinges on organizational design as compared with other types of products[22]. Mirroring is therefore stronger between software products and their development organizations.

Since the notion of mirroring between product and organization can trace its root to the software engineering literature (Conway, 1968), the idiosyncrasies of software

[22] This claim that software products embody more knowledge is evidenced by the fact that access to source code makes reproducing a functioning software system relatively trivial; but even with the blueprint to a complex car engine, most people would still be unable to reproduce a functioning engine.

might have biased subsequent research in other industries. Specifically, supporters of modularity seem to have implicitly accepted the idea that product modularity allows firms to easily decouple *both* design and manufacturing (Cabigiosu *et al.*, 2013). This review suggests that the effects of product modularity are in fact more nuanced and specific to the intended objectives. Modularity in product design and modularity in manufacturing process are not connected through a simple, deterministic link, even though they are certainly interrelated[23]. Importantly, only a subset of the relevant organizational knowledge that enables a product design is embodied in and hence observable from the product artifact. Software product happens to be the exceptional case that has much of the design knowledge explicitly encoded in the artifact. Consequently, the structure of the artifact closely resembles the knowledge and hence task dependencies of the development organization. The observation of strong mirroring in the software industry is therefore more an exception rather than the rule. Accordingly, I submit

> *Finding 5: The extent of mirroring between product and its development organization depends on the extent to which the product embodies the relevant interdependencies that impinge on the development tasks. The more the product artifact embodies these interdependencies, the stronger product and organization mirror each other.*

Summary. This review identifies three types of mirroring mechanisms: coordination, incentive, and competitive. Although coordination explains the majority of mirroring, at higher level, mirroring between product and organization involves more causal forces. This hierarchy of nested mechanisms suggests that the meanings of product modularity also have a nested structure. In terms of coordination, product modularity is

[23] Refer to the example of Henry Ford's assembly line manufacturing discussed earlier.

about the technical decomposition of a complex product system into loosely coupled components. In terms of incentive alignment, these loosely coupled components require additional institutional arrangements to function as fully modular components at the inter-firm level. Technical decomposition alone is therefore insufficient. Modular interface not only specifies how components should interact at the technical level, it also reflects how jointly created value is apportioned between the transacting firms. In terms of competitive explanation of mirroring, product modularity also denotes the market condition in which the locus of competition is well defined by the modular interface.

Comparative analysis across industries also reveals important insights. First, I submit that mirroring between product and organization depends not only on *how modular* the product is, but also on how the product is *modularized*. The trade-off between performance and cost versus organizational modularity sets an important boundary condition on product modularity's role as a functional equivalent of overt managerial coordination. Second, this comparative analysis also calls into question the tacit assumption that product necessarily embodies all the organizational knowledge that goes into the development process. To the extent that product as a design artifact is only a partial embodiment of the underpinning knowledge, mirroring between the artifact and the organization that holds the enabling knowledge would always be incomplete.

Tables and Figures

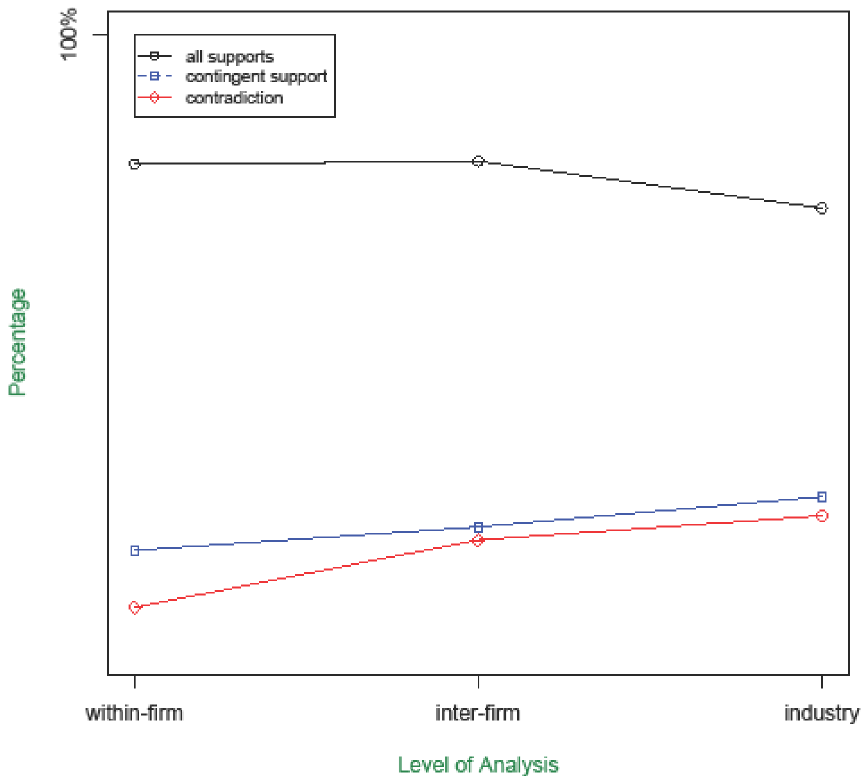
Administrative Science Quarterly	*Industrial and Corporate Change*
Academy of Management Review	*Industry and Innovation*
Academy of Management Journal	*Journal of Product Innovation and Management*
Organization Science	*IEEE Engineering Management Review*
Management Science	*IEEE Transactions on Engineering Management*
Organization Studies	*IEEE Software*
Journal of Management	*Research in Engineering Design*
Journal of Management Studies	*MIS Quarterly*
Strategic Management Journal	*Information Systems Research*
Strategic Entrepreneurship Journal	*Journal of Management Information Systems*
Strategic Organization	*Communications of the ACM*
Managerial and Decision Economics	*British Journal of Management*
Journal of International Business Studies	*International Journal of Management Reviews*
Journal of Business Venturing	*Harvard Business Review*
Journal of Business Research	*Sloan Management Review*
Research Policy	*California Management Review*

Table 2.1 The Journals Included

modular OR modularity OR "product architecture" OR "architectural innovation"
OR "mirroring hypothesis"
OR "Conway's Law"
OR "socio-technical congruence"
OR (product AND "organizational structure")
OR (product AND "division of labor")
OR (product AND "industry structure")
OR "technological change" OR "task partitioning"
OR "decomposability OR decomposition

Table 2.2. The Keyword Search String Used

Figure 2.1 Overview of Results



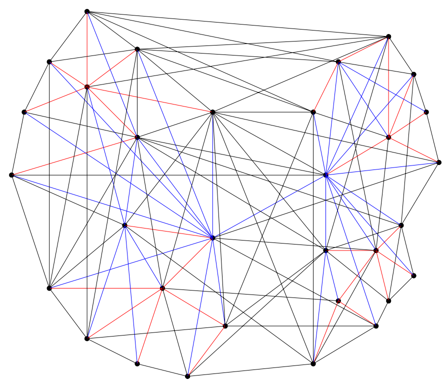


Figure 2.2 (a.)

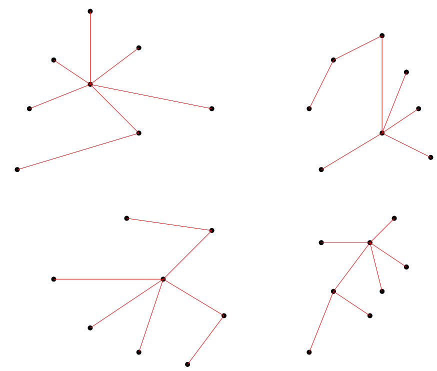


Figure 2.2 (b.)

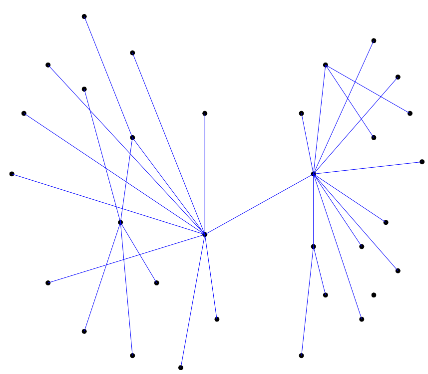


Figure 2.2 (c.)

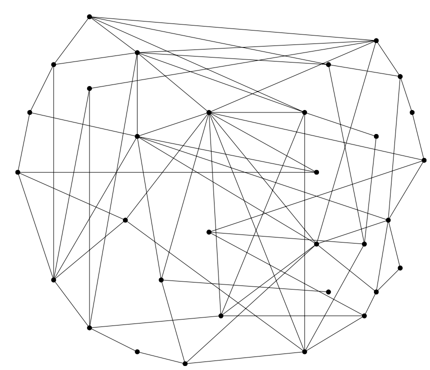


Figure 2.2 (d.)

In these figures, complex system is depicted by a network diagram. The nodes are elements in the system; and the lines connecting the nodes represent interdependencies. The colors of the lines represent specific dimensions. In figure (a.), all three dimensions are visible. It is hard to discern a clear pattern. In figure (b.), only the red dimension is made visible, and the picture clearly depicts four fully decomposed clusters. In figure (c.), only the blue dimension is made visible, and the picture shows a nearly decomposable system consisting of two loosely coupled clusters. In figure (d.), only the black dimension is made visible, and the picture shows a non-decomposable complex system. These diagrams illustrate that dimensionalizing a complex system can reveal in very different patterns of modularization.

CHAPTER 3

TOWARD A CONTINGENT MODEL OF MIRRORING – A KNOWLEDGE MANAGEMENT PERSPECTIVE

3.1 Introduction

Much of the contemporary management literature on modularity subscribes to the idea that increased product modularity is associated with advantageous increases in organizational modularity (e.g., Baldwin and Clark, 2000; Colfer and Baldwin, 2010; Jacobides, 2005; Langlois, 2002; MacCormack, Baldwin, and Rusnak, 2012; Sanchez and Mahoney, 1996; Sanchez, 1995, 1999; Sturgeon, 2002). Known as the *mirroring hypothesis* (Colfer and Baldwin, 2010; Colfer, 2007; Sanchez and Mahoney, 1996), this proposed relationship states that the structure of a product development organization and the architecture of the product being developed would come to "mirror" each other. However, as the review in the previous chapter shows, empirical research to date has not consistently found support for the hypothesis.

Moreover, extant theoretical discussions have not converged on a systematic explanation for the growing body of inconsistent empirical evidence.[24] An emerging view suggests that the proposed correspondence between product and organization actually reflects a more multiplex phenomenon than previously recognized (Cabigiosu and Camuffo, 2012; Hoetker, 2006) and that there is no simple deterministic link between

[24] Colfer and Baldwin (2010) report that 31% of the empirical studies in their review sample find no support for, or contradict, the mirroring hypothesis.

product and organization (Cabigiosu *et al.*, 2013; Ernst, 2005; Fixson and Park, 2008; Sako, 2004). Accordingly, leading scholars are calling for a more contingent model that goes beyond asking whether the mirroring hypothesis holds or not, but instead explains and predicts when the mirroring relationship will hold and when it will not (Cabigiosu and Camuffo, 2012; Colfer and Baldwin, 2010; Sanchez and Mahoney, 2013).

As a first step toward such a contingent model, this theory essay explores mirroring between product and organization at the *within-firm level*, which is the level of analysis in the original formulations of the mirroring hypothesis. Specifically, the proposed model is based on the premise that implicit assumptions underpinning the prediction of mirroring have been taken for granted and are in fact not always tenable. Therefore, a systematic examination of these implicit assumptions provides the basic structure for the theory development.

This essay begins with a brief overview of the extant theories of mirroring at the within-firm level and highlights some points of agreement and departure. The following section then identifies exogenous and endogenous factors that lower the likelihood that the mirroring hypothesis will hold and builds the logic for several propositions. This essay then concludes with a discussion on the implications of the proposed contingent framework.

3.2 Extant Theories of Mirroring

The idea that a relationship exists between product architecture and organization structure was developed more or less independently in different literatures. In software engineering, Conway (1968) observes that organizations are constrained to produce product designs that are copies of the communication structures of these organizations. Conway's (1968) reason for this constraint is that designers of different software components must communicate in order for these components to interface properly. This observation was later developed into the literature of socio-technical congruence, (Cataldo *et al.*, 2006, 2008), which posits that product development endeavors form a socio-technical system and that coordination requirements (as determined by product designs) and actual coordination activities (as embodied in organization structure) should be matched.

Similar ideas can be found in task dependency theory (Burns and Stalker, 1961; Chandler, 1962; Drazin and Van de Ven, 1985; Lawrence and Lorsch, 1967; Thompson, 1967; Woodward, 1965) and the information-processing view of organization (Brown and Eisenhardt, 1995; Burton and Obel, 2004; Galbraith, 1974; Tushman and Nadler, 1978) from the organizational theory literature, where it has long been recognized that organization designs should reflect the nature of the tasks to be performed (MacCormack *et al.*, 2012). To the extent that product architecture determines the nature of design tasks, the mirroring hypothesis can be understood as an application of this organizing principle to product design and development (Colfer and Baldwin, 2010).

In the contemporary management literature on modularity (e.g., Argyres and Bigelow, 2010; Cabigiosu and Camuffo, 2012; Hoetker, 2006), mirroring between product and organization has become a central topic. Researchers in this literature typically attribute the idea to the seminal works by Sanchez and Mahoney (1996) and

Baldwin and Clark (2000), which focus on how modularity as a design strategy facilitates loose coupling (Orton and Weick, 1990; Weick, 1976) in product and organization designs. Particularly, these scholars maintain that advanced architectural knowledge, which is the higher-order knowledge about how a product system's components interact and the system implications of these interactions, enables the standardization of *fully specified* component interfaces (Sanchez and Mahoney, 1996) or design rules (Baldwin and Clark, 2000). These fully specified component interfaces act as a functional equivalent of overt managerial coordination. Organizational modularity is therefore an option that product modularity enables. Since organizational modularity is generally considered desirable, subsequent research often assumes product modularity would lead to organizational modularity. Notably, the notion of mirroring in this literature has come to acquire distinct meanings that emphasize and advocate for the advantages of modularity.

Points of departure. While these theoretical accounts similarly predict structural correspondence between product architecture and organization structure, closer reading reveals significant differences among them. Specifically, extant theories appear to differ in the following ways:

1. Direction of causality. The mirroring hypothesis is often presented as a correlational statement, i.e., it predicts correspondence but does not impose direction of causality (Colfer and Baldwin, 2010; Sako, 2004). Correlational statement allows a diversity of causal specifications, which helps the literature remain inclusive to different theoretical narratives, e.g., effects flowing from organization to product (Conway, 1968; Henderson and Clark, 1990; Sanchez, Galvin, and Bach, 2013), from product to organization (Sanchez and Mahoney, 1996), or in both directions (Baldwin and Clark, 2000; Fixson and Park, 2008; Sako, 2004). Therefore, the predicted correspondence might be the combined effect of multiple mechanisms.

However, a correlational statement can mask causal details and create a false sense of determinism. For instance, subsequent research often cites Sanchez and Mahoney (1996) to put forth a simple correlational mirroring hypothesis; however, Sanchez and Mahoney's (1996) original statement is in fact more nuanced, suggesting that product designs *constrain*[25] *feasible choices* of organization designs. Specifically, Sanchez (1995) posits that integral product designs *require* integral organization designs, whereas modular product designs *enable* modular organization designs. Clearly, these carefully worded propositions leave room for contingent factors. Accordingly, Sako (2004) suggests product modularity gives greater scope for choice in organization design, but should not be understood as the sole determinant. The underpinning causal mechanisms are more complex than how mirroring is typically interpreted.

2. Need for communication. Another seemingly contradictory positions among the extant theories concern whether mirroring results from the organizational need to *facilitate* communication between designers of different components (e.g., Cataldo *et al.*, 2006, 2008; Conway, 1968; Henderson and Clark, 1990) or from the *elimination* of such need through product modularity (Baldwin and Clark, 2000; Sanchez and Mahoney, 2013). This apparent contradiction reflects two opposing perspectives on causal direction

[25] It is interesting that both Conway (1968) and Sanchez and Mahoney (1996), whose works appear to have been developed independently, use the word 'constrain' to depict the mirroring relationship. This choice of word suggests that these authors are conscious of other considerations that can impinge on product and organization designs.

(MacCormack *et al.*, 2012). One perspective underscores the need to match patterns of communication to the technical interdependencies that specific product designs entail, i.e., organizations *should* mirror products; the opposing perspective assumes that an organization's structure is fixed in the short-term and maintains that product designs are constrained by pre-existing communication channels, i.e., products *would* mirror organizations.

Put succinctly, both perspectives agree that communication needs and organization structures should be matched; but they disagree on how that match is best achieved. Empirical studies to date suggest that the direction of causality is idiosyncratic to a specific industry, or even a particular time period in an industry (Fixson and Park, 2008; Sako, 2004). Thus, an important unexplored research topic is to understand the contingent factors that influence the extent to which product modularity can reduce the need for communication vis-à-vis the difficulty of changing organization structure.

3. Role of modularity. Another apparent difference is the role of modularity in the theoretical narratives of mirroring. Even though modularity has long been considered an important software design principle, discussions about mirroring in the software engineering literature typically do not make reference to modularity. Instead, mirroring is more often stated as an isomorphism between technical and organization structures. Similarly, the organizational theory literature does not make reference to modularity to describe mirroring. In other words, mirroring is conceptualized in these literatures as structural similarity between product and organization, not just simply correlation between product and organizational modularity. Accordingly, Colfer and Baldwin (2010) specify an ideal test of the mirroring hypothesis in terms of the degree of structural similarity between product and organization.

Notably, the ideal test is much stronger than those specifications that conceptualize mirroring as correlation between product and organizational modularity. It can take into account the possibility that complex systems can be non-uniformly modularized[26]. For instance, if half of a complex system is highly modular but the other half remains highly integrated, a scalar modularity metric for the entire system would produce a medium value, which would not accurately characterize the true system architecture. In comparison, the ideal test does not depend on modularity measurements and is therefore less likely to produce false support for the mirroring hypothesis. Modularity as a design principle usefully promotes the benefits of systematic decomposition. However, as a description of system architecture, modularity strips away much of the structural details and should be best understood as a summary description. Formulating the mirroring hypothesis as correlation between two summary descriptions is therefore several steps removed from the actual mirroring process.

In sum, extant theories of mirroring are more diverse than commonly recognized in the contemporary literature. To make progress toward a more nuanced model of mirroring, taken for granted assumptions should be explicitly recognized to underscore important distinctions and reveal hidden contingencies.

Implicit assumptions. Much of the recent work in the modularity literature adopts Sanchez and Mahoney's (1996) emphasis on component interface standardization and

---

[26] However, the ideal test is also difficult to implement because of the need to construct matrix representations for both product and organization designs. This is why the ideal test is usually only feasible in the context of software development, where the software source code provides a convenient means to construct the needed matrix.

interprets it as synonymous with product modularity (e.g., Argyres and Bigelow, 2010; Jacobs *et al.*, 2011; Tiwana, 2008a, 2008b). As a result, interface standardization is often predicted to be associated with organizational modularity. However, a growing body of research reports that interface standardization does not always eliminate design interdependencies across components (Cabigiosu *et al.*, 2013; Chuma, 2006; Kotabe *et al.*, 2007; Ro *et al.*, 2008; Staudenmayer *et al.*, 2005; Takeishi, 2002; Tidd, 1995). For example, Staudenmayer and co-authors (2005) observe across multiple industries that interdependencies continually emerged despite having component interfaces that are defined ex-ante. These findings provide the substance to the debate over the empirical status of the mirroring hypothesis.

However, a closer reading of Sanchez and Mahoney (1996) reveals that their emphasis on interface standardization is predicated on the presence of *advanced architectural knowledge* about relevant components and their interactions. Similarly, Baldwin and Clark (2000) state that design rules become more complete as the properties of the system and the modules or the "innate interdependencies" become better understood. Subsequent research often overlooks this important premise or erroneously assumes the presence of standardized interface indicates the possession of advanced architectural knowledge. In addition, it is also implicitly assumed that architectural knowledge remains static as encoded in the standardized interface specification.

I submit that the central message of Sanchez and Mahoney (1996) is the strategic importance of advanced architectural knowledge. While they do emphasize how interface stability enables parallel component development and hence organizational modularity, their theoretical account does not presume the presence of sufficient architectural knowledge. In fact, by emphasizing the importance to acquire advanced architectural knowledge, they implicitly recognize its dynamic nature. Despite the presence of standardized interfaces, organizations can acquire new understanding of component interactions as the underlying technologies evolve. Organizations can even forget existing, structurally embedded architectural knowledge as Henderson and Clark (1990) show. Therefore, importantly, *modularity as observed in design artifacts is only a partial indicator of the level of architectural knowledge.*

Simply put, the extant literature on the mirroring hypothesis has come to implicitly adopt various assumptions of *completeness* – organizations adopting modular product development are assumed to possess architectural knowledge of all possible component interactions, such that the overall system architecture can be completely specified a priori. That is, these organizations need to be capable of specifying component interfaces that completely partition product systems into stand-alone modules that only interact as specified by the standardized interfaces. This architectural knowledge is also assumed to be completely embodied in the design artifacts, so that all technical interdependencies that would require organizational coordination can be observed directly from the artifacts. As I discuss in the following section, these assumptions are not always tenable and should be better understood as theoretical ideals. The extent to which these assumptions are borne out in practice places boundary conditions on the prediction of mirroring.

3.3 Contingent Model of Mirroring

Sosa, Eppinger, and Rowles (2004) maintain that although most of the architectural knowledge is explicit and known by development organizations, some architectural knowledge remains unspecified or unknown until detailed design. Following the same logic, I maintain that no product development organization is omniscient with regard to component interactions and their implications to system behaviors. Even with the best efforts at ex-ante standardization, some component interactions can only be identified during the design process itself, or even after the product system is put to use. In other words, I take the position that architectural knowledge is *inevitably incomplete*. Because architectural knowledge is always incomplete, it then follows that *ex-ante* standardization alone cannot fully eliminate the need for designers of different components to communicate. Importantly, the extent to which *ex-ante* standardization can reduce the need for communication depends on how much an organization knows about component interactions when the interface is being standardized. The more component interactions an organization is aware of, the more standardized interface can partition the product system into stand-alone modules and reduce the need for communication. For example, as automotive engineers understand more about how specific engine designs interact with the rest of the chassis to produce unwanted vibration, the way the engine is mounted can be standardized accordingly so that engine and chassis designers no longer have to communicate to reduce vibration. Without the knowledge about the interaction, engine and chassis designers still have to resort to joint problem solving to reduce vibration even if engine mounting is standardized according to other design considerations. Therefore, as a baseline, I propose that mirroring is contingent on the level of architectural knowledge.

Proposition 1: The extent of mirroring between product and organization increases with an increase in the level of architectural knowledge.

While some research studies take advantage of unique empirical settings to control for unobserved heterogeneity (e.g., Hoetker, 2006), which includes heterogeneity in architectural knowledge, most studies do not systematically assess this important contingent factor. This shortcoming is due in part to the current state of theory development that has not explicitly recognized this knowledge contingency, but is also due to the difficulty of measuring knowledge assets. Therefore, it might be useful to derive propositions with more observable constructs that correlate with an organization's possession of architectural knowledge.

Learning from experience. The literature on organizational learning considers learning from experience an important knowledge acquisition mechanism (Huber, 1991; Levitt and March, 1988). In the case of architectural knowledge, experience plays a particularly significant role in learning because component interactions that detrimentally impact system behavior are often unanticipated. These incidental interactions (Ulrich, 1994) only manifest themselves when components are physically put together during testing or final integration. Direct experience with a product architecture is therefore an irreplaceable element in architectural knowledge development. As a product development organization accumulates more experience with a particular product architecture it would encounter and become aware of more incidental component interactions, which enable the organization to achieve a higher level of modularization.

Furthermore, the likelihood that knowledge about incidental interaction becomes organizationally embedded as depicted by Henderson and Clark (1990) increases with the frequency of its occurrence and the severity of the problem it causes, both of which require time and experience to become clear. Organizational inertia (Hannan and Freeman, 1977, 1984) places further delay in the structural embodiment of the newly learned architectural knowledge. Mirroring between product and organization therefore takes time to realize, as organizations acquire experience with the product architecture and restructure themselves accordingly. Thus, I posit that mirroring becomes stronger as an organization accumulates more experience with a product architecture.

> *Proposition 2: The extent of mirroring between product and organization increases with an increase in the organization's cumulative experience with the product architecture.*

Vicarious learning. In addition to direct experience, organizations can also learn vicariously through other organizations' experience (Czepiel, 1975; Huber, 1991). The emergence of an industry dominant design (Abernathy and Utterback, 1978; Utterback and Abernathy, 1975) provides an opportunity for this form of learning. Dominant design embodies technological features that have won the allegiance of the marketplace and become a de facto standard (Suarez, 2004). In particular, dominant design encapsulates knowledge of component interactions and how to configure product architecture to better serve target user needs. It also signals that the industry has accumulated sufficient architectural knowledge for the dominant design to be taken for granted.

While adopting a dominant design does not imply the adopting organization necessarily possesses all of the underpinning architectural knowledge, it does indicate leading organizations in the industry have developed architectural knowledge that is being vicariously learned. That is, organizations with incomplete knowledge about component interactions can still manage to attain a modular partitioning of their product systems by imitating those organizations with more complete architectural knowledge. Mirroring is therefore also possible if the requisite architectural knowledge can be attained externally.

In contrast, organizations that choose to not adopt the industry's dominant design pass up the opportunity to benefit from industry knowledge and must rely on internal knowledge development. The ramification of this choice might be more significant than it first appears. Dominant design emerges as a synthesis of an industry's collective experience on the dominant architecture. It embodies the collective knowledge of a large number of organizations. Moreover, organizations forgoing industry dominant design are typically pursuing novel product architectures, of which relatively little is known. Thus, they are unlikely to achieve the same level of completeness in their architectural knowledge, as compared with those following the industry standard.

Furthermore, the adoption of a dominant design takes place as part of the broader institutional mimetic process as organizations model themselves on other organizations to reduce environmental uncertainty (DiMaggio and Powell, 1983). As a result of this isomorphic process, organization structures that mirror the dominant design would also tend to diffuse among those conforming to institutionalization, strengthening the mirroring relationship. Put differently, organizations that depend on vicarious learning to acquire architectural knowledge are also likely to imitate other organizational practices. I therefore suggest that mirroring is stronger if a product development organization adopts the dominant design of the industry.

Proposition 3: Mirroring between product and organization is greater when the organization adopts the industry dominant design, vis-à-vis those that choose not to adopt the industry dominant design.

Knowledge obsolescence. While organizations can benefit from industry knowledge, they are also confronted with challenges posed by industry dynamics. Innovation scholars have long recognized that some types of technological change can be competence-destroying and render extant knowledge assets obsolete (Abernathy and Clark, 1985; Tushman and Anderson, 1986). Architectural knowledge is therefore subject to the threat of obsolescence as technologies evolve. In particular, systemic or architectural innovations tend to render prior interface standards obsolete, hence reducing the completeness of an organization's architectural knowledge as old knowledge is made obsolete and new knowledge is yet to be fully internalized (Henderson and Clark, 1990; Zirpoli and Becker, 2011). In other words, architectural innovations "reset the clock" on an organization's experience with product architecture.

This threat of obsolescence is particularly pronounced in industries where competition occurs at the architectural level. Organizations in these industries compete by searching for novel product architectures that surpass their competitors along some performance dimensions. For instance, the semiconductor photolithographic alignment equipment industry in Henderson and Clark's (1990) study was shaken by four waves of architectural innovation between 1962 and 1986. During this period, the dominant architecture of the industry went through four disruptive transitions as new entrants successfully dethroned leading incumbents with novel product architectures. In contrast, product architecture has remained remarkably stable in the personal computer industry since its inception.

Therefore, in industries characterized by rapid architectural innovations, architectural knowledge is likely to be *persistently incomplete* as product development organizations engage in continuous explorations for novel architectures. In contrast, in industries with stable dominant architectures, architectural knowledge can be developed cumulatively toward increasing completeness. Moreover, as Henderson and Clark's (1990) study illustrates, architectural knowledge that has become structurally embedded significantly constrains an organization's ability to sense and respond to competitors' architectural innovations. Accordingly, strong mirroring can be a dangerous practice when architectural innovations are expected to happen frequently.

Organizations in industries with stable product architectures can accumulate more complete and nuanced knowledge about component interactions. They are also in a better position to take advantage of mirroring since the threat of knowledge embeddedness can be more easily mitigated when the underpinning technologies evolve in more predictable ways. Thus, I predict that mirroring would be stronger when the dominant product architecture remains stable.

> *Proposition 4: The extent of mirroring between product and organization increases with an increase in the stability of the dominant product architecture in the industry.*

In addition to architectural innovations, technological changes at the component level also contribute to the obsolescence of architectural knowledge. This connection has not received full attention because much of the management literature on modularity adopts Henderson and Clark's (1990) conception of modular innovation as innovation that changes the core component design concepts without changing the product architecture. Their conception has since been interpreted to imply a dichotomous conceptual separation of architectural and component innovations, which overlooks their interconnectedness.

In contrast, scholars from the systems integration literature (e.g., Brusoni *et al.*, 2001; Brusoni and Prencipe, 2001a, 2001b) subscribe to Rosenberg's (1976) idea that different modules in a multicomponent product system might change at different rates, thus creating a "technical imbalance" that focuses the innovative efforts of an organization. Particularly, this unevenness in technological progress across modules opens up opportunities for architectural innovations.

In fact, Henderson and Clark themselves have stated that "architectural innovation is often triggered by a change in a component" (1990: 12), and cite as example the jet aircraft industry, in which the introduction of jet engine technology provided a discontinuous performance improvement that changed the interactions between the engine and the rest of the plane in complex and subtle ways (Gardiner, 1986; Miller and Sawyers, 1968). Henderson and Clark (1990) suggest that failure of the incumbent firms to introduce new aircraft architecture accordingly was one factor that led to Boeing's rise to dominance. As their example illustrates, technological change at the component level can alter existing component interactions and generate new component interactions sufficiently significant to render extant architectural knowledge obsolete. In other words, component innovation can be the antecedent to architectural innovations.

One way to further clarify this connection is to revisit Henderson and Clark's (1990) seminal typology of technological change. Their typology classifies innovations along two dimensions: one captures whether an innovation reinforces or overturns the core component design concepts; the other captures whether an innovation maintains or changes the linkages between components. According to this classification, innovations can be incremental, modular, architectural, or radical (see Figure 3.1.a). What has become obscured in the subsequent literature is that in this typology, both incremental and modular innovations involve changes at the component level; and both architectural and radical innovations involve changes at the architectural level[27]. Moreover, the two dimensions are not meant to be symmetrical as they are often misinterpreted to be (see Figure 3.1.b). Whereas the horizontal dimension captures the extent of an innovation's impact on component, the vertical dimension captures whether or not an innovation impacts extant product architecture *at all.* This asymmetry is intended to highlight their core message – seemingly subtle reconfigurations of existing component technologies, i.e., architectural innovations, can pose significant challenge to incumbents.

My purpose here is to explore the connection between component and architectural innovations. Thus, I suggest that Henderson and Clark's (1990) incremental and modular innovations can be usefully relabeled "incremental component innovation" and "radical component innovation" to emphasize that they are both changes at the component level but differ in the extent of their impacts. This relabeling places component innovations along a continuum and allows us to assess how they trigger architectural innovation.

[27] Note that Henderson and Clark (1990) define radical innovations as ones that involve change at both the component and architectural levels.

By definition, component innovations do not impact existing product architecture. In theory, one can simply replace the old component with the innovation and the overall product system should continue to work. However, component innovation could interact with other components in ways that leave extant architecture usable but suboptimal. For example, a jet engine can be used, in theory, with an airframe designed to work with a propeller engine. However, the much greater output of the jet engine easily exceeds the specifications the propeller airframe was designed to handle. Early jet aircraft designers struggled with serious vibration and metal fatigue problems and only managed to overcome these problems with new airframe architectures and newly developed aluminum alloys (Bright, 1978). Otherwise, aircraft designers would have needed to throttle back jet engines considerably to be used with existing airframes. In other words, component innovations present product designers with the dilemma of choosing between disrupting system architecture and passing up the performance advantage the component innovations create.

Radical component innovations overturn the core design concepts of the affected components. They are therefore more likely to create greater technical imbalance among components than incremental component innovations, and render more extant architectural knowledge obsolete. Consequently, following radical component innovations, the extent of mirroring between product and organization would likely decrease as product development organizations seek more suitable architectures to take full advantage of the innovations. It is anticipated that the more radical the component innovation, the greater the reduction in mirroring.

Proposition 5: The extent of mirroring between product and organization is negatively correlated with the extent of component innovation, i.e., radical component innovations would trigger more reduction in mirroring than incremental component innovations.

In summary, I maintain that the implicit assumption of complete architectural knowledge is untenable in most scenarios and should be best understood as a theoretical ideal. Mirroring between product and organization is contingent on the organization's architectural knowledge about component interactions – organizations must be aware of component interactions before they can contain interactions through interface standardization. This knowledge prerequisite can be met either through internal knowledge development or through external knowledge access[28]. Mirroring is therefore stronger as a product development organization accumulates more experience with the product architecture or imitates the industry's dominant design. Moreover, architectural knowledge is subject to the threat of obsolescence. Architectural innovations can be planned in a top-down manner; they can also emerge from the bottom up as advancements in component technologies create opportunities to re-imagine better system architectures. Therefore, both architectural and component innovations render part of the extant architectural knowledge obsolete and thereby reduce the extent of mirroring.

3.4 Discussion

[28] My discussion here focuses on meeting the knowledge prerequisite that brings about mirroring. I do not intend to suggest that internal knowledge development and external knowledge access are perfect substitutes for architectural knowledge acquisition. The choice has obvious competitive implications.

Garud, Jain, and Tuertscher comment that the traditional discourse on design extols the virtues of completeness: "[c]ompleteness allows for the pre-specification of a problem, the identification of pre-existing alternatives and the choice of the most optimal solution" (2008: 351). This traditional approach to design follows the logic of constrained optimization (Sanchez and Mahoney, 1996) and pervades much of management thinking, education, and research (Romme, 2003). However, in environments characterized by changes, an emphasis on completeness leads to designs that foreclose future options (Barry and Rerup, 2006; Garud *et al.*, 2008).

This insight reveals a contradiction hidden in the extant modularity literature. On the one hand, modularity scholars attribute the organizational benefits of modularity to the information stability provided by the complete specification of component interface, i.e., completeness in architectural design creates the organizational benefits of modularity (e.g., Baldwin and Clark, 2000; Sanchez and Mahoney, 1996). On the other hand, these scholars also promote modularity as an option-creating design approach that helps meet the increasing demand for flexibility, variety, and extensibility (e.g., Baldwin and Clark, 1997, 2000; Langlois and Robertson, 1992), i.e., an important objective of modularity is the ability to adapt to future changes.

Importantly, these seemingly contradictory characterizations reflect an inherent tension in modularity as a design approach. Whereas design completeness at the architectural level is the enabling virtue, modular design at the component level is intended to be open-ended, though bounded by the standardized component interface. Therefore, the unspoken rationale of modular design is that component flexibility is attained through the acceptance of architectural rigidity as a trade-off. Consistent with

Farjoun's (2010) assertion that stability can enable change and adaptability, it is the commitment to a rigid architectural design, i.e., the standardized component interface, that enables the flexibility and autonomy of component design. Thus, modular design can be understood as an instance of the dual search for stability and change that pervades all forms of organizing (Weick, 1979: 136).

Recognizing this duality helps us gain a more nuanced understanding of modular design. While the extant literature emphasizes completeness in architectural design, when a modular system is considered across all levels (i.e., not just at the architectural level), modularity as a design approach actually harnesses what Garud *et al.* call "the generative forces of *incompleteness*" (2008: 356). This claim can be stated in terms that are more familiar to the extant literature: modularity entails the partition of design parameters into visible information about component interface and hidden information that only affects the functioning of the local modules (Baldwin and Clark, 2000); the hidden information is in effect an *incomplete specification of design parameters* from the perspective of the system as a whole, since component designers have the autonomy to change these parameters unilaterally without causing cascading changes throughout the system. Put simply, the essence of modular design is the creation of a hierarchy that paradoxically allows the co-existence of two opposing design logics. Modular design is intended to be simultaneously complete and incomplete,[29] so that it can be continually evolving in an organized fashion.

---

[29] This interpretation of modularity is consistent with Orton and Weick's (1990) dialectic interpretation of loose coupling as the juxtaposition of responsiveness and distinctiveness within the same system.

Accordingly, the contingent model of mirroring put forth above incorporates this nuanced understanding of modular design through the explicit recognition of incomplete architectural knowledge; namely, mirroring between product and organization is contingent on the level of architectural knowledge about component interactions. This knowledge management perspective recasts the current deterministic model of mirroring in terms of the inherent tension in modular design. The resulting contingent dynamics shed light on a fundamental trade-off product development organizations face when they encounter and learn new architectural knowledge about component interactions- Do they preserve the extant architecture and the established correspondence between product and organization so the benefits of interface stability can be sustained? Or do they disrupt extant architecture and the established correspondence between product and organization so that the full potential of the newly acquired architectural knowledge can be unleashed? As will be discussed in the next chapter, I submit that the target demand characteristics inform the organizations on how to make this trade-off, which in turns steers product and organization toward or away from stronger mirroring.

3.5 Conclusion

This essay contributes to the literature on the mirroring hypothesis by advancing a contingent model of mirroring that explicitly recognizes the important role of architectural knowledge. The proposed contingent model offers a way to bridge the gap between the mainstream narratives of mirroring and the emerging revisionist perspective. Specifically, the focus on how architectural knowledge is acquired and managed helps reconcile the mainstream literature with the systems integration literature, which maintains that no one-to-one mapping exists between product architecture and organizational architecture (e.g., Brusoni *et al.*, 2001; Brusoni and Prencipe, 2001a, 2001b). The proposed contingent model points to the changing level of architectural knowledge as the underpinning technologies evolve to help explain and predict when the mirroring relationship will hold and when it will not. Furthermore, the contingent role of demand characteristics also explains the contradictory findings against the mirroring hypothesis in the systems integration literature, since many of these studies take place in empirical contexts where system failures would result in catastrophic economic loss, e.g., aircraft engine (Brusoni *et al.*, 2001; Brusoni and Prencipe, 2001a), and chemical engineering industries (Brusoni and Prencipe, 2001a).

Sanchez and Mahoney (1996) famously suggest that although organizations ostensibly design products, it can also be argued that products design organizations. Instead, this essay takes the position that people with knowledge design both products and organizations. Modular design's duality of architectural rigidity and component flexibility requires thoughtful consideration to maintain. Mirroring between product and organization is therefore a transient state that reflects the combined effect of the underpinning dynamics as product designers and organization managers strive to maintain the organizational and product hierarchies that reconcile the opposing needs of stability and change. The proposed contingent model puts forth the first step toward revealing the underpinning mirroring dynamics and helps build a deeper understanding of how organizations manage the knowledge associated with the architecture of the products they design. Understanding how organizations manage this higher-order knowledge and how it impacts product and organization design offers important insights on how complex innovations are organized.

Tables and Figures

		Core Component Design Concepts	
		Reinforced	Overturned
Linkages between Components	Maintained	Incremental Innovation (Incremental Component Innovation)	Modular Innovation (Radical Component Innovation)
	Changed	Architectural Innovation	Radical Innovation

Figure 3.1 (a.)

Adapted from Henderson and Clark (1990)

Changes at Component Level

Changes at Architectural Level		Incremental	Radical
	Incremental	Incremental Innovation	Modular Innovation
	Radical	Architectural Innovation	Radical Innovation

Figure 3.1 (b.)

Common Misinterpretation of the Henderson and Clark's (1990) Typology of Innovation

CHAPTER 4

BREAKING MIRROR FOR THE CUSTOMERS – AN EMPIRICAL STUDY ON THE DEMAND-SIDE CONTINGENCIES OF THE MIRRORING HYPOTHESIS

4.1 Introduction

The mirroring hypothesis posits that a correspondence exists between product and organizational architectures. However, as the review in chapter 2 shows, extant theories of mirroring actually encompass a multitude of distinct causal mechanisms, suggesting that the predicted correspondence might be the combined effect of multiple interacting forces with potential contingencies. Accordingly, I develop in chapter 3 a contingent model of mirroring at the within-firm level by systematically unpacking the implicit but untenable assumption of completeness in architectural knowledge. The proposed model explicitly recognizes the contingent role of architectural knowledge and offers a way to reconcile the mainstream narratives of mirroring and the emerging revisionist perspective (e.g., Brusoni *et al.,* 2001; Brusoni and Prencipe, 2001a, 2001b) that challenges the mainstream predictions of mirroring. However, one limitation of the proposed contingent model, as well as the extant theories on mirroring, is that much of the discussion focuses only on supply-side issues, e.g., product design, firm's possession or access to architectural knowledge, and organizational design. Whether demand-side issues have an impact on mirroring remains unexplored.

In the broader strategic management literature, a growing number of scholars have been calling for more research attention on how demand heterogeneity contributes to firm heterogeneity (e.g., Adner, 2002, 2004; Adner and Levinthal, 2001; Adner and

Snow, 2010; Priem, 2007). In particular, Priem *et al.* (2012) maintain that strategy researchers have not given sufficient attention to demand-side issues. In practice, managers certainly need to take demand-side issues into consideration in order to formulate effective strategies for their organizations. Thus, we may be missing opportunities for new management knowledge that come from the demand-side.

This empirical study continues to explore contingencies on the mirroring relationship between product and organization. Following the call for more attention on demand-side issues, I develop a model of mirroring that is contingent on the demand characteristics of a firm's target customers. Specifically, I examine mirroring in the context of systems integration firms that have adopted industry standard modular architecture. Given the same product architecture and underpinning technologies, extant theories of mirroring would predict similar organizational architectures among these firms. Instead, I predict that the extent of mirroring would be reduced for firms targeting customers with high demands for *system performance*, as well as firms targeting customers who would suffer great economic loss in the event of system failures, i.e., customers with high demands for *system reliability*. I test these hypotheses with a distinctive empirical setting that allows variance in demand characteristics while still holding product architecture constant. Thus, the observed variance in organizational modularity can be attributed to the contingent role of demand characteristics. The empirical results lend support for the hypothesized demand-side contingencies, and also point to the need for more careful theoretical and empirical investigation to untangle the impacts of different demand characteristics on the mirroring relationship. The findings contribute to the growing empirical evidence of contingencies on mirroring (e.g., Cabigiosu and Camuffo, 2012; Cabigiosu *et al.,* 2013; Furlan *et al.,* 2014) and to the debate on the organizational implications of modularity.

This chapter proceeds as follows. The next section develops the theory on how demand characteristics impact mirroring between product and organization. I then discuss the empirical challenges to test this theory and how my research design overcomes these challenges. Next, I present my empirical method. After a discussion of the results, a final section concludes with implications and limitations.

4.2 Theory Development

The mainstream narratives of mirroring express a deterministic view linking the engineering structure of a product with the related organizational structure (Cabigiosu *et al.,* 2013). On the contrary, the revisionist literature (Brusoni *et al.,* 2001; Brusoni and Prencipe, 2001a; Cabigiosu *et al.,* 2013; Takeishi, 2001; Zirpoli and Camuffo, 2009) maintains that firms' knowledge boundaries extend beyond firms' task boundaries and hence there is no one-to-one mapping between product architecture and organizational architecture (Brusoni *et al.,* 2001). At the heart of this debate is whether firms need to know more than what they actually make in order to leverage external sources of innovation (Brusoni *et al.,* 2001; Takeishi, 2001) or they can rely on standardized component interface to achieve coordination without component-specific knowledge (Baldwin and Clark, 2000; Sanchez and Mahoney, 1996). Essentially, much of this debate is about the extent to which product modularity can reduce the amount of component-specific knowledge needed to perform systems integration.

I submit that the amount of component-specific knowledge needed to perform systems integration depends on the specific demand characteristics of the target customers. Therefore, different demand characteristics can be the important contingencies that help reconcile this long-standing debate. Specifically, I will show that target customers' demands for system performance and system reliability influence the amount of component-specific knowledge needed to perform systems integration. To substantiate this claim, I need to first make clear the distinction between system fine-tuning and interdependent component development[30].

System fine-tuning vs. component development. The review in chapter 2 identified an alternative interpretation of modularity that reveals an important but not well known distinction. Specifically, Fujimoto (2001) and Chuma (2006) contend that the defining characteristic of a non-modular product architecture is the need to perform post-assembly system fine-tuning because fine-tuning entails *mutual adjustments* of component configurations, which requires some component-specific knowledge. Accordingly, the defining characteristic of a modular product architecture is that the overall system performance can be automatically guaranteed if the specified performance of the constituent components is assured ex ante (Chuma, 2006). In other words, modular components are supposed to just "plug and play." This interpretation is a significant departure from mainstream interpretations of product modularity. It explicitly assumes the perspective of systems integrators and conceptualizes product modularity as the extent to which a component can be treated as a "black box" in the context of systems

[30] In Sanchez and Mahoney's (1996) terms, interdependent component development includes sequential product development and overlapping problem solving, both of which require heavy information flows between component development processes.

integration - the more modular a component, the less component-specific knowledge is required of the systems integrator to use such a component.

The distinction between system fine-tuning and interdependent component development is subtle but critical to my purpose here. Even though they are both instances of mutual adjustments across components (i.e., instances of reciprocal interdependencies across components), system fine-tuning does not change the designs of the components; more importantly, system fine-tuning does not change the standardized component interface, thus preserving the loosely coupled coordination among component developers. Fine-tuning only changes the *configurations* of the components within the range of component variation permitted by the standardized interface. For example, fine-tuning a computer system to serve a busy website might involve mutual adjustments of the networking module of the operating system and the database system that stores the content of the website; however, fine-tuning does not involve a redesign of either the network module or the database system. In other words, fine-tuning occurs during systems integration, at which point component development has already concluded. Therefore, system fine-tuning is an instance of *ex post* reciprocal interdependencies across components, whereas interdependent component development (i.e., non-modular component development) is an instance of *ex ante* reciprocal interdependencies.

Importantly, whereas component interface standardization reduces reciprocal interdependencies during component development, it does not necessarily reduce the need for post-assembly system fine-tuning. In fact, interface standards often parameterize aspects of component design in order to leave room for post-assembly fine-tuning. This practice allows systems design flexibility and component reusability toward future use cases that had not been conceived when component interface was standardized.

It follows from this distinction that systems integrators can fine-tune their product systems to have different qualities, even if they use the same components and implement identical interface standard. It also follows that contrary to Sanchez and Mahoney's (1996) claim, ex ante standardization of component interface does not always completely decouple component knowledge and architectural knowledge, if post-assembly system fine-tuning, and hence some component knowledge, is still required to ensure the integrated systems perform as intended.

Moreover, making clear this distinction helps us reach the counter-intuitive conclusion that even for the same product system, product modularity as experienced by component developers (i.e., interdependencies across components during component development) and product modularity as experienced by systems integrators (i.e., interdependencies across components during systems integration) need not be the same. This distinction thus provides a way to reconcile the mainstream narratives of mirroring and the revisionist challenge: advocates of modularity are right in that product modularity enables loosely-coupled coordination *among component developers* due to the embedded coordination of the interface standard (Sanchez and Mahoney, 1996), i.e., product modularity reduces *ex ante* reciprocal interdependencies; revisionists are also right in that product modularity does not always reduce reciprocal interdependencies *during systems integration*, i.e., product modularity does not always reduce *ex post* reciprocal interdependencies, which is consistent with their observations that integrators that have outsourced component production or even component design often still possess deep component-specific knowledge (Brusoni *et al.,* 2001; Takeishi, 2001). Thus, using standard modular components to build integrated systems is not always a simple matter of "plug and play." The contingent factor that influences the amount of component-specific knowledge needed during systems integration is therefore the focus of my following discussion.

Demand-side contingencies. In order to promote adoption, industry standard components are usually designed with the typical customer requirements in mind. For those systems integrators targeting customers with high performance demands, extant literature suggests that there are two ways to meet the stricter requirements.

First, they can identify the components that are critical to overall product performance and select the best performing components from the market. That is, systems integrators can take advantage of the increased rate of innovation enabled by the widespread adoption of modular designs (Baldwin and Clark, 1997). In this case, overall product performance is improved through modular innovations in performance-critical components. Much of the history of the personal computer industry can be characterized by this type of performance improvement. The second approach embraces the power of integrality (Fixson and Park, 2008) or synergistic specificity (Schilling, 2000), which achieves greater system performance by adopting a more integrated architecture. For example, Fixson and Park's (2008) historical case study on the bicycle drivetrain industry shows how a firm turned its formerly competitive industry into a near-monopoly through *decreasing* its product modularity. However, this approach requires a transition to a new product architecture, which is a fundamental change that breaks down established interface standards (Chesbrough, 2003). Overall product performance is improved through architectural innovation, often quite significantly; however, this approach requires heavy investment to establish new industry standards.

The earlier discussion on system fine-tuning points to a third approach that is commonly used but has not received much attention in the research literature. Namely, systems integrators can continue to use current interface standard, but improve system performance through system fine-tuning. This approach combines the advantages of the first two approaches. It does not break down established interface standards so systems integrators can continue to take advantage of the benefits of using established standards. At the same time, it allows integrators to achieve a tighter integration to improve system performance. Performance gains from fine-tuning (as well as modular innovations) eventually approach the limit of the current architecture (Ernst, 2005), at which point it becomes necessary to establish a new architecture in order to improve system performance further. However, for firms competing in the high performance market tier, being able to squeeze as much of the theoretical performance as possible out of the current interface standard through system fine-tuning provides an important competitive advantage.

Since system fine-tuning requires some component-specific knowledge to perform mutual adjustments of component configurations, systems integrators targeting customers with high performance demands need to retain more component-specific knowledge than predicted by the extant mirroring theories. Moreover, these systems integrators also need to track advancements in component technologies, which can create opportunities emerging from the bottom up to better fine-tune the overall system. In addition, these systems integrators also benefit from greater ability to recognize and react to architectural innovations, which means that they should avoid turning architectural knowledge inert by embedding it in organizational routines and information channels (Henderson and Clark, 1990). That is, systems integrators aspiring to compete in the high performance market tier should avoid strongly mirroring product architecture so they do not fall into the so-called "modularity trap" (Chesbrough and Kusunoki, 2001). This line of reasoning is also consistent with Prahalad and Hamel's (1990) warning against outsourcing core competencies and Cohen and Levinthal's (1990) advice to invest in absorptive capacity.

Taken together, whether systems integrators targeting high performance market tier choose to optimize performance within current interface standards through fine-tuning or disrupt current interface standards with architectural innovations, their organizational structure needs to remain relatively integrated to meet the additional knowledge requirements. Therefore, I propose:

Proposition 1: The extent of mirroring between product and organization is reduced for systems integration firms targeting customers with high performance demands.

In addition to performance, reliability is another demand characteristic that could impact the extent of mirroring. As I assert in chapter 3, architectural knowledge is inevitably incomplete, which results in incidental component interactions (Ulrich, 1994). Incidental interactions are those insidious interactions across components that remain unknown or unnoticed for much (or even all) of the product development process (Sosa *et al.,* 2004). Incidental component interactions often manifest themselves as system reliability issues. These "glitches" (Hoopes and Postrel, 1999) can range from minor annoyances (e.g., a personal computer crash) to catastrophic system failures (e.g., a plane crash). Even for the same product system, system failures can cause drastically different economic losses, depending on the specific use case. A computer system intended for personal gaming and a computer system intended to control radiation therapy equipment would have very different reliability requirements, even if these systems are built on identical component technologies and architecture.

Customers who suffer high economic losses in the event of system failures would pay a premium for products that are highly reliable. In order to cater to these customers, systems integrators need to organize themselves to discover and contain as many incidental component interactions as possible. Adhering to interface standards alone is insufficient to achieve a high level of reliability, because unlike functionality, system reliability is not easily decomposable into stand-alone components (Zirpoli and Becker, 2011). One reason for the incomplete decomposability is that multiple, interdependent functional dimensions (e.g., speed, noise, vibration, energy consumption, etc.) compose the overall system characteristics. More fundamentally, system reliability is not easily decomposable because it essentially is the outcome of a mitigation strategy against unforeseen circumstances. Thus, ex ante knowledge about component interactions (i.e., prior architectural knowledge) cannot fully substitute organizational integration to mitigate incidental interactions. Therefore, I propose:

> *Proposition 2: The extent of mirroring between product and organization is reduced for systems integration firms targeting customers with high reliability demands.*

4.3 Empirical Challenge and Solution

Understanding complex phenomena requires that we hold some units of observation constant. Hoetker (2006) comments that it has been difficult to empirically test the mirroring hypothesis because we rarely observe design processes that differ in their degree of product and organizational modularity, but not along other dimensions. To address this challenge, Hoetker (2006) used a unique empirical setting to control for confounding factors present in previous studies. Subsequent research have similarly used unique empirical settings to test the mirroring relationship (e.g., Argyres and Bigelow, 2010; Cabigiosu and Camuffo, 2012; Furlan *et al.,* 2014; MacCormack *et al.,* 2012).

The model of mirroring with demand-side contingencies presents an additional challenge. To empirically test demand contingencies, we not only need to control for confounding factors, but at the same time, allow variance in demand conditions. To address this challenge, I observe the organizational design choices of computer systems integration firms using the industry standard Intel x86 computer architecture, effectively holding product architecture constant in terms of software and hardware compatibility. Importantly, Intel x86 architecture is *not* synonymous with IBM PC compatible, since x86 computer architecture is also widely used in a large variety of computer systems beyond personal computing. Thus, this empirical context allows the needed variance in demand characteristics. In addition, the long-time market dominance of x86 architecture results in the proliferation of commercial off-the-shelf (COTS) components for all the components needed to build a functioning computer system. Even for more specialized use cases (e.g., avionics systems, defense systems, and telecommunication devices), systems integrators still have COTS components readily available from the market place[31].

Consequently, systems integrators adopting x86 architecture can easily mix and match modular components from a wide variety of readily available COTS components to build computer systems that serve different use cases. This combinative flexibility is one of the key benefits suggested by the proponents of modularity (Baldwin and Clark, 1997; Schilling, 2000). Furthermore, the proliferation of COTS components also means that systems integration firms do not need to possess component development capabilities to build functioning computer systems. In fact, building x86 compatible computer systems has become so accessible, even people without much technical knowledge and resources can manage to do so easily. The vibrant DIY PC building community is a testament to the widespread access to this standard architecture. This high degree of vertical specialization is consistent with the prediction of the mirroring hypothesis (Baldwin and Clark, 2000; Sanchez and Mahoney, 1996). Therefore, in this empirical context, decisions to not use readily available COTS components can be interpreted as a move away from spot markets towards integration, which are therefore cases of reduced mirroring, or deviation from the prediction of mirroring. Accordingly, two hypotheses can be derived in this empirical context to test the contingent model presented earlier:

> *Hypothesis 1: An increase in target customers' performance demand increases a systems integration firm's likelihood of deviating from using COTS components.*

> *Hypothesis 2: An increase in target customers' reliability demand increases a systems integration firm's likelihood of deviating from using COTS components.*

---

[31] This claim is verified by interviews with practitioners in the industrial computer manufacturing segment. Interviewees reported that almost all components they chose to develop internally have COTS counterparts available. These practitioners also provided catalogs for specialized COTS components.

4.4 Data and Method

To test these hypotheses, I carried out a cross-sectional quantitative study of computer systems integrators' decisions to deviate from using readily available COTS components, which indicate reduced mirroring. Interviews at three computer systems integrators[32] in the Silicon Valley supplemented the quantitative study.

Sample. The proliferation of x86 computer architecture into a large variety of industries presents a challenge for data collection. No single directory lists all systems integration firms adopting the x86 standard because these firms operate in different industries. To construct the sample of qualified systems integrators, I identified the SIC codes for 7 example firms that use x86 standard to implement computer systems in a variety of industries (personal computer, high performance engineering workstation, server computer, industrial computer, defense system, telecommunication device, and security device). With the 5 SIC codes identified for the 7 example firms, I used Hoover's Industry Directory to identify 14,214 firms in the 6 largest U.S. high-tech clusters according to reports[33] from the Milken Institute and Brookings Institute. These 6

[32] The three systems integrators interviewed were selected from different industry segments. One systems integrator produces industrial computer systems for a variety of specialized use cases in industrial or otherwise harsh environments. The second systems integrator produces Linux-based computer server and workstation for high performance computing. The third systems integrator specializes in servers and desktop computers for business applications. All of these firms produce systems fully compatible with the Intel x86 standard.

[33] The two reports referenced are *North America's High-Tech Economy: The Geography of Knowledge-Based Industries* by the Milken Institute and *High Tech Specialization: A Comparison of High Technology Centers* by the Center on Urban and Metropolitan Policy at the Brookings Institute.

clusters account for 16.4% of North American employment and 25.4% of North American wages in high-tech manufacturing and services industries[34]. I then screened these firms to identify systems integrators that meet the following conditions:

(1.) The company builds fully integrated computer systems; firms that only build partially assembled systems (known as "barebone" systems) that require further integration were excluded. This condition ensures that the included firm is directly responsible to the customers for the overall system performance and reliability;

(2.) The computer systems produced are fully compatible with Intel x86 architecture. This condition ensures that firms in the sample do have the choice between readily available x86-compatible COTS components vs. internal development or other sources of custom-design components.

This screening process identified the sample of 177 strategic business units (out of 173 firms) that sell fully integrated computer systems based on Intel x86 compatible architecture.

Dependent variable. A computer system can be conceptualized as a three-layer stack. At the bottom is the hardware layer, which consists of various semiconductor chips integrated on a printed circuit board called the "motherboard." At the middle is the system software layer, which includes the operating system (e.g., Microsoft Windows, Linux) and various hardware component controlling programs called "drivers." Hardware component firms develop these drivers in accordance with predefined interface standards so that their components can be compatible with the rest of the computer system. At the top is the application software layer, which includes packaged software programs (e.g.,

[34] According to the report by the Milken Institute.

Microsoft Office, Internet browsers) that interact directly with the users. Packaged software programs usually require an operating system to function. They are developed in accordance with the operating system's application programming interface (API), which ensures compatibility with the computer system.

Since x86 architecture is highly modular and standardized, and COTS components are readily available for all kinds of use cases, systems integrators in theory do not need to possess the capabilities to develop or modify any component across these three layers. Compatible hardware and software components from the spot markets are all supposed to "plug and play." Thus, an observation of internal development or modification activities in any of the three layers by a systems integrator indicates a deviation from perfect mirroring.

Along with two industry experts, I collected data for this variable by reviewing the company's product catalogs. Since systems integrators have an incentive to advertise their differentiating capabilities, it was easy to observe instances of deviation. We contacted those companies that did not provide sufficient information in their product catalogs to determine the value of this variable. The indicator variable DEVIATION is set to 1 if a systems integrator is observed to engage in component development or modification activities in any of the three layers. For example, if a systems integrator develops its own driver program for a hardware component instead of using the generic driver program provided by the component vendor, the indicator variable DEVIATION is set to 1; or if a systems integrator works with packaged software vendor to optimize or certify otherwise compatible packaged software, the indicator variable DEVIATION is also set to 1. Since the value of this variable is based on objective observation, the three coders achieved high agreement in the initial coding (agreement for 163 of the 177 strategic business units, or 92.09% agreement). We resolved the cases of disagreement after discussion.

Independent variables. To measure the performance and reliability demands of the focal firm's target customers, the two industry experts and I rated the company's product catalogs, websites, or any other available marketing materials we could obtain. We developed and pretested the initial coding procedures with 20 firms excluded from the final sample due to their adoption of non-x86 computer architectures, but otherwise compete in similar market segments as the included firms. However, it was determined that the two constructs (i.e., target customers' performance and reliability demand) were initially operationalized in a manner that lacks distinctiveness. Coders often confused the two constructs. For example, in many cases if a computer system performs too poorly, the resulting low performance could result in severe economic loss to the customer in a manner similar to system crashes, i.e., insufficient performance can result in the same devastating economic loss as total loss of performance in demanding use cases. In these use cases, the coders tended to code it as *both* high performance demand and high reliability demand.

In order to improve distinctiveness, the two constructs were subsequently recoded with new operationalizations. In particular, reliability demand was operationalized *strictly* in terms of *unexpected failure to meet design specifications.* Insufficient computing power, so long as it is not a result of unexpected failure to meet design specifications, is therefore made conceptually distinct from insufficient reliability. The new operationalizations were tested with another 20 firms not included in the final sample. The coders discussed discrepancies in the coding and refined the coding protocol accordingly.

To assess intercoder reliability with 3 coders and interval scale, the appropriate measure is Krippendorff's alpha (Krippendorff, 2004). For target customers' performance demand, the Krippendorff's alpha among the 3 coders is 0.763; for target customers' reliability demand, the Krippendorff's alpha among the three coders is 0.842. These reliability measures are above the common threshold of 0.7 in content analysis research (Krippendorff, 2004). The independent variable PERFORMANCE is set to the average value of the three expert codings. Similarly, the independent variable RELIABILITY is set to the average of the three expert codings.

Control variables. While the objective of this study is to assess the influence of demand characteristics on a firm's decision to forgo using COTS components, other factors may influence this decision as well. For instance, production volume can influence the likelihood of deviating from using COTS components due to its impact on the overall cost structure. There are fixed costs associated with product development. Firms that choose to forgo using readily available COTS components have to bear these fixed costs. Therefore, there needs to be sufficient volume for internal component development to be an economically viable option. In addition, firms with high production volume can potentially achieve significant savings if the components are custom designed to reduce the cost of production by eliminating some unwanted features from industry standard components. At large volume, even a slight decrease in component cost can easily outweigh the fixed costs of custom design and even result in additional profits. Thus, production volume of the integrated systems is included as a control variable to ensure that the observed relationships between the dependent variable and the theoretical variables are not influenced by it.

Exact production volume is difficult to measure consistently across all the systems integration firms in the sample. Market research firms like International Data Corporation and Gartner provide unit sales estimates for the large PC vendors; however, for systems integrators in specialized categories, such information is difficult to obtain. Therefore, I used number of employees found in Hoover's Industry Directory as a proxy to a firm's production volume. Since the distribution for number of employees in my sample is highly skewed, I used natural log of the number of employees as the measure of production volume for this research.

In addition, if a systems integration firm also has business selling internally developed components (not just retailing components) to other systems integrators, this firm is likely to prefer the components developed internally over other COTS components regardless of demand characteristics. Participation in the component business also indicates the firm possesses component development capability independent of their systems integration business. Since capabilities influence firms' vertical boundary choices (Leiblein and Miller, 2003), participation in the component business should be controlled as well.

Moreover, there are reasons to believe that firms selling integrated computer systems to military and other government agencies might organize their component development differently due to the certification requirements in accordance with relevant military standards and specifications. I therefore also controlled for firms that have obtained certifications for military standards.

Table 4.1 lists all the variables and their operationalization. Correlations and descriptive statistics for all variables included in the models are presented in Table 4.2.

Model specification. I employed a binary choice logit model to assess the relationship between a set of covariates and whether or not a systems integration firm deviates from using COTS components. Specifically, the binary choice model assumes a firm's decision to deviate is determined by an unobservable, latent variable explained by several regressors. The observation of a firm's deviation decision is therefore assumed to indicate whether the value of the latent variable exceeds a threshold value. This model specification produces the following multivariate statistical model:

$$\text{DEVIATION} = \beta_0 + \beta_{1\text{-}3}\,\text{Controls} + \beta_4\,\text{PERFORMANCE} + \beta_5\,\text{RELIABILITY} + \varepsilon$$

4.5 Results

Table 4.3 summarizes the coefficient estimates and goodness-of-fit measures for the 4 logit models used to test the hypotheses. These models estimate the effects of the covariates on the probability that a systems integration firm will deviate from using readily available COTS components. Since this industry is well known for its high level of product modularity and a large variety of COTS components are available for all the needed system components, deviation from using COTS components can be interpreted as deviation from perfect mirroring between product and organization. Thus, a positive coefficient indicates the variable is positively related to the probability of deviation from perfect mirroring.

Model 1 is the baseline model with only the control variables included. Only PRODUCTION VOLUME is statistically significant. The coefficient estimate is positive as expected. Model 2 and Model 3 introduce the independent variables PERFORMANCE and RELIABILITY respectively. The coefficients are all highly significant and are all positive as predicted. Model 4 introduces all independent and control variables. The coefficients for all the independent variables are statistically significant and are all positive as predicted as well. The results from these models provide strong support for the two hypotheses.

To determine the net effects of PERFORMANCE and RELIABILITY on the likelihood of deviation from perfect mirroring, I took the coefficients obtained from Model 4 and plotted the predicted probabilities of deviation against the two independent variables, with all other variables evaluated at their mean values. Figure 4.1 indicates that both PERFORMANCE and RELIABILITY have a positive impact on a systems integration firm's probability to deviate from perfect mirroring. Moreover, RELIABILITY appears to have a stronger marginal effect on probability to deviate for most of the range in the dataset.

Robustness checks. To assess the robustness of the results, I ran additional models with interactions between variables. Because customers with high performance or high reliability demands are underserved by the mainstream market, these customers would be willing to pay a price premium to obtain the high performance or high reliability systems they need (Christensen *et al.,* 2002). This price premium reduces the production volume needed to justify the fixed costs associated with internal component development. In other words, the positive effects of customer's performance and reliability demands on the likelihood of deviation should be larger for firms with higher production volume.

According to Hoetker (2007), the marginal effect of an interaction between two variables in a logit model is not simply the coefficient for their interaction. Due to the nonlinear nature of logit models, the magnitude and even the sign of the marginal effect can differ across observations (Huang and Shields, 2000). Thus, interpretation of interactions is more complicated in logit models. To make it easier to assess interactions, I transformed the continuous variable PRODUCTION VOLUME into a categorical variable by median splitting into low and high categories. Model 5 replaces the continuous variable PRODUCTION VOLUME in Model 4 with the categorical VOLUME DUMMY variable. Results from Model 5 are consistent with results from Model 4, both in terms of coefficient estimates and goodness-of-fit measures, suggesting that the categorical variable can be an acceptable substitute for the continuous variable. This model specification produces the following multivariate statistical model:

DEVIATION =
$\beta_0 + \beta_{1\text{-}2}$ Controls + β_3 VOLUME DUMMY +
β_4 PERFORMANCE + β_5 RELIABILITY +
β_6 VOLUME_DUMMY × PERFORMANCE +
β_7 VOLUME_DUMMY × RELIABILITY + ε

Table 4.4 summarizes the coefficient estimates and goodness-of-fit measures for the 3 logit models used to assess interaction between production volume and the two independent variables. Model 6 includes only the independent variable PERFORMANCE and the interaction term. Consistent with Model 4, coefficient for PERFORMANCE is statistically significant and positive. However the interaction term is not statistically significant in this model. Model 7 includes only the independent variable PERFORMANCE and the interaction term. Consistent with Model 4, coefficient for RELIABILITY is statistically significant and positive. However the interaction term is also not statistically significant in this model. Model 8 is the full model with both independent variables and interaction terms included. Coefficients for RELIABILITY and the interaction term between RELIABILITY and VOLUME DUMMY are statistically significant. Coefficients for PERFORMANCE and the interaction term between PERFORMANCE and VOLUME DUMMY are not statistically significant.

Interpretation is further complicated by the fact that the significance of the interaction effect in logit models cannot be determined by just the significance of the interaction coefficient (Hoetker, 2007). To help interpret the results obtained, I followed Hoetker's (2007) recommendation to produce graphical presentation in order to provide the most complete understanding of the interaction's effect. In addition, I also followed Zelner's (2009) recommended simulation-based approach, as implemented in STATA's marginsplot command, to produce the 95% confidence interval in Figure 4.2(a.) and 4.2(b.) to help interpret the interaction.

Even though the coefficient for the interaction term between RELIABILITY and VOLUME DUMMY is statistically significant, Figure 4.2(b.) shows the 95% confidence intervals are clearly separated only between the RELIABILITY = 5.5 and RELIABILITY

= 8.5. This pattern indicates that statistically, the interaction effect is significant only in a specific range. These results suggest only reliability demand reduces the production volume needed to justify the fixed costs associated with internal component development, and only over a specific range.

In addition, I ran additional models to assess the interaction between the independent variables with the control variable COMPONENT BUSINESS. Participation in the component business indicates possession of component development capability, which might interact with the two independent variables in their impact on probability to deviate from perfect mirroring. Firms participating in the component business should be more likely to deviate from using COTS components from the market given the same level of performance and reliability demand. This model specification produces the following multivariate statistical model:

$$\begin{aligned} &\text{DEVIATION} = \\ &\beta_0 + \beta_{1\text{-}3}\,\text{Controls} + \beta_4\,\text{PERFORMANCE} + \beta_5\,\text{RELIABILITY} + \\ &\beta_6\,\text{COMPONENT BUSINESS} \times \text{PERFORMANCE} + \\ &\beta_7\,\text{COMPONENT BUSINESS} \times \text{RELIABILITY} + \varepsilon \end{aligned}$$

Table 4.5 summarizes the coefficient estimates and goodness-of-fit measures for the 3 logit models used to assess interaction between participation in component business and the two independent variables. Similar to the interaction with production volume, the results only indicate interaction between RELIABILITY and COMPONENT BUSINESS. However, contrary to expectation, the sign of the interaction terms between RELIABILITY and COMPONENT BUSINESS is consistently negative.

In summary, the empirical results strongly support the two hypotheses. Increase in production volume increases the likelihood of using custom designed components instead of readily available COTS components. Increase in target customers' performance and reliability demand reduces the extent of mirroring for systems integration firms. As expected, target customers' reliability demand interact with production volume, while target customers' performance demand does not appear to interact with production volume.

4.6 Discussion

Within the mainstream modularity literature, product modularity is said to be associated with loosely coupled organizations that use market-based coordination mechanisms to coordinate their product development activities (Baldwin and Clark, 2000; Sanchez and Mahoney, 1996). This study explains why firms adopting standardized modular product architecture sometimes deviate from this prediction. I proposed and found strong empirical support that systems integration firms would refrain from perfect mirroring if they target customers with high performance or high reliability demands, because these firms are more reliant on system fine-tuning to achieve the desired product qualities. The computer industry has become the paradigmatic example of the mirroring hypothesis (e.g., Baldwin and Clark, 2000; Langlois, 1992; Langlois and Robertson, 1992). Finding clear evidence of demand contingencies in this paradigmatic context provides strong support for the contingent nature of the mirroring hypothesis.

Furthermore, the empirical results also indicate that target customers' reliability demand has a greater and more consistent impact on the mirroring relationship than target customers' performance demand. Thus, different demand-side factors impact the mirroring relationship in different ways, suggesting the need for more careful theoretical and empirical investigation to untangle the different mechanisms. Theoretically, the extant literature has suggested two alternative ways of improving product system performance: 1.) modular innovation accessible through the component market; 2.) architectural innovation as the result of standard disruption. My discussion earlier suggests a third alternative: namely, through careful fine-tuning or “tweaking” the system. The equifinality in performance improvement mechanisms suggests that systems integrators would select the least costly approach. Empirically, knowledge about this industry provides the cost explanations to the observed differential impacts of performance and reliability demands. Specifically, the underpinning semiconductor technology has achieved a persistent doubling of performance approximately every two years, an observation known as the Moore’s Law (Moore, 1965). This trajectory translates to million-fold cost reductions and performance improvements in one of the performance critical components of a computer system. The unique exponential performance growth and cost reduction diminish the cost-effectiveness of system fine-tuning as a way to attain marginal performance gain, which helps explain the weaker impact of performance demand on mirroring in this context.

System reliability, on the other hand, does not automatically improve as the underpinning component technologies improve. In addition, system reliability is also relative to the unique use case the integrated product system is intended for. Thus, unlike performance, reliability is more specific to the particular target customers' needs, since each unique use case can potentially introduce product deployment conditions that had not been considered when the component standard was defined. Systems integrators pursuing high reliability thus have to resort to tighter organizational integration to discover and contain incidental component interactions that cause reliability issues.

The differential impacts of performance and reliability on mirroring also reveals that there are in fact two different kinds of system fine-tuning. Fine-tuning for better system performance is more often guided by existing knowledge of component interactions, i.e., fine-tuning for better performance is enabled by extant architectural knowledge. The improved system performance is the intended consequence of the fine-tuning efforts. On the other hand, fine-tuning for better reliability proceeds as an experiment to uncover unintended and therefore unknown component interactions, i.e., fine-tuning for better reliability is in essence an organized search effort for new architectural knowledge, which requires a more integrated organizational structure. Fine-tuning for reliability therefore has a greater impact on mirroring between product and organization.

4.7 Limitations

This study has several limitations. First, reliance on subjective expert coding can potentially introduce measurement issues for the key variables. Even though the coding procedure produced acceptable intercoder reliability, reliability does not guarantee construct validity. The dichotomous coding for the dependent variable also reduces the observed variation in the statistical analysis. Since deviation from perfect mirroring was observed across the three-layer stack (i.e., hardware, system software, and application software), multinomial logit model could have been employed to exploit the observed variation more. However, the limited sample size prevented such an approach.

Second, there are reasons to believe that the constructed sample does not cover all industries that adopt Intel x86 architecture. Anecdotal evidence indicates widespread adoption of this technology in the medical device and defense industries. However, these industries were not well represented in the sample, because firms in these industries are reluctant to disclose their product details due to security or liability concerns. Therefore, the constructed sample might be biased, although interviewees at the three systems integrators did provide similar accounts for their engagements in these industries.

Finally, there are alternative explanations besides system fine-tuning that cannot be fully ruled out due to the limitations of the empirical design. Specifically, differential capabilities in component technologies might better explain firms' vertical boundary choices (Leiblein and Miller, 2003). Even though I included the variable COMPONENT BUSINESS to control for this alternative explanation, the control variable is not statistically significant as expected, suggesting that systems integrators' participation in component business is perhaps not a god measurement for component capabilities in this empirical context. A related alternative explanation reinforces this concern. Perhaps a systems integrator forgoes using readily available COTS component because it possesses *unique*, superior component development capabilities. This firm might be able to extract monopoly rent if it chooses to always bundle the component with the rest of the system. In this case, lack of participation in component business is in fact the result of superior capabilities.

Interviews with industry practitioners suggest an additional alternative interpretations of my empirical results on reliability. Customers intending to deploy in mission-critical application sometimes demand component service and replacement availability far exceeding the typical time period provided by COTS component vendors. These customers are unwilling to take on the uncertainty of discontinued component service or replacement availability because once they certify the system for their mission-critical applications, they would prefer not to change any detail of their deployment. Thus, even without the need to fine-tune for better reliability, systems integrators might still internalize the component development tasks in order to satisfy the extended service and availability expectations. However, these customers also typically demand high reliability in their systems. My current empirical design is therefore unable to tease apart the two mechanisms.

4.8 Conclusion

This study contributes to the modularity literature in several ways. First, this study puts forth a contingent model and provides empirical evidence that help reconcile the mainstream narratives of mirroring and the emerging revisionist perspective that challenges the mainstream predictions. As discussed earlier, there is an inherent tension in modular design as systems integration firms try to improve overall product performance. These firms can rely on modular innovations in performance-critical components to deliver better overall performance, while preserving the benefits of having established interface standards. However, this approach places an upper limit on performance that is inherent in the current architecture. In addition, this approach also means the system integrators are dependent on external component suppliers to improve performance-critical components. Alternatively, these firms can choose to disrupt established interface standards with architectural innovations, which can potentially provide significantly better performance but at a much higher cost and risk of failure.

I suggest that there is a third commonly used approach of system fine-tuning, which combines the advantages of the first two approaches. The reliance on system fine-tuning as a mechanism to optimize performance within current product architecture provides the demand-side contingencies that reconcile the long-standing debate. Firms that rely more on system fine-tuning are expected to have reduced mirroring between product and organization.

Second, the discussion on system fine-tuning also leads to a counter-intuitive conclusion that challenges an implicit assumption of symmetry in the extant literature. Because the architectural knowledge on component interactions can never be complete, product modularity as experienced during system decomposition and product modularity as experienced during system integration are not perfectly symmetrical. There will always be some unforeseen integration issues, in spite of ex ante standardization of component interface (Cabigiosu *et al.*, 2013; Chuma, 2006; Kotabe *et al.*, 2007; Ro *et al.*, 2008; Sosa *et al.,* 2004; Staudenmayer *et al.*, 2005; Takeishi, 2002; Tidd, 1995).

This asymmetry not only provides a way to reconcile the debate on mirroring, it also points to the need to further refine our conceptualization of modularity. Along similar line of logic, Andriani and Carignani (2014) puts forth the concept of modular exaptation, which is the cooption of existing component technologies to serve system functions not originally designed for the components. In this case, system integration is likely to encounter much more incidental interactions not accounted for in the existing component interface. Fixson and Park (2008) suggest that the set of modular design operators put forth by Baldwin and Clark (2000) is incomplete. Future theory development on modular design should address this potential asymmetry of decomposition and integration.

Third, this study provides an empirical analysis on how demand heterogeneity contributes to firm heterogeneity. Consistent with Priem and coauthors' (2012) observation, researchers have not given sufficient attention to demand-side issues. This study is the first to my knowledge that looks into how target customers' demands impact the mirroring relationship between product and organization. Future research can further explore how different demand factors impact the mirroring relationships differently, shedding more light on how managers manage the interactions between the technologies under development and the organizations that develop these technologies.

Tables and Figures

Conceptual variable	Empirical data	Variable name
Control variable	Is this firm also selling system components at either hardware, system software, or application software level to other customers (0/1)	COMPONENT BUSINESS
Control variable	Is this firm's products certified for relevant United States Defense Standards and Specifications (0/1)	MILITARY
Control variable	Natural log of the number of employees in the firm (or in the specific strategic business unit for multi-divisional firms)	PRODUCTION VOLUME
Control variable	Categorical variable created by median splitting production volume (0=low; 1=high)	VOLUME_DUMMY
Independent variables		
Target customer's performance demand	Average of the three expert coding of the firm's target customer's system performance demand	PERFORMANCE
Target customer's reliability demand	Average of the three expert coding of the firm's target customer's system reliability demand	RELIABILITY
Dependent variable		
Deviation from perfect modularity	Does the firm deviate from using commercially available, "off-the-shelf" components at either hardware, system software, or application software level (0/1)	DEVIATION

Table 4.1 Conceptual variables and corresponding empirical data

	Mean	S.D.	Min.	Max.	1	2	3	4	5	6	7
1. DEVIATION	0.51	0.50	0	1	1.000						
2. COMPONENT BUSINESS	0.15	0.36	0	1	0.255	1.000					
3. MILITARY	0.06	0.24	0	1	0.157	-0.109	1.000				
4. PERFORMANCE	3.85	2.47	1	10	0.572	0.229	0.015	1.000			
5. RELIABILITY	4.58	2.91	1	10	0.712	0.151	0.327	0.553	1.000		
6. LOG(EMPLOYEES)	3.42	2.23	1.1	11.39	0.516	0.556	0.049	0.417	0.414	1.000	
7. VOLUME_DUMMY	0.51	0.50	0	1	0.57	0.255	0.157	0.488	0.553	0.715	1.000

Table 4.2 Descriptive statistics and correlations

	Model 1	Model 2	Model 3	Model 4
COMPONENT BUSINESS	0.506	0.682^{**}	0.223	0.530
	(1.20)	(2.82)	(0.18)	(0.39)
MILITARY	1.125	1.249^{*}	-0.976	-0.453
	(1.32)	(2.24)	(-1.07)	(-0.51)
PRODUCTION VOLUME	0.872^{***}	0.640^{***}	0.517^{***}	0.517^{***}
	(5.84)	(6.50)	(4.35)	(3.47)
PERFORMANCE		0.558^{**}		0.358^{**}
		(3.69)		(2.84)
RELIABILITY			0.732^{***}	0.639^{***}
			(11.50)	(6.50)
Constant	-2.710^{***}	-4.170^{***}	-4.889^{***}	-6.079^{***}
	(-10.26)	(-7.43)	(-6.04)	(-5.96)
Observations	177	177	177	177
Adjusted Count R^2	0.477	0.628	0.663	0.686
McFadden's R^2	0.279	0.415	0.529	0.568
Log-likelihood	-88.466	-71.745	-57.738	-53.008

t statistics in parentheses
* $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$

Table 4.3 Results of logistic regression analyses for deviation from perfect modularity

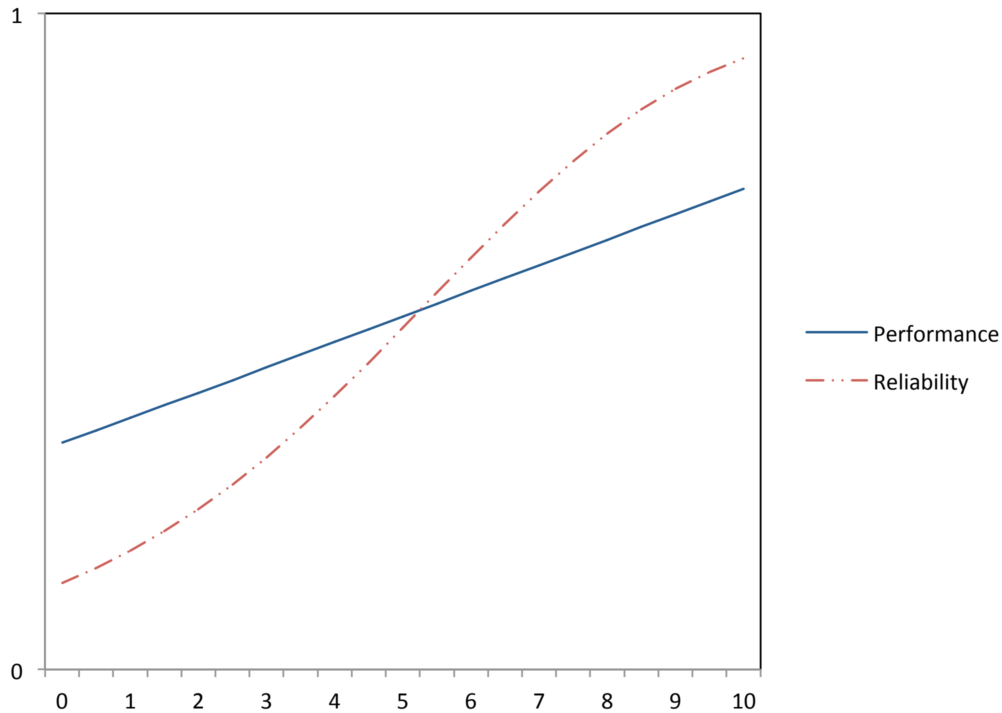


Figure 4.1 Predicted Probabilities

	Model 4	Model 5	Model 6	Model 7	Model 8
COMPONENT BUSINESS	0.530	1.873*	1.193***	1.628*	1.961*
	(0.39)	(2.17)	(3.31)	(2.55)	(2.42)
MILITARY	-0.453	-0.471	1.188*	-1.008	-0.469
	(-0.51)	(-0.73)	(2.01)	(-1.44)	(-0.65)
PRODUCTION VOLUME	0.517***				
	(3.47)				
VOLUME DUMMY		1.218***	3.074**	0.645	0.733
		(3.62)	(2.68)	(1.23)	(0.53)
PERFORMANCE	0.358**	0.320*	0.696**		0.476
	(2.84)	(2.45)	(2.78)		(1.58)
RELIABILITY	0.639***	0.619***		0.635***	0.439***
	(6.50)	(8.45)		(5.54)	(4.04)
VOLUME DUMMY * PERFORMANCE			-0.338		-0.240
			(-1.40)		(-0.70)
VOLUME DUMMY * RELIABILITY				0.171	0.317**
				(1.55)	(2.63)
	(-5.96)	(-7.70)	(-4.81)	(-8.31)	(-4.17)
Observations	177	177	177	177	177
Adjusted Count R^2	0.686	0.698	0.651	0.651	0.733
McFadden's R^2	0.568	0.544	0.410	0.516	0.554
Log-likelihood	-53.008	-55.865	-72.401	-59.342	-54.656

t statistics in parentheses

* $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$

Table 4.4 Results of logistic regression analyses for deviation from perfect modularity

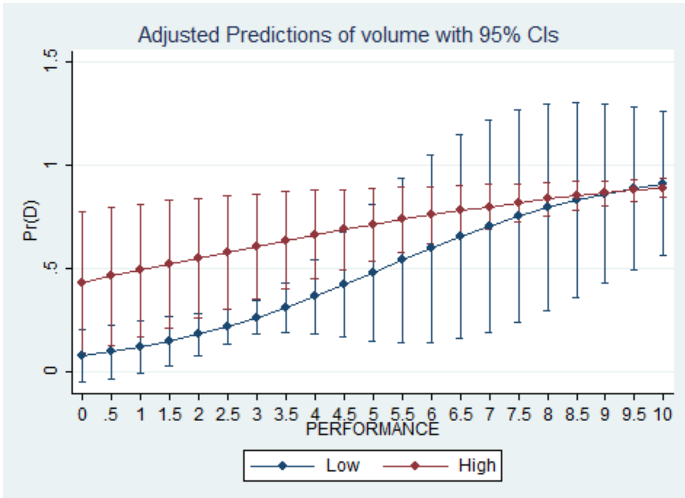


Figure 4.2 (a.)

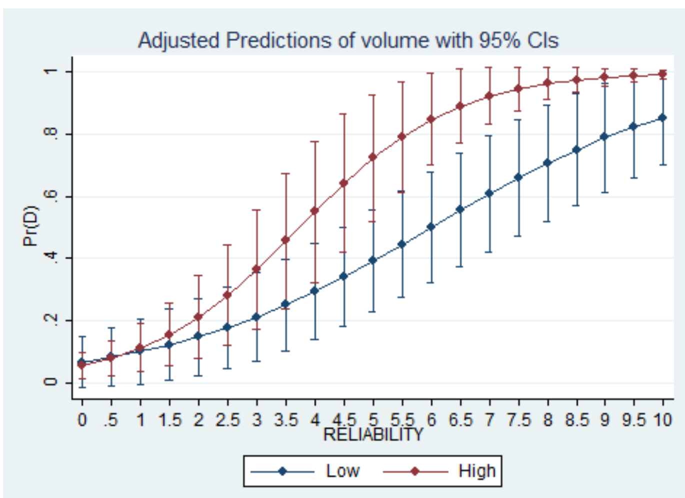


Figure 4.2 (b.)

	Model 9	Model 10	Model 11
COMPONENT BUSINESS	1.518	3.463^{**}	4.835^{*}
	(1.80)	(2.53)	(2.38)
MILITARY	1.252^{*}	-1.387	-0.827
	(2.29)	(-1.33)	(-0.88)
PRODUCTION VOLUME	0.638^{***}	0.479^{***}	0.446^{***}
	(6.45)	(6.08)	(4.77)
PERFORMANCE	0.578^{***}		0.434^{**}
	(4.11)		(3.06)
RELIABILITY		0.885^{***}	0.835^{***}
		(7.00)	(4.46)
COMPONENT BUSINESS * PERFORMANCE	-0.242		-0.170
	(-1.55)		(-0.93)
COMPONENT BUSINESS * RELIABILITY		-0.664^{**}	-0.726^{**}
		(-2.63)	(-2.68)
Constant	-4.237^{***}	-5.477^{***}	-7.203^{***}
	(-8.03)	(-5.66)	(-4.59)
Observations	177	177	177
Adjusted Count R^2	0.640	0.698	0.721
McFadden's R^2	0.416	0.553	0.598
Log-likelihood	-71.556	-54.864	-49.271

t statistics in parentheses
* $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$

Table 4.5 Results of logistic regression analyses for deviation from perfect modularity

REFERENCES

Abelson H, Sussman GJ. 1996. *Structure and Interpretation of Computer Programs - 2nd Edition*, second edition. The MIT Press: Cambridge, MA.

Abernathy WJ, Clark KB. 1985. Innovation: Mapping the winds of creative destruction. *Research Policy* **14**(1): 3–22.

Abernathy WJ, Utterback JM. 1978. Patterns of Industrial Innovation. *Technology Review* **80**(7): 40–47.

Adner R. 2002. When are technologies disruptive? A demand-based view of the emergence of competition. *Strategic Management Journal* **23**(8): 667–688.

Adner, R. 2004. A demand-based perspective on technology life cycles. In *Business Strategy over the Industry Lifecycle*, Advances in Strategic Management, Baum, Joel, McGahan A (eds). Emerald Group Publishing Limited, Vols. 1-0, 21: 25–43. Available at: http://www.emeraldinsight.com/doi/abs/10.1016/S0742-3322(04)21001-4.

Adner R, Levinthal D. 2001. Demand Heterogeneity and Technology Evolution: Implications for Product and Process Innovation. *Management Science* **47**(5): 611–628.

Adner R, Snow D. 2010. Old technology responses to new technology threats: demand heterogeneity and technology retreats. *Industrial and Corporate Change* : 1655–1675.

Alexander C. 1964. *Notes on the Synthesis of Form*. Harvard University Press: Cambridge, MA.

Andriani P, Carignani G. 2014. Modular exaptation: A missing link in the synthesis of artificial form. *Research Policy* **43**(9): 1608–1620.

Argyres N, Bigelow L. 2010. Innovation, Modularity, and Vertical Deintegration: Evidence from the Early U.S. Auto Industry. *Organization Science* **21**(4): 842–853.

Baiman S, Fischer PE, Rajan MV. 2001. Performance Measurement and Design in Supply Chains. *Management Science* **47**(1): 173–188.

Baldwin CY, Clark KB. 1997. Managing in an Age of Modularity. *Harvard Business Review* **75**(5): 84–93.

Baldwin CY, Clark KB. 2000. *Design Rules: The Power of Modularity*. MIT Press: Cambridge, MA.

Barry D, Rerup C. 2006. Going Mobile: Aesthetic Design Considerations from Calder and the Constructivists. *Organization Science* **17**(2): 262–276.

Bertalanffy von L. 1968. *General system theory: foundations, development, applications*. G. Braziller: New York.

Biggerstaff T, Richter C. 1987. Reusability Framework, Assessment, and Directions. *IEEE Software* **4**(2): 41 –49.

Bright CD. 1978. *The jet makers: the aerospace industry from 1945 to 1972*. Regents Press of Kansas: Lawrence, KS.

Brown SL, Eisenhardt KM. 1995. Product Development: Past Research, Present Findings, and Future Directions. *The Academy of Management Review* **20**(2): 343–378.

Brusoni S. 2005. The limits to specialization: Problem solving and coordination in 'Modular Networks'. *Organization Studies* **26**(12): 1885–1907.
Brusoni S, Prencipe A. 2001a. Unpacking the Black Box of Modularity: Technologies, Products and Organizations. *Industrial and Corporate Change* **10**(1): 179–205.
Brusoni S, Prencipe A. 2001b. Managing Knowledge in Loosely Coupled Networks: Exploring the Links between Product and Knowledge Dynamics. *Journal of Management Studies* **38**(7): 1019–1035.
Brusoni S, Prencipe A. 2006. Making Design Rules: A Multidomain Perspective. *Organization Science* **17**(2): 179–189.
Brusoni S, Prencipe A, Pavitt K. 2001. Knowledge Specialization, Organizational Coupling, and the Boundaries of the Firm: Why Do Firms Know More Than They Make? *Administrative Science Quarterly* **46**(4): 597–621.
Burns T, Stalker GM. 1961. *The management of innovation*. Tavistock Publications: London.
Burton RM, Obel B. 2004. *Strategic organizational diagnosis and design: the dynamics of fit*. Springer: New York.
Bush AA, Tiwana A, Rai A. 2010. Complementarities Between Product Design Modularity and IT Infrastructure Flexibility in IT-Enabled Supply Chains. *IEEE Transactions on Engineering Management* **57**(2): 240–254.
Cabigiosu A, Camuffo A. 2012. Beyond the 'Mirroring' Hypothesis: Product Modularity and Interorganizational Relations in the Air Conditioning Industry. *Organization Science* **23**(3): 686–703.
Cabigiosu A, Zirpoli F, Camuffo A. 2013. Modularity, interfaces definition and the integration of external sources of innovation in the automotive industry. *Research Policy* **42**(3): 662–675.
Callebaut WG. 2005. The Ubiquity of Modularity. In *Modularity: Understanding The Development And Evolution Of Natural Complex Systems*. MIT Press: Cambridge, MA: 3–28.
Campagnolo D, Camuffo A. 2010. The Concept of Modularity in Management Studies: A Literature Review. *International Journal of Management Reviews* **12**(3): 259–283.
Cataldo M, Herbsleb JD, Carley KM. 2008. Socio-technical congruence: a framework for assessing the impact of technical and work dependencies on software development productivity. In *Proceedings of the Second ACM-IEEE international symposium on Empirical software engineering and measurement*, ESEM '08. ACM: New York, NY, USA: 2–11. Available at: http://doi.acm.org/10.1145/1414004.1414008.
Cataldo M, Mockus A, Roberts JA, Herbsleb JD. 2009. Software Dependencies, Work Dependencies, and Their Impact on Failures. *IEEE Transactions on Software Engineering* **35**(6): 864 –878.
Cataldo M, Wagstrom PA, Herbsleb JD, Carley KM. 2006. Identification of coordination requirements: implications for the Design of collaboration and awareness tools. In *Proceedings of the 2006 20th anniversary conference on Computer supported cooperative work*, CSCW '06. ACM: New York, NY, USA: 353–362. Available at: http://doi.acm.org/10.1145/1180875.1180929.

Chandler AD. 1962. *Strategy and Structure: Chapters in the History of the Industrial Enterprise*. M.I.T. Press: Cambridge, MA.
Chesbrough H. 2003. Towards a Dynamics of Modularity A Cyclical Model of Technical Advance. In *The business of systems integration*, Prencipe A (ed). Oxford University Press: New York: 174–198.
Chesbrough HW, Kusunoki K. 2001. The Modularity Trap: Innovation, Technology Phase Shifts and the Resulting Limits of Virtual Organizations. In *Managing Industrial Knowledge: Creation, Transfer and Utilization*, Nonaka I, Teece DJ (eds). SAGE: Thousand Oaks, CA: 202–230.
Christensen CM, Verlinden M, Westerman G. 2002. Disruption, disintegration and the dissipation of differentiability. *Industrial and Corporate Change* **11**(5): 955 –993.
Chuma H. 2006. Increasing complexity and limits of organization in the microlithography industry: implications for science-based industries. *Research Policy* **35**(3): 394–411.
Clark KB. 1985. The interaction of design hierarchies and market concepts in technological evolution. *Research Policy* **14**(5): 235–251.
Cohen WM, Levinthal DA. 1990. Absorptive Capacity: A New Perspective on Learning and Innovation. *Administrative Science Quarterly* **35**(1): 128–152.
Colfer L. 2007, November 22. The Mirroring Hypothesis: Theory, Evidence and Exceptions. Available at: http://www.people.hbs.edu/cbaldwin/DR2/ColferMirroring.pdf [27 July 2012].
Colfer L, Baldwin CY. 2010, February 18. The Mirroring Hypothesis: Theory, Evidence and Exceptions. Available at: http://hbswk.hbs.edu/item/6361.html [27 July 2012].
Conway M. 1968. How Do Committees Invent? *Datamation* **14**(4): 28–31.
Cooper HM. 1998. *Synthesizing Research: A Guide for Literature Reviews*. SAGE: Thousand Oaks, CA.
Cross N. 2008. *Engineering Design Methods: Strategies for Product Design*. Wiley: New York.
Czepiel JA. 1975. Patterns of Interorganizational Communications and the Diffusion of a Major Technological Innovation in a Competitive Industrial Community. *Academy of Management Journal* **18**(1): 6–24.
Davies HT, Crombie IK. 1998. Getting to grips with systematic reviews and meta-analyses. *Hospital medicine (London, England: 1998)* **59**(12): 955–958.
Denyer D, Neely A. 2004. Introduction to special issue: Innovation and productivity performance in the UK. *International Journal of Management Reviews* **5**(3-4): 131–135.
DiMaggio PJ, Powell WW. 1983. The Iron Cage Revisited: Institutional Isomorphism and Collective Rationality in Organizational Fields. *American Sociological Review* **48**(2): 147–160.
Drazin R, Van de Ven AH. 1985. Alternative Forms of Fit in Contingency Theory. *Administrative Science Quarterly* **30**(4): 514–539.
Ernst D. 2005. Limits to Modularity: Reflections on Recent Developments in Chip Design. *Industry & Innovation* **12**(3): 303–335.
Evans D, Pearson A. 2001. Systematic reviews: gatekeepers of nursing knowledge. *Journal of Clinical Nursing* **10**(5): 593–599.

Farjoun M. 2010. Beyond Dualism: Stability and Change as a Duality. *Academy of Management Review* **35**(2): 202–225.
Fink A. 2010. *Conducting Research Literature Reviews: From the Internet to Paper*. SAGE: Thousand Oaks, CA.
Fixson SK. 2002. *Linking modularity and cost: A methodology to assess cost implications of product architecture differences to support product design*. Ph.D., Massachusetts Institute of Technology, United States -- Massachusetts. Available at: http://search.proquest.com/docview/305498247/13B4B12B7DC7A474EC5/2?accountid=14553.
Fixson SK. 2003. *The multiple faces of modularity - a literature analysis of a product concept for assembled hardware products.* Technical Report 03-05, Industrial and Operations Engineering, University of Michigan.
Fixson SK. 2007a. What Exactly is Product Modularity? The Answer Depends on Who You Ask. *SSRN eLibrary*. Available at: http://papers.ssrn.com/sol3/papers.cfm?abstract_id=952045.
Fixson SK. 2007b. Modularity and Commonality Research: Past Developments and Future Opportunities. *Concurrent Engineering* **15**(2): 85 –111.
Fixson SK, Park J-K. 2008. The power of integrality: Linkages between product architecture, innovation, and industry structure. *Research Policy* **37**(8): 1296–1316.
Friedman DP, Wand M, Haynes CT. 2001. *Essentials of Programming Languages*. MIT Press: Cambridge, MA.
Fujimoto T. 2001. Architecture no Sanygo-ron (Theory of industry in view of architecture]: Business Architecture). In *Business Architecture: Seihin Soshiki Process no Senyakuteki Sekkei (Business Architecture: Strategic Design of Products, Organizations, Processes).* Yuhikaku.
Furlan A, Cabigiosu A, Camuffo A. 2014. When the mirror gets misted up: Modularity and technological change. *Strategic Management Journal* **35**(6): 789–807.
Galbraith JR. 1974. Organization Design: An Information Processing View. *Interfaces* **4**(3): 28–36.
Galvin P, Morkel A. 2001. The effect of product modularity on industry structure: The case of the world bicycle industry. *Industry and Innovation* **8**(1): 31–47.
Gardiner JP. 1986. Design trajectories for airplanes and automobiles during the past fifty years. In *Design, Innovation, and Long Cycles in Economic Development*, Freeman C (ed). St. Martins's Press: London: 121–141.
Garud R, Jain S, Tuertscher P. 2008. Incomplete by Design and Designing for Incompleteness. *Organization Studies* **29**(3): 351–371.
Garud R, Kumaraswamy A. 1995. Technological and organizational designs for realizing economies of substitution. *Strategic Management Journal* **16**(S1): 93–109.
Gershenson JK, Prasad GJ, Allamneni S. 1999. Modular product design: A life-cycle view. *Journal of Integrated Design and Process Science* **3**(4): 13–26.
Gershenson JK, Prasad GJ, Zhang Y. 2003. Product modularity: definitions and benefits. *Journal of Engineering Design* **14**(3): 295.
Gershenson JK, Prasad GJ, Zhang Y. 2004. Product modularity: measures and design methods. *Journal of Engineering Design* **15**(1): 33–51.

Glass GV. 1976. Primary, Secondary, and Meta-Analysis of Research. *Educational Researcher* **5**(10): 3–8.
Glass GV, McGaw B, Smith ML. 1981. *Meta-analysis in social research*. Sage Publications: Thousand Oaks, CA.
Hannan MT, Freeman J. 1977. The Population Ecology of Organizations. *The American Journal of Sociology* **82**(5): 929–964.
Hannan MT, Freeman J. 1984. Structural Inertia and Organizational Change. *American Sociological Review* **49**(2): 149–164.
Hart C. 1998. *Doing a literature review: releasing the social science research imagination*. Sage Publications: Thousand Oaks, CA.
Henderson RM, Clark KB. 1990. Architectural Innovation: The Reconfiguration of Existing Product Technologies and the Failure of Established Firms. *Administrative Science Quarterly* **35**(1): 9–30.
Henkel J, Baldwin C. 2011, January 12. Modularity for Value Appropriation--How to Draw the Boundaries of Intellectual Property — HBS Working Knowledge. Available at: http://hbswk.hbs.edu/item/6585.html [5 January 2013].
Herbsleb JD, Grinter RE. 1999. Architectures, coordination, and distance: Conway's law and beyond. *IEEE Software* **16**(5): 63 –70.
Von Hippel E. 1990. Task partitioning: An innovation process variable. *Research Policy* **19**(5): 407–418.
Hirtz J, Stone RB, McAdams DA, Szykman S, Wood KL. 2002. A functional basis for engineering design: Reconciling and evolving previous efforts. *Research in Engineering Design* **13**(2): 65–82.
Hoetker G. 2006. Do modular products lead to modular organizations? *Strategic Management Journal* **27**(6): 501–518.
Hoetker G. 2007. The use of logit and probit models in strategic management research: Critical issues. *Strategic Management Journal* **28**(4): 331–343.
Hoetker G, Swaminathan A, Mitchell W. 2007. Modularity and the Impact of Buyer–Supplier Relationships on the Survival of Suppliers. *Management Science* **53**(2): 178–191.
Hoopes DG, Postrel S. 1999. Shared Knowledge, 'Glitches,' and Product Development Performance. *Strategic Management Journal* **20**(9): 837–865.
Huang C, Shields TG. 2000. Interpretation of Interaction Effects in Logit and Probit Analyses Reconsidering the Relationship Between Registration Laws, Education, and Voter Turnout. *American Politics Quarterly* **28**(1): 80–95.
Huber GP. 1991. Organizational Learning: The Contributing Processes and the Literatures. *Organization Science* **2**(1): 88–115.
Jacobides MG. 2005. Industry Change through Vertical Disintegration: How and Why Markets Emerged in Mortgage Banking. *Academy of Management Journal* **48**(3): 465–498.
Jacobs M, Droge C, Vickery SK, Calantone R. 2011. Product and Process Modularity's Effects on Manufacturing Agility and Firm Growth Performance. *Journal of Product Innovation Management* **28**(1): 123–137.
Jones TC. 1984. Reusability in Programming: A Survey of the State of the Art. *IEEE Transactions on Software Engineering* **SE-10**(5): 488 –494.

Keupp MM, Palmié M, Gassmann O. 2012. The Strategic Management of Innovation: A Systematic Review and Paths for Future Research. *International Journal of Management Reviews* **14**(4): 367–390.
Kotabe M, Parente R, Murray JY. 2007. Antecedents and outcomes of modular production in the Brazilian automobile industry: a grounded theory approach. *Journal of International Business Studies* **38**(1): 84–106.
Krippendorff K. 2004. *Content Analysis: An Introduction to Its Methodology*. SAGE: Thousand Oaks, California.
Langlois RN. 1992. External Economies and Economic Progress: The Case of the Microcomputer Industry. *The Business History Review* **66**(1): 1–50.
Langlois RN. 2002. Modularity in technology and organization. *Journal of Economic Behavior & Organization* **49**(1): 19–37.
Langlois RN. 2006. The secret life of mundane transaction costs. *Organization Studies* **27**(9): 1389–1410.
Langlois RN, Robertson PL. 1992. Networks and innovation in a modular system: Lessons from the microcomputer and stereo component industries. *Research Policy* **21**(4): 297–313.
Lawrence PR, Lorsch JW. 1967. Differentiation and Integration in Complex Organizations. *Administrative Science Quarterly* **12**(1): 1–47.
Lee J, Berente N. 2012. Digital Innovation and the Division of Innovative Labor: Digital Controls in the Automotive Industry. *Organization Science* **23**(5): 1428–1447.
Leiblein MJ, Miller DJ. 2003. An Empirical Examination of Transaction- and Firm-Level Influences on the Vertical Boundaries of the Firm. *Strategic Management Journal* **24**(9): 839–859.
Levitt B, March JG. 1988. Organizational Learning. *Annual Review of Sociology* **14**(1): 319–338.
Linden G, Somaya D. 2003. System-on-a-chip integration in the semiconductor industry: industry structure and firm strategies. *Industrial and Corporate Change* **12**(3): 545–576.
MacCormack A, Baldwin C, Rusnak J. 2012. Exploring the duality between product and organizational architectures: A test of the 'mirroring' hypothesis. *Research Policy* **41**(8): 1309–1324.
MacCormack A, Rusnak J, Baldwin CY. 2006. Exploring the Structure of Complex Software Designs: An Empirical Study of Open Source and Proprietary Code. *Management Science* **52**(7): 1015 –1030.
MacDuffie JP. 2008. Technological and Organizational Barriers to Modularity: Persistent Integrality in the Global Automotive Industry. Minneapolis, MN.
Mikkola JH. 2003. Modularity, component outsourcing, and inter-firm learning. *R&D Management* **33**(4): 439–454.
Miller RE, Sawyers D. 1968. *The technical development of modern aviation*. Routledge & K. Paul: London.
Monteverde K. 1995. Technical Dialog As an Incentive for Vertical Integration in the Semiconductor Industry. *Management Science* **41**(10): 1624–1638.
Moore GE. 1965. Cramming More Components onto Integrated Circuits. *Electronics* : 114–117.

Müller M. 2000. Abstraction benchmarks and performance of C++ applications. In *Proceedings of the Fourth International Conference on Supercomputing in Nuclear Applications*.
Nijmeijer KJ, Fabbricotti IN, Huijsman R. 2013. Making Franchising Work: A Framework Based on a Systematic Review. *International Journal of Management Reviews* : n/a–n/a.
Novak S, Eppinger SD. 2001. Sourcing by design: Product complexity and the supply chain. *Management Science* **47**(1): 189–204.
Orton JD, Weick KE. 1990. Loosely Coupled Systems: A Reconceptualization. *The Academy of Management Review* **15**(2): 203–223.
Pahl G, Beitz W, Wallace K. 1996. *Engineering Design: Systematic Approach*. Springer-Verlag GmbH: Berlin, Germany.
Parnas DL. 1972. On the criteria to be used in decomposing systems into modules. *Commun. ACM* **15**(12): 1053–1058.
Pierce BC. 2002. *Types and Programming Languages*. MIT Press: Cambridge, MA.
Pittaway L, Robertson M, Munir K, Denyer D, Neely A. 2004. Networking and innovation: a systematic review of the evidence. *International Journal of Management Reviews* **5**(3-4): 137–168.
Prahalad CK, Hamel G. 1990. The Core Competence of the Corporation. *Harvard Business Review* **68**(3): 79–91.
Priem RL. 2007. A Consumer Perspective on Value Creation. *The Academy of Management Review* **32**(1): 219–235.
Priem RL, Li S, Carr JC. 2012. Insights and New Directions from Demand-Side Approaches to Technology Innovation, Entrepreneurship, and Strategic Management Research. *Journal of Management* **38**(1): 346–374.
Prieto-Diaz R. 1993. Status report: software reusability. *IEEE Software* **10**(3): 61 –66.
Ravasi D, Stigliani I. 2012. Product Design: A Review and Research Agenda for Management Studies. *International Journal of Management Reviews* **14**(4): 464–488.
Ro YK, Liker JK, Fixson SK. 2007. Modularity as a Strategy for Supply Chain Coordination: The Case of U.S. Auto. *IEEE Transactions on Engineering Management* **54**(1): 172–189.
Ro YK, Liker JK, Fixson SK. 2008. Evolving models of supplier involvement in design: The deterioration of the Japanese model in US auto. *IEEE Transactions on Engineering Management* **55**(2): 359–377.
Romme AGL. 2003. Making a Difference: Organization as Design. *Organization Science* **14**(5): 558–573.
Rosenberg N. 1976. *Perspectives on Technology*. Cambridge University Press: New York.
Rousseau DM, Manning J, Denyer D. 2008. Chapter 11: Evidence in Management and Organizational Science: Assembling the Field's Full Weight of Scientific Knowledge Through Syntheses. *Academy of Management Annals* **2**(1): 475–515.
Sahaym A, Steensma HK, Schilling MA. 2007. The Influence of Information Technology on the Use of Loosely Coupled Organizational Forms: An Industry-Level Analysis. *Organization Science* **18**(5): 865–880.

Sako M. 2004. Modularity and outsourcing: The nature of co-evolution of product architecture and organisation architecture in the global automotive industry. In *The Business of Systems Integration*, Prencipe A, Davies (eds). Oxford University Press: New York: 229–253.
Sanchez R. 1995. Strategic flexibility in product competition. *Strategic Management Journal* **16**(S1): 135–159.
Sanchez R. 1999. Modular Architectures in the Marketing Process. *Journal of Marketing* **63**: 92–111.
Sanchez R, Galvin P, Bach N. 2013, October 1. Towards a Reverse Mirroring Hypothesis. Available at: http://hdl.handle.net/10398/8758.
Sanchez R, Mahoney JT. 1996. Modularity, Flexibility, and Knowledge Management in Product and Organization Design. *Strategic Management Journal* **17**: 63–76.
Sanchez R, Mahoney JT. 2013. Modularity and Economic Organization: Concepts, Theory, and Observations. In *Handbook of Economic Organization: Integrating Economic and Organization Theory*, Grandori A (ed). Edward Elgar: Northampton, MA: 383–399.
Schaller RR. 1997. Moore's law: past, present and future. *IEEE Spectrum* **34**(6): 52–59.
Schilling M. 2002. Modularity in multiple disciplines. In *Managing in the Modular Age: Architectures, Networks, and Organizations*, Garud R, Kumaraswamy A, Langlois R (eds). Wiley: New York.
Schilling MA. 2000. Toward a General Modular Systems Theory and Its Application to Interfirm Product Modularity. *The Academy of Management Review* **25**(2): 312–334.
Scott ML. 2000. *Programming Language Pragmatics*. Morgan Kaufmann: San Francisco, CA.
Shibata T, Yano M, Kodama F. 2005. Empirical analysis of evolution of product architecture: Fanuc numerical controllers from 1962 to 1997. *Research Policy* **34**(1): 13–31.
Simon HA. 1962. The Architecture of Complexity. *Proceedings of the American Philosophical Society* **106**(6): 467–482.
Sosa ME, Eppinger SD, Rowles CM. 2004. The Misalignment of Product Architecture and Organizational Structure in Complex Product Development. *Management Science* **50**(12): 1674–1689.
Staudenmayer N, Tripsas M, Tucci CL. 2005. Interfirm modularity and its implications for product development. *Journal of Product Innovation Management* **22**(4): 303–321.
Sturgeon TJ. 2002. Modular production networks: a new American model of industrial organization. *Industrial and Corporate Change* **11**(3): 451–496.
Suarez FF. 2004. Battles for technological dominance: an integrative framework. *Research Policy* **33**(2): 271–286.
Suddaby R. 2010. Editor's Comments: Construct Clarity in Theories of Management and Organization. *Academy of Management Review* **35**(3): 346–357.
Suh NP. 1990. *The Principles of Design*. Oxford University Press: New York.
Suh NP. 2001. *Axiomatic Design: Advances and Applications*. Oxford University Press: New York.

Suh NP. 2005. *Complexity: Theory And Applications*. Oxford University Press: New York.
Susarla A, Barua A, Whinston AB. 2010. Multitask Agency, Modular Architecture, and Task Disaggregation in SaaS. *Journal of Management Information Systems* **26**(4): 87–117.
Takeishi A. 2002. Knowledge partitioning in the interfirm division of labor: The case of automotive product development. *Organization Science* **13**(3): 321–338.
Taura T, Yoshikawa H. 1994. Managing Function Concepts in the Design Process. In *Management of Design*, Dasu S, Eastman C (eds). Springer Netherlands: 179–203. Available at: http://link.springer.com/chapter/10.1007/978-94-011-1390-8_10.
Thompson JD. 1967. *Organizations in Action: Social Science Bases of Administrative Theory*. Transaction Publishers: New Brunswick, NJ.
Tidd J. 1995. Development of novel products through intraorganizational and interorganizational networks the case of home automation. *Journal of Product Innovation Management* **12**(4): 307–322.
Tiwana A. 2008a. Does technological modularity substitute for control? A study of alliance performance in software outsourcing. *Strategic Management Journal* **29**(7): 769–780.
Tiwana A. 2008b. Does interfirm modularity complement ignorance? A field study of software outsourcing alliances. *Strategic Management Journal* **29**(11): 1241–1252.
Tranfield D, Denyer D, Smart P. 2003. Towards a Methodology for Developing Evidence-Informed Management Knowledge by Means of Systematic Review. *British Journal of Management* **14**(3): 207–222.
Tushman ML, Anderson P. 1986. Technological Discontinuities and Organizational Environments. *Administrative Science Quarterly* **31**(3): 439–465.
Tushman ML, Nadler DA. 1978. Information Processing as an Integrating Concept in Organizational Design. *The Academy of Management Review* **3**(3): 613–624.
Ulrich K. 1994. Fundamentals of Product Modularity. In *Management of Design*, Dasu S, Eastman C (eds). Springer Netherlands: 219–231. Available at: http://link.springer.com/chapter/10.1007/978-94-011-1390-8_12.
Ulrich K. 1995. The role of product architecture in the manufacturing firm. *Research Policy* **24**(3): 419–440.
Utterback JM, Abernathy WJ. 1975. A dynamic model of process and product innovation. *Omega* **3**(6): 639–656.
Veldhuizen TL. 2004. Optimization for Domain-Specific Programming. In *Domain-Specific Program Generation: International Seminar, Dagstuhl Castle, Germany, March 23-28, 2003, Revised Papers*, Lengauer C (ed). Springer: New York: 307–323.
Viswanathan M. 2005. *Measurement Error and Research Design*. SAGE: Thousand Oaks, CA.
Weick KE. 1976. Educational Organizations as Loosely Coupled Systems. *Administrative Science Quarterly* **21**(1): 1–19.
Weick KE. 1979. *The Social Psychology of Organizing*, 2nd edition. McGraw-Hill Humanities/Social Sciences/Languages: New York.

Whittemore R, Knafl K. 2005. The integrative review: updated methodology. *Journal of Advanced Nursing* **52**(5): 546–553.
Williamson OE. 1975. *Markets and hierarchies, analysis and antitrust implications: a study in the economics of internal organization*. Free Press: New York.
Williamson OE. 1985. *The economic institutions of capitalism: firms, markets, relational contracting*. Free Press: New York.
Woodward J. 1965. *Industrial organization: theory and practice*. Oxford University Press: Oxford, UK.
Worren N, Moore K, Cardona P. 2002. Modularity, strategic flexibility, and firm performance: a study of the home appliance industry. *Strategic Management Journal* **23**(12): 1123–1140.
Zelner BA. 2009. Using simulation to interpret results from logit, probit, and other nonlinear models. *Strategic Management Journal* **30**(12): 1335–1348.
Zirpoli F, Becker MC. 2011. The limits of design and engineering outsourcing: performance integration and the unfulfilled promises of modularity. *R&D Management* **41**(1): 21–43.
Zirpoli F, Camuffo A. 2009. Product architecture, inter-firm vertical coordination and knowledge partitioning in the auto industry. *European Management Review* **6**(4): 250–264.

APPENDIX

Argyres N, Bigelow L. 2010. Innovation, Modularity, and Vertical Deintegration: Evidence from the Early U.S. Auto Industry. *Organization Science* **21**(4): 842–853.

Baiman S, Fischer PE, Rajan MV. 2001. Performance Measurement and Design in Supply Chains. *Management Science* **47**(1): 173–188.

Baldwin C, Hippel E von. 2011. Modeling a Paradigm Shift: From Producer Innovation to User and Open Collaborative Innovation. *Organization Science* **22**(6): 1399–1417.

Baldwin CY. 2008. Where do transactions come from? Modularity, transactions, and the boundaries of firms. *Industrial and Corporate Change* **17**(1): 155–195.

Baldwin CY, Clark KB. 1997. Managing in an Age of Modularity. *Harvard Business Review* **75**(5): 84–93.

Bapna R, Barua A, Mani D, Mehra A. 2010. Research Commentary—Cooperation, Coordination, and Governance in Multisourcing: An Agenda for Analytical and Empirical Research. *Information Systems Research* **21**(4): 785–795.

Batallas DA, Yassine AA. 2006. Information Leaders in Product Development Organizational Networks: Social Network Analysis of the Design Structure Matrix. *IEEE Transactions on Engineering Management* **53**(4): 570–582.

Blackburn JD, Hoedemaker G, Van Wassenhove LN. 1996. Concurrent software engineering: prospects and pitfalls. *IEEE Transactions on Engineering Management* **43**(2): 179–188.

Bonjour E, Micaelli J-P. 2010. Design Core Competence Diagnosis: A Case From the Automotive Industry. *IEEE Transactions on Engineering Management* **57**(2): 323–337.

Browning TR. 2001. Applying the design structure matrix to system decomposition and integration problems: a review and new directions. *IEEE Transactions on Engineering Management* **48**(3): 292–306.

Brusoni S. 2005. The Limits to Specialization: Problem Solving and Coordination in ‘Modular Networks’. *Organization Studies* **26**(12): 1885–1907.

Brusoni S, Prencipe A. 2001. Managing Knowledge in Loosely Coupled Networks: Exploring the Links between Product and Knowledge Dynamics. *Journal of Management Studies* **38**(7): 1019–1035.

Brusoni S, Prencipe A. 2006. Making Design Rules: A Multidomain Perspective. *Organization Science* **17**(2): 179–189.

Brusoni S, Prencipe A, Pavitt K. 2001. Knowledge Specialization, Organizational Coupling, and the Boundaries of the Firm: Why Do Firms Know More than They Make? *Administrative Science Quarterly* **46**(4): 597–621.

Bush AA, Tiwana A, Rai A. 2010. Complementarities Between Product Design Modularity and IT Infrastructure Flexibility in IT-Enabled Supply Chains. *IEEE Transactions on Engineering Management* **57**(2): 240–254.

Cabigiosu A, Camuffo A. 2012. Beyond the ‘Mirroring’ Hypothesis: Product Modularity and Interorganizational Relations in the Air Conditioning Industry. *Organization Science* **23**(3): 686–703.

Cabigiosu A, Zirpoli F, Camuffo A. 2013. Modularity, interfaces definition and the integration of external sources of innovation in the automotive industry. *Research Policy* **42**(3): 662–675.

Carmel E, Espinosa JA, Dubinsky Y. 2010. ‘Follow the Sun’ Workflow in Global Software Development. *Journal of Management Information Systems* **27**(1): 17–37.

Ceci F, Masciarelli F. 2010. A Matter of Coherence: The Effects of Offshoring of Intangibles on Firm Performance. *Industry & Innovation* **17**(4): 373–392.
Ceci F, Prencipe A. 2008. Configuring Capabilities for Integrated Solutions: Evidence from the IT Sector. *Industry & Innovation* **15**(3): 277–296.
Chanron V, Lewis K. 2005. A study of convergence in decentralized design processes. *Research in Engineering Design* **16**(3): 133–145.
Chesbrough HW, Teece DJ. 1996. Organizing for Innovation. (cover story). *Harvard Business Review* **74**(1): 65–73.
Christensen CM, Verlinden M, Westerman G. 2002. Disruption, disintegration and the dissipation of differentiability. *Industrial and Corporate Change* **11**(5): 955–993.
Chuma H. 2006. Increasing complexity and limits of organization in the microlithography industry: implications for science-based industries. *Research Policy* **35**(3): 394–411.
Dibiaggio L. 2007. Design complexity, vertical disintegration and knowledge organization in the semiconductor industry. *Industrial and Corporate Change* **16**(2): 239–267.
Ernst D. 2005. Limits to Modularity: Reflections on Recent Developments in Chip Design. *Industry & Innovation* **12**(3): 303–335.
Ethiraj SK. 2007. Allocation of inventive effort in complex product systems. *Strategic Management Journal* **28**(6): 563–584.
Ethiraj SK, Levinthal D. 2004. Modularity and Innovation in Complex Systems. *Management Science* **50**(2): 159–173.
Fixson SK, Park J-K. 2008. The power of integrality: Linkages between product architecture, innovation, and industry structure. *Research Policy* **37**(8): 1296–1316.
Furlan A, Cabigiosu A, Camuffo A. 2013. When the mirror gets misted up: Modularity and technological change. *Strategic Management Journal*: n/a–n/a.
Garud R, Kumaraswamy A. 1995. Technological and organizational designs for realizing economies of substitution. *Strategic Management Journal* **16**(S1): 93–109.
Gentry RJ, Elms H. 2009. Firm Partial Modularity and Performance in the Electronic Manufacturing Services Industry. *Industry & Innovation* **16**(6): 575–592.
Gokpinar B, Hopp WJ, Iravani SMR. 2010. The Impact of Misalignment of Organizational Structure and Product Architecture on Quality in Complex Product Development. *Management Science* **56**(3): 468–484.
Gosain S, Malhotra A, Sawy OAE. 2004. Coordinating for Flexibility in e-Business Supply Chains. *Journal of Management Information Systems* **21**(3): 7–45.
Grunwald R, Kieser A. 2007. Learning to Reduce Interorganizational Learning: An Analysis of Architectural Product Innovation in Strategic Alliances*. *Journal of Product Innovation Management* **24**(4): 369–391.
Helper S, Sako M. 2010. Management innovation in supply chain: appreciating Chandler in the twenty-first century. *Industrial and Corporate Change* **19**(2): 399–429.
Henderson RM, Clark KB. 1990. Architectural Innovation: The Reconfiguration of Existing Product Technologies and the Failure of Established Firms. *Administrative Science Quarterly* **35**(1): 9–30.
Herbsleb JD, Grinter RE. 1999. Architectures, coordination, and distance: Conway's law and beyond. *IEEE Software* **16**(5): 63–70.
Von Hippel E. 1990. Task partitioning: An innovation process variable. *Research Policy* **19**(5): 407–418.

Van Hoek R. 1998. Managing in an age of modularity. *Harvard Business Review* **76**(2): 168–169.
Hoetker G. 2006. Do modular products lead to modular organizations? *Strategic Management Journal* **27**(6): 501–518.
Hoetker G, Swaminathan A, Mitchell W. 2007. Modularity and the Impact of Buyer–Supplier Relationships on the Survival of Suppliers. *Management Science* **53**(2): 178–191.
Huang GQ, Zhang XY, Lo VHY. 2007. Integrated Configuration of Platform Products and Supply Chains for Mass Customization: A Game-Theoretic Approach. *IEEE Transactions on Engineering Management* **54**(1): 156–171.
Iansiti M. 1995. Technology integration: Managing technological evolution in a complex environment. *Research Policy* **24**(4): 521–542.
Iansiti M. 1997. From technological potential to product performance: an empirical analysis. *Research Policy* **26**(3): 345–365.
Jacobides MG. 2006. The architecture and design of organizational capabilities. *Industrial and Corporate Change* **15**(1): 151–171.
Jacobs M, Droge C, Vickery SK, Calantone R. 2011. Product and Process Modularity's Effects on Manufacturing Agility and Firm Growth Performance. *Journal of Product Innovation Management* **28**(1): 123–137.
Joglekar NR, Rosenthal SR. 2003. Coordination of Design Supply Chains for Bundling Physical and Software Products*. *Journal of Product Innovation Management* **20**(5): 374–390.
Kapoor R, Adner R. 2012. What Firms Make vs. What They Know: How Firms' Production and Knowledge Boundaries Affect Competitive Advantage in the Face of Technological Change. *Organization Science* **23**(5): 1227–1248.
Kotabe M, Parente R, Murray JY. 2007. Antecedents and Outcomes of Modular Production in the Brazilian Automobile Industry: A Grounded Theory Approach. *Journal of International Business Studies* **38**(1): 84–106.
Koushik MV, Mookerjee VS. 1995. Modeling Coordination in Software Construction: An Analytical Approach. *Information Systems Research* **6**(3): 220–254.
Krikke H, le Blanc I, van de Velde S. 2004. Product Modularity and the Design of Closed-Loop Supply Chains. *California Management Review* **46**(2): 23–39.
Kusunoki K, Nonaka I, Nagata A. 1998. Organizational Capabilities in Product Development of Japanese Firms: A Conceptual Framework and Empirical Findings. *Organization Science* **9**(6): 699–718.
Kwan I, Cataldo M, Damian D. 2012. Conway's Law Revisited: The Evidence for a Task-Based Perspective. *IEEE Software* **29**(1): 90–93.
Langlois RN. 2006. The Secret Life of Mundane Transaction Costs. *Organization Studies* **27**(9): 1389–1410.
Langlois RN, Garzarelli G. 2008. Of Hackers and Hairdressers: Modularity and the Organizational Economics of Open-source Collaboration. *Industry & Innovation* **15**(2): 125–143.
Langlois RN, Robertson PL. 1992. Networks and innovation in a modular system: Lessons from the microcomputer and stereo component industries. *Research Policy* **21**(4): 297–313.

Lea M, O'Shea T, Fung P. 1995. Constructing the Networked Organization: Content and Context in the Development of Electronic Communications. *Organization Science* **6**(4): 462–478.
Lee J, Berente N. 2012. Digital Innovation and the Division of Innovative Labor: Digital Controls in the Automotive Industry. *Organization Science* **23**(5): 1428–1447.
Lee J, Veloso FM. 2008. Interfirm Innovation under Uncertainty: Empirical Evidence for Strategic Knowledge Partitioning*. *Journal of Product Innovation Management* **25**(5): 418–435.
Lei D, Hitt MA, Goldhar JD. 1996. Advanced Manufacturing Technology: Organizational Design and Strategic Flexibility. *Organization Studies* **17**(3): 501–523.
Loch CH, Terwiesch C, Thomke S. 2001. Parallel and Sequential Testing of Design Alternatives. *Management Science* **47**(5): 663–678.
Lundqvist M, Sundgren N, Trygg L. 1996. Remodularization of a Product Line: Adding Complexity to Project Management. *Journal of Product Innovation Management* **13**(4): 311–324.
MacCormack A, Baldwin C, Rusnak J. 2012. Exploring the duality between product and organizational architectures: A test of the 'mirroring' hypothesis. *Research Policy* **41**(8): 1309–1324.
MacCormack A, Rusnak J, Baldwin CY. 2006. Exploring the Structure of Complex Software Designs: An Empirical Study of Open Source and Proprietary Code. *Management Science* **52**(7): 1015–1030.
Miozzo M, Grimshaw D. 2005. Modularity and innovation in knowledge-intensive business services: IT outsourcing in Germany and the UK. *Research Policy* **34**(9): 1419–1439.
Nagarajan A, Mitchell W. 1998. Evolutionary diffusion: internal and external methods used to acquire encompassing, complementary, and incremental technological changes in the lithotripsy industry. *Strategic Management Journal* **19**(11): 1063–1077.
Natter M, Mild A, Feurstein M, Dorffner G, Taudes A. 2001. The Effect of Incentive Schemes and Organizational Arrangements on the New Product Development Process. *Management Science* **47**(8): 1029–1045.
Novak S, Eppinger SD. 2001. Sourcing By Design: Product Complexity and the Supply Chain. *Management Science* **47**(1): 189–204.
O'Connor GC, DeMartino R. 2006. Organizing for Radical Innovation: An Exploratory Study of the Structural Aspects of RI Management Systems in Large Established Firms. *Journal of Product Innovation Management* **23**(6): 475–497.
Orsenigo L, Pammolli F, Riccaboni M. 2001. Technological change and network dynamics: Lessons from the pharmaceutical industry. *Research Policy* **30**(3): 485–508.
Ozman M. 2010. The Knowledge Base of Products: Implications for Organizational Structures. *Organization Studies* **31**(8): 1129–1154.
Pentland BT, Feldman MS. 2007. Narrative Networks: Patterns of Technology and Organization. *Organization Science* **18**(5): 781–795.
Pisano GP, Shih WC. 2012. Does America Really Need Manufacturing? *Harvard Business Review* **90**(3): 94–102.
Pisano GP, Teece DJ. 2007. How to Capture Value from Innovation: Shaping Intellectual Property and Industry Architecture. *California Management Review* **50**(1): 278–296.
Press K, Geipel MM. 2010. Vanishing hands? On the link between product and organization architecture. *Industrial and Corporate Change* **19**(5): 1493–1514.

Rauscher TG, Smith PG. 1995. From experience time-driven development of software in manufactured goods. *Journal of Product Innovation Management* **12**(3): 186–199.
Richard PJ, Devinney TM. 2005. Modular Strategies: B2B Technology and Architectural Knowledge. *California Management Review* **47**(4): 86–113.
Ro YK, Liker JK, Fixson SK. 2007. Modularity as a Strategy for Supply Chain Coordination: The Case of U.S. Auto. *IEEE Transactions on Engineering Management* **54**(1): 172–189.
Ro YK, Liker JK, Fixson SK. 2008. Evolving Models of Supplier Involvement in Design: The Deterioration of the Japanese Model in U.S. Auto. *IEEE Transactions on Engineering Management* **55**(2): 359–377.
Rosenkopf L, Schilling MA. 2007. Comparing alliance network structure across industries: observations and explanations. *Strategic Entrepreneurship Journal* **1**(3-4): 191–209.
Sahaym A, Steensma HK, Schilling MA. 2007. The Influence of Information Technology on the Use of Loosely Coupled Organizational Forms: An Industry-Level Analysis. *Organization Science* **18**(5): 865–880.
Sanchez R. 1995. Strategic flexibility in product competition. *Strategic Management Journal* **16**(S1): 135–159.
Sanchez R, Mahoney JT. 1996. Modularity, Flexibility, and Knowledge Management in Product and Organization Design. *Strategic Management Journal* **17**: 63–76.
Schmickl C, Kieser A. 2008. How much do specialists have to learn from each other when they jointly develop radical product innovations? *Research Policy* **37**(6–7): 1148–1163.
Seki K, Nishimura H. 2011. A module-based thermal design approach for distributed product development. *Research in Engineering Design* **22**(4): 279–295.
Shibata T, Yano M, Kodama F. 2005. Empirical analysis of evolution of product architecture: Fanuc numerical controllers from 1962 to 1997. *Research Policy* **34**(1): 13–31.
Siggelkow N, Levinthal DA. 2003. Temporarily Divide to Conquer: Centralized, Decentralized, and Reintegrated Organizational Approaches to Exploration and Adaptation. *Organization Science* **14**(6): 650–669.
Sinha KK, Ven AHV de. 2005. Designing Work Within and Between Organizations. *Organization Science* **16**(4): 389–408.
Sosa M. 2008. A structured approach to predicting and managing technical interactions in software development. *Research in Engineering Design* **19**(1): 47–70.
Sosa ME, Eppinger SD, Pich M, McKendrick DG, Stout SK. 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 ME, Eppinger SD, Rowles CM. 2004. The Misalignment of Product Architecture and Organizational Structure in Complex Product Development. *Management Science* **50**(12): 1674–1689.
Srikanth K, Puranam P. 2011. Integrating distributed work: comparing task design, communication, and tacit coordination mechanisms. *Strategic Management Journal* **32**(8): 849–875.
Staudenmayer N, Tripsas M, Tucci CL. 2005. Interfirm Modularity and Its Implications for Product Development*. *Journal of Product Innovation Management* **22**(4): 303–321.
Sturgeon TJ. 2002. Modular production networks: a new American model of industrial organization. *Industrial and Corporate Change* **11**(3): 451–496.

Subramanyam R, Ramasubbu N, Krishnan MS. 2012. In Search of Efficient Flexibility: Effects of Software Component Granularity on Development Effort, Defects, and Customization Effort. *Information Systems Research* **23**(3-Part-1): 787–803.
Susarla A, Barua A, Whinston AB. 2010. Multitask Agency, Modular Architecture, and Task Disaggregation in SaaS. *Journal of Management Information Systems* **26**(4): 87–117.
Tafti A, Mithas S, Krishnan MS. 2013. The Effect of Information Technology–Enabled Flexibility on Formation and Market Value of Alliances. *Management Science* **59**(1): 207–225.
Takeishi A. 2002. Knowledge Partitioning in the Interfirm Division of Labor: The Case of Automotive Product Development. *Organization Science* **13**(3): 321–338.
Tanriverdi H, Konana P, Ge L. 2007. The Choice of Sourcing Mechanisms for Business Processes. *Information Systems Research* **18**(3): 280–299.
Thomke S, Reinertsen D. 1998. Agile Product Development: Managing Development Flexibility in Uncertain Environments. *California Management Review* **41**(1): 8–30.
Tidd J. 1995. Development of novel products through intraorganizational and interorganizational networks the case of home automation. *Journal of Product Innovation Management* **12**(4): 307–322.
Tidd J. 2001. Innovation management in context: environment, organization and performance. *International Journal of Management Reviews* **3**(3): 169–183.
Tiwana A. 2008a. Does interfirm modularity complement ignorance? A field study of software outsourcing alliances. *Strategic Management Journal* **29**(11): 1241–1252.
Tiwana A. 2008b. Does technological modularity substitute for control? A study of alliance performance in software outsourcing. *Strategic Management Journal* **29**(7): 769–780.
Tripathy A, Eppinger SD. 2011. Organizing Global Product Development for Complex Engineered Systems. *IEEE Transactions on Engineering Management* **58**(3): 510–529.
Ulrich K. 1995. The role of product architecture in the manufacturing firm. *Research Policy* **24**(3): 419–440.
Verganti R, Buganza T. 2005. Design Inertia: Designing for Life-Cycle Flexibility in Internet-Based Services*. *Journal of Product Innovation Management* **22**(3): 223–237.
Wang Q, von Tunzelmann N. 2000. Complexity and the functions of the firm: breadth and depth. *Research Policy* **29**(7–8): 805–818.
Westerman G, McFarlan FW, Iansiti M. 2006. Organization Design and Effectiveness over the Innovation Life Cycle. *Organization Science* **17**(2): 230–238.
Wolter C, Veloso FM. 2008. The Effects of Innovation on Vertical Structure: Perspectives on Transaction Costs and Competences. *Academy of Management Review* **33**(3): 586–605.
Worren N, Moore K, Cardona P. 2002. Modularity, strategic flexibility, and firm performance: a study of the home appliance industry. *Strategic Management Journal* **23**(12): 1123–1140.
Yoo Y, Henfridsson O, Lyytinen K. 2010. Research Commentary—The New Organizing Logic of Digital Innovation: An Agenda for Information Systems Research. *Information Systems Research* **21**(4): 724–735.
Zyglidopoulos S. 1999. Initial Environmental Conditions and Technological Change. *Journal of Management Studies* **36**(2): 241–262.