INSEAD

Faculty & Research

The Misalignment of Product Architecture and Organizational Structure in Complex Product Development

by

M. Sosa
S. Eppinger
and
C. Rowles

2003/68/TM

(Revised Version of 2002/76/TM)

Working Paper Series

The Misalignment of Product Architecture and Organizational Structure in Complex Product Development

Manuel E. Sosa[1]
INSEAD, Technology Management Area
Boulevard de Constance, 77300 Fontainebleau, France

Steven D. Eppinger
Sloan School of Management
Massachusetts Institute of Technology, Cambridge, Massachusetts 02139

Craig M. Rowles
Advance Engine Program
Pratt & Whitney Aircraft, East Hartford, Connecticut 06108

ABSTRACT

Product architecture knowledge is typically embedded in the communication patterns of established development organizations, which hinders their capability to cope with novel architectures, especially when developing complex products. Yet, structured methods addressing this issue are lacking. Previous research has studied complex product development from two separate perspectives, product architecture and organizational structure. Our research combines these two viewpoints into a structured approach to study how design interfaces in the product architecture map onto communication patterns within the development organization. Specifically, we study the following cases: 1) known design interfaces not addressed by team interactions and 2) observed team interactions not predicted by design interfaces. We hypothesize how organizational and system boundaries, system modularity, design interface strength, and indirect interactions impact the alignment of design interfaces and team interactions. This paper is the first to bring together design structure matrix representation and social network methods for statistical modeling and hypothesis testing. Our results offer important managerial insights into factors to "watch out for" when dealing with interdependences across organizational and functional boundaries. In particular, we show how the effects of organizational and system boundaries are moderated by system modularity, strength of design interfaces and indirect team interactions. The research uses data collected from a large commercial aircraft engine development process.

Keywords: Product Architecture; Product Development Organizations; Technical Communication; Design Structure Matrix; Statistical Network Analysis.

---

[1] Corresponding author: Tel: ++33-1-60-72-4536. Fax: ++33-1-60-74-6179. E-mail: *manuel.sosa@insead.edu*

1. Introduction

Understanding how organizations manage the knowledge associated with the architecture of the products they design is a critical challenge for firms developing complex products. As highlighted by Henderson and Clark (1990, p. 9), "architectural knowledge tends to become embedded in the structure and information-processing procedures of established organizations", which hinders organizational capability to deal with novel architectures. Hence, innovation managers must understand how they manage the architectural knowledge of the products they currently develop. This is especially relevant in complex product development due to the large number of both physical components and design participants involved in the process. Unfortunately, methods, frameworks, and/or tools available to address this challenge are scarce.

Consider the typical job of a design engineer during the development of a complex product such as an aircraft engine. Typically, design engineers are part of cross-functional design teams organized around different components of the product (Robertson and Allen 1992, Pimmler and Eppinger 1994, McCord and Eppinger 1993). For example, during the design phase of an aircraft engine development, some teams design the components of the fan system, other teams conceive the components of the low- and high-pressure compressors while yet other teams develop the components of the low- and high-pressure turbines. During the design phase, the team responsible for designing any engine component (e.g., the blades of the low-pressure turbine) needs to balance the technical requests from other design teams in addition to managing its own design constraints (Mihm *et al.* 2003). Typically, design demands from other teams depend on the physical nature of the interfaces between the components of the engine. For example, when examining the design interfaces between the vanes and the blades of the low-pressure turbine studied in this paper we learned that there is a potential transfer of energy (vibration) from the vanes to the blades that needs to be avoided. Therefore, design teams designing those components are expected to interact to address such a design interface. (Refer to Sosa *et al.* 2003 for further technical details.) In general, managing the integration effort across design teams creates an interesting challenge for managers of complex development projects who typically raise the following questions: Are design teams "talking" about the right things? If not, why? Are all design interfaces between product components identified and addressed during the design phase? If not, why? Furthermore, how can managers answer these questions in a structured way for better complex product development?

The situation described above highlights the importance of not only identifying the *interfaces* between product components but also evaluating whether or not the corresponding teams carry out

interactions to address those interfaces properly. Of course, it is not difficult to argue that if two components share design interfaces, the teams that design such components need to interact (Thompson 1967, Galbraith 1973). However, in the development of highly complex products, it would be naïve to expect a perfect mapping between *design interfaces* and *team interactions*. Hence, we want to investigate the factors that prevent such mapping occurring. Below are the key research questions we address in this paper:

1. Can we expect any level of misalignment of product architecture and organizational integration effort? If so, how can we uncover it?
2. What factors may impact such a misalignment? More specifically:
    - Why do some design interfaces between product components not correspond to technical interactions between the design teams that develop such components?
    - Why do some technical interactions between design teams take place even though no design interface is identified between the components they design?

Investigating these questions is crucial to our understanding of how product and organizational structures align during the design phase of complex development efforts. Previous research has studied complex product development separately, from two important perspectives: the product architecture and the organizational structure. Rather, we bring these two perspectives together to examine how interfaces between product components map onto interactions between teams designing these components.

This paper offers three important contributions. From a theoretical viewpoint, we integrate, for the first time, two separate streams of research to hypothesize factors that impact the alignment of product architecture and organizational integration effort. From a methodological perspective, the research method presented in this paper involves a structured approach to examine the alignment of product and organizational domains in complex products, and thus far, uncover any misalignment between them. Doing so has important managerial implications from strategic, project planning, and project execution viewpoints. From an analytical point of view, we apply a novel statistical network analysis technique (based on p^* models of Wassserman and Pattison 1996) to rigorously test hypothesized effects while controlling for dyadic and triadic tendencies typically embedded in network data.

The remainder of the article is structured as follows: Section 2 presents an integrated literature review. We describe the theoretical framework for hypothesis development in Section 3, and present our research approach in Section 4. Section 5 details the empirical study. We present the statistical

modeling and hypothesis testing in Section 6. In Section 7, we discuss the results and threats to their validity. In Section 8, we summarize theoretical and managerial implications and conclude the article.

2. Literature Review on Complex Product Development

The literature on product architecture focuses on how the product is decomposed into functional and physical elements and the impact of such decomposition to the operations of the firm. On the other hand, the organizational literature emphasizes the need for integrating the design effort of organizational units to successfully develop a product. We argue that, in order to capture a complete view of a product development system, we need to study both perspectives simultaneously, which is the purpose of this work. First, however, let us examine the key findings of these two literature streams and how they relate to our work.

2.1. Product Architecture Perspective

The focus of this stream of research is the product itself, considered as "a complex assembly of interacting components". (Krishnan and Ulrich 2001, p. 3). Ulrich (1995) defines product architecture as "the scheme by which the function of a product is allocated to physical components". Furthermore, based on how functions map onto physical components we can distinguish modular and integral product architectures (Ulrich and Eppinger 2000).

Engineering design literature has explored design methods to map functional and physical elements of the product (e.g., Kirschman and Fadel 1998, Hirtz *et al.* 2002) and the impact of product architecture decisions on the family of products a firm develops (e.g., Gonzalez-Zugasti *et al.* 2000, Simpson *et al.* 2001). More recently, we extended the concepts of modular and integral products to the system level by introducing a new notion of system modularity based upon the way components share design interfaces across systems (Sosa *et al.* 2003).

Operations and management science literature has investigated the link between product architecture and the supply chain (e.g., Ulrich and Ellison 1999, Gupta and Krishnan 1999, Novak and Eppinger 2001), the impact of product modularity on platform planning (Robertson and Ulrich 1998, Krishnan *et al.* 1999), and the effect of product modularity on experimentation and problem solving strategies (Thomke 1998, Loch *et al.* 2001, Mihm *et al.* 2003).

In order to study the structure of product architectures in terms of component interactions, we use the design structure matrix (DSM) tool, an analytical method introduced by Steward (1981) and used by Eppinger *et al.* (1994) to study interdependence between product development activities. DSM representation has also been used to document product decomposition (Pimmler and Eppinger 1994) and team interdependence (McCord and Eppinger 1993) separately. Based upon these results, we

formulate a method to study the degree of alignment of product and organizational interdependences. Furthermore, this is the first study to extend the DSM literature by building statistical models based on social network methods for rigorous hypothesis testing using DSM data.

Although previous work has advanced our understanding of architectural knowledge and its impact on some operational aspects of the firm, the explicit link between product architecture and organizational structure has been largely neglected. Although most of the architectural knowledge is explicit and known by development organizations, some interfaces between components are tacit or unknown and only discovered during the design process itself. We believe that by simultaneously analyzing the design network of components and the communication network of design teams we are able to uncover those tacit design interfaces and achieve a more complete view of the architecture of the product.

2.2. Organizational Perspective

Adopting the information-processing viewpoint, Brown and Eisenhardt (1995) summarize that "frequent and appropriately structured task communication" results in better performing development processes. They also highlight the fact that "the principal shortcoming of this perspective is that it is *so focused on communication* by project team members that other factors (e.g., organization of the work, *product attributes*, market attractiveness) are neglected." (p. 359, emphasis is added). By comparing product architecture linkages with technical interactions between design teams, we aim to address this deficit in this stream of literature.

A large body of research focuses on the communication process in development organizations, much of which has studied how factors such as physical distance, organizational structures, task structures, and communication media affect technical communication (e.g., Allen 1977, Griffin and Hauser 1992, Morelli *et al.* 1995, Sosa *et al.* 2002). Allen's "distance-communication" curve, showing the negative effects of distance on technical communication in R&D organizations, is one of the best-known empirical findings on this topic (Allen 1977, p. 239). Morelli *et al.* (1995) compare the actual communication network of a development organization with a predicted communication network based on the task structure of the project. The main drawback of these empirical findings is their lack of statistical conclusion validity due to the limited independence between observations used. Van den Bulte and Moenaert (1998) addressed such a limitation by using statistical modeling techniques for sequential network data (Wasserman and Iacobucci 1988) to examine how collocation affects communication patterns in a development organization.

Organizational researchers have empirically studied coordination mechanisms in product development organizations that practice concurrent engineering (e.g., Adler 1995, Eisenhardt and Tabrizi 1995, Terwiesch *et al.* 2002), but none of these studies explicitly considers the impact of product architecture on coordination strategies. To the best of our knowledge, no previous work has rigorously compared the communication network of a development organization with the network of components of the systems they design, which is the focus of this paper.

Recent research has also studied the role of communication when sharing knowledge in multi-unit organizations. Hansen (1999) argues that weak ties (distant and infrequent communications) are better for searching knowledge of low complexity, whereas strong ties (close and frequent communications) are better for transferring knowledge of high complexity. With the same empirical data set, Hansen (2002) shows that technical knowledge can flow through intermediary units, particularly when it is well codified. Tsai (2002) studies the tension between coordination and competition when sharing knowledge in a large multiunit company. This stream of research highlights the importance of established relationships to effectively transfer knowledge across organizational units (e.g., Gupta and Govindarajan 2000). Since these studies examine knowledge sharing across different units in large firms, they do not consider the structure of the product as an important determinant in intraorganizational knowledge sharing. The main influence of this work on our study is that, contrary to other studies on innovation, it considers the flow of technical information through trustful intermediary units.

The organizational literature on product innovation considers products as hierarchically arranged sets of subsystems with defined interfaces (e.g., Alexander 1964, Henderson and Clark 1990, Sanchez and Mahoney 1996, Baldwin and Clark 2000, Gatignon *et al.* 2002). This line of research has examined the impact of the architecture of the product on the innovation process from a strategic viewpoint. Henderson and Clark (1990) based their theory about why established firms fail when facing architectural innovation on the observation that, as products evolve, information filters and communication channels develop and help engineers to solve problems efficiently. These mechanisms also inhibit recognition of potentially novel product architectures and the development of new architectural knowledge. Sanchez and Mahoney (1996) discuss the notion of modular organization to hypothesize that standardized design interfaces between components in a product design provide a "means to embed coordination of loosely coupled component development processes" (p. 66). Baldwin and Clark (2000) discuss the strategic benefits of developing modular products. Gatignon *et al.* (2002) use a structural approach to distinguish several types of innovations, their locus in the product, and their main characteristics. This stream of research emphasizes the importance of providing methods for

managers to understand technical communication patterns and how they map onto the architecture of the product under development, which is precisely the purpose of this work. In addition, we challenge previous results that favor product modularity by cautioning managers about the potential negative effects that system modularity has during the design phase of complex product development.

3. Design Interfaces and Team Interactions

We seek to understand the association between product architectures and the technical communication patterns of development organizations. More specifically, we study how linkages between physical components (*design interfaces*) map onto linkages between design teams (*technical team interactions*) during the detailed design phase of complex development projects.

In the product architecture domain, we define a *design interface* between component *i* and component *j* as component *i* depending upon component *j* for functionality. That is, component *j* either imposes geometry constraints or transfers forces, material, energy, and/or signals to component *i* in order for component *i* to function properly. In the organizational domain, we define *team interaction* between design team *i* and design team *j* as team *i* requesting technical information directly from team *j* during the detailed design phase of the development process. Note that our definitions for both design interface and team interaction imply a direction. That is, component *i*'s functionality is affected by component *j*, and technical information flows from team *j* to team *i* (See Figure 1).

We argue that, during the detailed design phase of a complex development effort, design interfaces are the primary source of team interdependence. Hence, for projects where the task structure is of the form "*team i* designs *component i*" (which is typical in complex product development), it is not difficult to argue that the existence (or absence) of a design interface between component *i* and component *j* should correspond to the existence (or absence) of technical interaction of team *i* with team *j*. (See Figure 2.) Note that our variable of interest is whether or not design interfaces and team interactions are aligned. Hence, we are not claiming causality but association. That is, although we believe design interfaces drive team interactions for the most part, we are also open to the possibility of team interactions determining some design interfaces.

The lower-left and upper-right cells of Figure 2 represent the expected cases. *Aligned presence of design interfaces and team interactions* corresponds to the established interdependencies where we found both team interactions and design interfaces. Similarly, *aligned absence of design interfaces and team interactions* corresponds to the lack of interdependence, where we found neither team interactions nor design interfaces. The lower-right and upper-left cells of Figure 2 represent the unexpected cases, which are far more interesting, and therefore the focus of our study. *Unmatched*

design interfaces correspond to design interfaces that are not addressed by direct team interactions whereas *unmatched team interactions* correspond to communication between teams whose components do not explicitly share design interfaces.

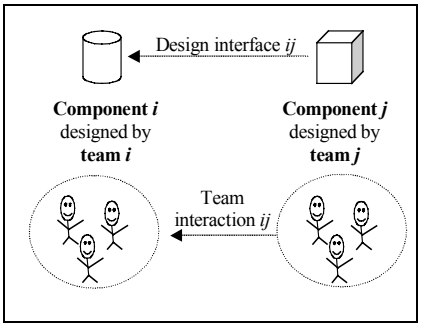


Figure1. Design interface and team interaction

Team Interactions	NO	**Unmatched design interfaces**	**Aligned absence** of interfaces and interactions
	YES	**Aligned presence** of interfaces and interactions	**Unmatched team interactions**
		YES	NO
		Design Interfaces	

Figure 2. Mapping Design Interfaces and Team Interactions

3.1. Understanding the Misalignment of Design Interfaces and Team Interactions

To the best of our knowledge, previous research has largely ignored the significant existence of unmatched design interfaces and unmatched team interactions. However, we believe that considering these cases is important because their existence would indicate that not all known product-related interdependences are necessarily addressed by direct technical communication, and that technical communication (where not expected) may uncover "unknown" product-related interdependences.

There are two types of factors that may prevent alignment of design interfaces and team interactions. First, d*ynamic factors,* which refer to how previous and future development efforts may affect the likelihood of encountering misaligned cases (Henderson and Clark 1990, Adler 1995, Sanchez and Mahoney 1996.) Although dynamic factors are important, we focus this study on understanding *static* effects. We discuss in Section 7.1 (threats to validity) how, by leaving the consideration of dynamic effects for future work, we do not compromise the results presented in this paper. *Static effects* refers to the factors related to both the current product architecture and organizational structure that impact the likelihood of misalignment of design interfaces and team interactions. We consider four types of static effects: boundaries, modularity, criticality, and indirect ties. We elaborate on each of these hypothesized effects in the following sub-sections.

3.1.1. Effects of Organizational and System Boundaries

Complex product development requires both decomposing the product into systems and components, and splitting the organization into groups of design teams (Ulrich and Eppinger, 2000). As a result of such a physical and organizational breakdown, boundaries are formed between

components that belong to different systems (system boundaries) and between design teams that belong to different groups of teams (organizational boundaries).

Previous research on R&D management suggests that organizational boundaries between functional groups impose communication barriers which inhibit cross-team interactions (Allen 1977, Tushman 1977, Griffin and Hauser 1992, Van den Bulte and Moenaert 1998). People within these groups are subjected to organizational bonds that promote the development of a language and an identity inherent to the group to which they belong. As a result, the higher the degree of specialization of groups, the higher the communication barriers across them. In fact, Van den Bulte and Moenaert (1998) found that functional boundaries (between R&D, marketing, and manufacturing) hinder cross-group communication independently of physical distance. Accordingly, in complex product development projects, we expect organizational boundaries to significantly reduce the number of cross-boundary team interactions. However, does this mean that we should expect a significantly larger proportion of unmatched design interfaces across boundaries? We think so. In our framework, lack of "apparently necessary" technical communication is reflected by the existence of unmatched design interfaces. Since organizational boundaries hinder team interaction, we expect design teams to exhibit a lower tendency to communicate about cross-boundary design interfaces than about within-boundary interfaces. Hence, based on previous findings, we would envisage a higher likelihood of encountering unmatched design interfaces across organizational boundaries.

In the product architecture domain, components are typically clustered into systems, and so we expect to find a significantly larger proportion of design interfaces within system boundaries. This may impose architectural knowledge barriers which inhibit explicit identification of design interfaces across systems by the design experts (Henderson and Clark 1990, Sanchez and Mahoney 1996). Nevertheless, in order to develop working systems, design teams find that they need to interact, and do so, which results in unmatched team interactions. Hence, since system barriers prevent the identification of some existent (but unknown) design interfaces, we expect a higher likelihood of encountering unmatched team interactions across system boundaries.

Considering the effects of organizational and system boundaries described above, we posit the following hypothesis to test:

H1: Misalignment of design interfaces and team interactions is more likely to take place across (organizational/system) boundaries than within boundaries.

3.1.2. Effects of System Modularity

We can apply the concept of modular and integral product architectures to the many systems and sub-systems within a complex product. In Sosa *et al.* (2003), we define modular and integrative systems based on how components share design interfaces across system boundaries. We define *modular systems* as "those whose design interfaces with other systems are clustered among a few physically adjacent systems", whereas *integrative systems* are "those whose design interfaces span all, or most of, the systems that comprise the product due to their physically distributed or functionally integrative nature throughout the product". (Sosa *et al.* 2003, p. 240).

Considering this notion of system modularity, we question whether system modularity moderates the effects of organizational/system boundaries. That is, does system modularity support or hinder the alignment of design interfaces and team interactions across boundaries?

Previous research suggests that the information structure of a development organization depends on the type of product architecture they design (Henderson and Clark 1990, Sanchez and Mahoney 1996). Given the physically distributed or functionally integrative nature of integrative systems, we anticipate integrative design teams to be more accustomed to crossing organizational boundaries than design teams that develop modular systems. As a result, we argue that, for complex systems the hindering effects of organizational/system boundaries are more severe between modular systems. That is, we expect to observe a significantly higher number of unmatched design interfaces and unmatched team interactions across modular systems. Hence, we posit the following hypothesis to test:

H2: System modularity prevents the alignment of design interfaces and team interactions across (system and organizational) boundaries. That is, misalignment of design interfaces and team interactions across boundaries is more likely to occur between components that belong to different modular systems than with components that belong to integrative systems.

3.1.3. Effects of Design Interface Strength

The degree of task interdependence describes the degree to which tasks require collective action (Thompson 1967). The greater the degree of task interdependence, the greater the coordinative and innovative information requirements (Galbraith 1973). This is consistent with previous research that has shown that a greater degree of task interdependence leads to greater communication (e.g., Adler 1995). Moreover, when studying the task structure of development projects, Smith and Eppinger (1997) use the strength of task interdependency to identify the set of activities that require greater effort to coordinate. Considering that in many complex development efforts, a significant proportion of the task structure directly maps onto the product structure under development (i.e., *task i* is defined as

"designing component *i*"), we expect stronger design interfaces (i.e., those interfaces more critical for the functionality of product components) to be more likely aligned with team interactions. Thus far, we expect to find empirical support for the following hypothesis:

H3: Low criticality of design interface strength prevents the alignment of design interfaces and team interactions.

H1 posits that organizational boundaries hinder the alignment of design interfaces and team interactions, whereas H3 suggests that greater component interdependence favors the occurrence of such alignment. We then ask: Which effect is stronger? We argue it is the latter. That is, we claim that teams address a significantly smaller proportion of design interfaces across boundaries (H1), but those that are aligned to team interactions are the ones that are perceived to be more critical for the functionality of the components they design (H3). This is consistent with Tushman (1977) who suggests that specialized gatekeepers "may not attend to all external communication areas, but may specialize in those external areas most critical to the work of their unit" (p. 592).

Moreover, evidence from the telecommunications industry suggests that communication barriers imposed by geographical separation are alleviated between teams that are highly interdependent (Sosa *et al.* 2002). Extending this insight further, we argue that the strength of the design interface may also help managers overcome the hindering effects of organizational boundaries. That is, we expect engineers to cross their organizational and functional boundaries to address critical design interfaces with components that are designed by teams outside their organization. Hence, we posit the following hypothesis to test:

H4: The stronger the design interface the more likely the alignment of design interface and team interaction ACROSS boundaries.

3.1.4. Indirect team interactions and indirect design interfaces

We refer to *indirect team interactions* as technical information flow that takes place between two teams through an intermediary design team. Research in social networks has long supported the notion of indirect communication via intermediary units (Granovetter 1973), and more recently, research about knowledge sharing has also considered indirect relations to conduit information in multiunit firms (e.g., Hansen 1999, 2002). Although early work in product innovation identified the organizational benefits of having a gatekeeper who could gather relevant information from the team's external environment and pass it to the rest of the team (e.g., Allen 1977, Tushman 1977), indirect interaction between design teams has been largely neglected as a coordination mechanism to address their interdependence in product development organizations (e.g., Adler 1995, Terwiesch *et al.* 2002).

We introduce the concept of indirect team interaction to hypothesize that team *i* whose component has a design interface with component *j* may not report direct interaction with team *j* because it interacts with an intermediary team (team *k,* which also interacts with team *j*) which passes the information (to team *i*) that would otherwise have flowed directly from team *j* to team *i*. Hence, we expect a higher likelihood of finding unmatched design interfaces between teams that communicate indirectly through intermediary teams. As a result, we posit the following hypothesis to test:

H5: Indirect team interactions are likely to address unmatched design interfaces.

In the product architecture domain, we refer to *indirect design interfaces* as the indirect impact of component *j* over component *i* through an intermediary component *k.* Although the notion of considering the product as a set of interrelated elements is well established (Krishnan and Ulrich 2001), no previous work in the product architecture literature explicitly considers the impact of other components through intermediary components. Similarly to the case of indirect team interactions, we hypothesize that the existence of intermediary components between two components that do not share a direct design interface increases the likelihood that the corresponding design teams interact, resulting in an unmatched team interaction. Hence, we posit the following hypothesis to test:

H6: Indirect design interfaces are likely to be addressed by unmatched team interactions.

4. Research Approach

This section describes our method of comparing and analyzing the architecture of a product with its development organization. Our approach involves four major steps (Figure 3):

1) **Identify design interfaces**. By interviewing design experts who have a deep understanding of the architecture of the product, we identify how the product is decomposed into systems, and how these are further decomposed into components. We then ask the experts to identify the types and criticality of the design dependencies between all the components. We represent this network of component dependencies in a *design interface matrix*.
2) **Identify team interactions**. We identify the design teams responsible for developing each of the product's components. We then survey key members of each team to capture the intensity of the technical interactions between them. We represent the communication network of the development organization in a *team interaction matrix*.
3) **Map design interfaces and team interactions**. We compare the design interface matrix with the team interaction matrix and capture this comparison in the *alignment matrix*. When each design team is responsible for the design of only one physical component, the alignment matrix is

obtained by overlaying the identically sequenced design interface matrix and team interaction matrix. Refer to Sosa (2000) for details of comparing products and organizations whose mapping is not one-to-one.

4) **Analyze the alignment matrix**. We use statistical network analysis techniques to rigorously analyze the patterns exhibited in the alignment matrix and test hypothesized effects that may systematically cause a significant misalignment of design interfaces and team interactions.

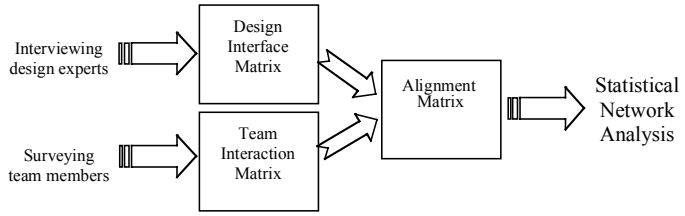


Figure 3. Our Research Approach

5. The Study

We applied our approach to study the detail design period of the development of a large commercial aircraft engine, the PW4098 engine developed by Pratt & Whitney. Several factors justified the selection of this project to study. First, the project was a complex design that exhibited explicit decomposition of the engine into systems, and these into components. Second, the assignment of a single design team to each component facilitated the implementation of our approach. Third, the model studied was the most recent engine program to complete design and development, and almost all team members involved in the detail design development phase were still accessible. Fourth, the development organization was over 90% collocated in a single facility, so effects of geographical boundaries were minimal. Finally, the engine studied was part of a family of large commercial engines with two more derivatives planned, whose development programs had the potential to gain directly from this study (See Rowles 1999 for details).

5.1. Identifying design interfaces

The engine studied was decomposed into eight systems (See Figure 4). Each of these systems was further decomposed into five to ten components each, for a total of 54 components. Six of the eight systems were identified as *modular systems*, whereas the other two systems (mechanical components system and externals and controls system) were recognized as *integrative systems* because of the physically distributed and functionally integrative features of their components (Sosa *et al.* 2003).

Fig 4. PW4098 Commercial Aircraft Engine Studied

After documenting the general decomposition of the product, we identified design interfaces between the 54 components of the engine. We distinguished five types of design dependencies to define the design interfaces between the physical components (Table 1). In addition, we used a five-point scale to capture the level of criticality of each dependency for the overall functionality of the component in question (Table 2). We discuss these metrics at length in Sosa *et al.* (2003). We use the type and criticality of design interfaces to assess their strength. Thus, we define the *design interface strength* as the sum of the absolute criticality of each design dependency for each interface. Hence,

$$[\text{design interface strength}]_{i,j} = \sum \left| c_{ij}^{d} \right|$$

where,

d = dependency type = [spatial, structural, material, energy, information]

c_{ij}^{d} = level of criticality for design interface (i,j) of type d = [-2,-1,0,+1,+2]

Table 1. Types of design dependency

Dependency Type	Description
Spatial	Functional requirement related to physical adjacency for alignment, orientation, serviceability, assembly, or weight.
Structural	Functional requirement related to transferring loads, or containment
Energy	Functional requirement related to transferring heat, vibration, electric, or noise energy
Material	Functional requirement related to transferring airflow, oil, fuel, or water
Information	Functional requirement related to transferring signals or controls

Table 2. Level of criticality of design dependencies

Level of criticality	Description
Required (+2)	Dependency is necessary for functionality
Desired (+1)	Dependency is beneficial, but not absolutely necessary for functionality
Indifferent (0)	Dependency does not affect functionality
Undesired (-1)	Dependency causes negative effects, but does not prevent functionality
Detrimental (-2)	Dependency must be prevented to achieve functionality

For the 569 non-zero design interfaces, the mean (sd) of *design interface strength* was 4.41 (1.92). Similar to network studies that consider valued ties (Granovetter 1973, Marsden 1990), we define an indicator variable, $STRENGTH_{ij}$, which trichotomizes the criticality of the design interface as follows:

$STRENGTH_{ij}$ = NULL = 0 if $[\text{design interface strength}]_{ij} = 0$

$STRENGTH_{ij}$ = WEAK = 1 if $0 < [\text{design interface strength}]_{ij} \leq 4$

$STRENGTH_{ij}$ = STRONG = 2 if $[\text{design interface strength}]_{ij} > 4$

Under this categorization, we determined 319 WEAK interfaces and 250 STRONG interfaces. This is consistent with observations of complex products in which there are fewer critical interfaces than less important ones (Smith and Eppinger 1997). We considered alternative definitions of $STRENGTH_{ij}$ which not only resulted in a skewed distribution of non-zero design interfaces, but more importantly, they were somewhat limited in capturing both type and criticality of the design interfaces. We also

completed categorical data analyses with the alternative definitions of $STRENGTH_{ij}$ and the results were consistent with the ones reported in this paper.

We mapped the design interface data into a trichotomous design interface matrix (See Figure 5). The off-diagonal entries of the matrix are marked with either a "W" or "S" to indicate the existence of a WEAK or STRONG design interface, respectively, between two components. Reading across a row corresponding to a particular component indicates the other components with which it has interfaces. Note that the matrix is not symmetric due to the fact that each row captures the dependencies necessary for one component's functions. We purposely sequenced the matrix by clustering the components that belong to the same system together so that heavy boxes along the diagonal indicate the eight system boundaries of the engine. (For further details refer to Sosa *et al.* 2003).

Figure 5. Design Interface Matrix

5.2. Identifying team interactions

The organization responsible for the development of the aircraft engine was structured into 60 design teams exclusively dedicated to the project. Fifty-four of these teams were responsible for developing the 54 components of the engine, and were grouped into eight system-design groups mirroring the architecture of the engine studied. The remaining six design teams were system integration teams, who had no specific hardware assigned to them, and whose responsibility was to assure that the engine worked as a whole. Examples are the rotordynamics and secondary flow teams.

In order to assess the integration effort of the organization, we captured the intensity of the technical interactions between the design teams involved in the development process. We focused on capturing task-related interactions between design teams, referred to as "coordination-type communication" by Allen (1977) and Morelli *et al.* (1995). Similar to previous work that captures technical communication data (e.g., Allen 1977) and social network studies (Marsden 1990), we surveyed key members of 57 of the 60 design teams. The three teams whose responses were missing were the interface team at Boeing (a system integration team), one team in the high-pressure compressor group and one team in the externals and controls system group. Responses were obtained from at least two key members of each design team surveyed, including the team leaders. We asked each design team to rate the peak frequency and criticality of their technical interactions with each of the other teams during the ten-month detailed design phase of the engine development project. We used a six-point scale that combines the frequency and criticality of each interaction into a single

metric called interaction intensity.[2] The criticality component of our metric allows asymmetry in the interaction intensity of each pair of design teams.

We conducted follow-up interviews with respondents to verify reported data. Many interactions originally reported were eliminated from the data set because they were social or advise-related rather than coordination-type communication. After consulting with other team leaders, and based on empirical evidence suggesting that technical interactions are likely to be reciprocal (Morelli *et al.* 1995), we assumed reciprocal interactions for the teams whose responses were missing. We also completed our analyses without these components/teams and the results were consistent with those reported here. After data purification was completed, we identified a total of 656 non-zero technical team interactions within the development organization, with mean (sd) intensity of 2.02 (1.58), of which 423 interactions were between the 54 component teams.

Consistent with Marsden and Campbell (1984), using a scale with both criticality and frequency of interactions, helped respondents focus their responses on information exchanges that actually took place (i.e., the "way it was", not the "way it should have been"). However, similar to previous research in technical communication (e.g., Allen 1977, Van den Bulte and Moenaert 1998), we chose the presence or absence of *significant* information exchange as the binary variable of interest. We define *significant* information exchange as those technical interactions that were relevant during the design phase due to their criticality and/or frequency. Such interactions are captured by a non-zero interaction intensity in our scale. We organized the team interaction data into a square (60 x 60) team interaction matrix, whose off-diagonal cells marked "O" indicate each *significant* team interaction revealed.

Figure 6. Team Interaction Matrix (Binary)

5.3. Mapping design interfaces and team interactions

The one-to-one assignment of the 54 components to the 54 design teams allows the direct comparison of the design interface matrix with the team interaction matrix. Hence, by overlaying the design interface matrix with the team interaction matrix, we obtain the alignment matrix exhibited in Figure 7. Note that we omitted the six integration teams from this analysis. These teams interact with almost every component design team in the organization, which prevents us from infering any particular communication pattern in which they were involved. (See also sub-Section 7.1).

Figure 7. Alignment Matrix

---

[2] This is consistent with Marsden and Campbell (1984) who found closeness or intensity as best indicators of an unobserved tie strength. More recently, Hansen (1999, 2002) combined frequency and closeness into a single metric called *interunit tie weakness*.

The alignment matrix provides the basis for the analysis completed to test the hypotheses posed above. Figure 8 exhibits the possible states for each cell of the alignment matrix. As expected, the majority of the cases (90% of the cells) are aligned presence, or absence, of design interfaces and team interactions ("#" and "blank" cells, respectively). The unexpected cases accounted for 10% of the cells; these were the *unmatched design interfaces* (220 "W" or "S" cells, or 39% of the 569 design interfaces), and the *unmatched team interactions* (74 "O" cells, or 17% of the 423 team interactions).

Team Interactions	NO (2439)	**W or S** (220)	(2219)
	YES (423)	**#** (349)	**O** (74)
		YES (569)	NO (2293)
		Design Interfaces	

Figure 8. Overall Results

5.4. Descriptive Categorical Data Analysis

The objective of our analysis is to study how certain factors such as organizational/system boundaries, system modularity, design interface strength, and indirect ties influence the alignment of design interfaces and team interaction characterized by the four states illustrated in Figure 8. We first complete a descriptive categorical data analysis (See Table 3) based on chi-square tests of independence and homogeneity which make the strong assumption of independence between cells of both the design interface matrix and team interaction matrix. (See Sosa 2000 for details.) The results presented in Table 3 are just a first order approximation to hypothesis testing; hence the conclusions listed are only preliminary. In order to effectively test whether the observed differences are larger than what one would expect from random variation in the data, we need to control for other factors that are typically embedded in network data. We turn to this task next.

6. Statistical Network Analysis

Similar to social network data, in our data set, each component and design team appears as many times as they share interfaces or interact with others, resulting in observations that are clearly not independent. Moreover, by visually inspecting both the design interface matrix and the team interaction matrix, we can observe strong tendencies for reciprocation of ties as well as clustering of ties within boundaries. Other tendencies that can be present in our data are propensities of components and teams to generate or attract linkages. Such deviations from randomness embedded in our network data make our statistical analysis problematic. We tackled this challenge by considering two statistical network approaches: *log-linear* p_1 and *logit* p^* analyses. Note that the use of quadratic assignment

procedure (QAP), commonly used in organizational studies, is not appropriate in our research problem, given the discrete nature of our variable of interest and the strong tendency for reciprocation and clustering exhibited by our data (Krackhardt, 1988).

Table 3. Results of Descriptive Categorical Data Analysis

Hypothesis	Sample	Results	$\chi 2$	Preliminary Conclusion
H1a: Effect of organizational boundaries	569 design interfaces	81% of the 231 within-boundary design interfaces were matched by team interactions, whereas 48% of the 338 cross-boundary design interfaces were matched by team interactions	63.101	H1 tentatively supported [a]
H1b: Effect of systems boundaries	423 team interactions	90% of the 208 within-boundary team interactions were predicted by design interfaces, whereas 75% of the 215 cross-boundary team interactions were predicted by design interfaces	15.517	H1 tentatively supported [a]
H2a: Moderating effect of system modularity on organizational boundaries	338 cross-boundary design interfaces	54% of the 228 cross-boundary design interfaces with integrative systems were matched by team interactions, whereas 36% of the 110 cross-boundary design interfaces between modular systems were matched by team interactions	8.740	H2 tentatively supported [a]
H2b: Moderating effect of system modularity on system boundaries	215 cross-boundary team interactions	81% of the 150 cross-boundary team interactions with integrative teams were predicted by design interfaces, whereas 62% of the 65 cross-boundary team interaction between modular teams were predicted by design interfaces	9.566	H2 tentatively supported [a]
H3: Effect of design interface strength	569 design interfaces	72% of the 250 strong design interfaces were matched by team interactions, whereas 53% of the 319 weak design interfaces were matched by team interactions	21.385	H3 tentatively supported [a]
H4: Moderating effect of design interface strength	338 cross-boundary design interfaces	54% of the 115 strong cross-boundary design interfaces were matched by team interactions, whereas 45% of the 223 weak cross-boundary design interfaces were matched by team interactions	2.501	H4 tentatively NOT supported [a]
H5: Effect of indirect team interactions	2439 potential cases for indirect team interactions	20% of the 459 cases with more than one potential indirect team interaction corresponded to unmatched design interfaces, whereas 10% of the 579 cases with one potential indirect team interactions corresponded to unmatched design interfaces, and only 5% of the 1401 cases with no potential indirect interactions corresponded to unmatched design interfaces	93.911	H5 tentatively supported [b]
H6: Effect of indirect design interfaces	2293 potential cases for indirect interfaces	7% of the 645 cases with more than three potential indirect interfaces corresponded to unmatched team interactions, whereas 4% of the 643 cases with less than three potential indirect interfaces corresponded to unmatched team interaction, and less than 1% of the 1005 cases with no indirect interfaces corresponded to unmatched team interaction.	51.561	H6 tentatively supported [b]

a: The null hypothesis is rejected when χ^2 is greater than the critical $\chi^2_{(0.99,1)}=6.635$

b: The null hypothesis is rejected when χ^2 is greater than the critical $\chi^2_{(0.99,2)}=9.210$

6.1. *Log-linear p_1* analysis

We built log-linear models of the alignment matrix based on the p_1 distribution introduced by Holland and Leinhardt (1981). Much research has been done focusing on extending the p_1 model to analyze social network data (e.g., Fienberg and Wasserman 1981, Fienberg *et al.* 1985, Wasserman and Iacobucci 1988). Similar to Van den Bulte and Moenaert (1998), we used these results to build, in five steps, a log-linear model for the probabilities of the dyads of our alignment matrix to test H1 and H2. Subsequently, we developed more elaborated *log-linear p_1* models that consider discrete design interfaces (Wasserman and Iacobucci 1986) to test the effects of design interface strength (H3 and H4). Although this statistical modeling approach suits well our research problem, its independence dyad

assumption is quite limiting and unrealistic. Moreover, since these models do not explicitly handle triadic effects, we are not able to use them to test our hypotheses concerning indirect relations (H5 and H6). Nonetheless, we used this analysis to cross-validate the results obtained from our main statistical approach, the *logit p** analysis described next. Details of our *log-linear* p_1 analysis are available as an optional appendix available from the authors upon request.

6.2. *Logit p** analysis

In order to address the limitations of p_1 models, a new generation of exponential family models, *p*,* have been developed by Wasserman and Pattison (1996), based on the Markov random graphs of Frank and Strauss (1986). These models not only release the independent dyad assumption but also allow researchers to formulate them in a standard *response-explanatory variables* form in which the response variable is the log odds (or logit) of the probability that a network tie is present, and the explanatory variables can be either any hypothesized network structures or network actor attributes.

The general log-linear form of *p** can be expressed as (Wasserman and Pattison, 1996):

$$\Pr(\mathbf{X}=\mathbf{x}) = \frac{\exp\{\theta' z(\mathbf{x})\}}{\kappa(\theta)} = \frac{\exp\{\theta_1 z_1(\mathbf{x}) + ... + \theta_n z_n(\mathbf{x})\}}{\kappa(\theta)} \qquad (1)$$

where **x** is the observed *matrix* of the network of interest. The response variable is the probability of the observed **x**. θ is a vector of n model parameters and **z**(**x**) is the vector of the n explanatory variables. The θ parameters are unknown "regression-type" coefficients that must be estimated. The function $\kappa(\theta)$ is a constant that ensures a proper probability distribution (i.e., the sum of Pr(**X**=**x**) over all possible **X** is unity).

The problem with this formulation is that the constant κ is very difficult to determine analytically and computationally for most networks (except for very small ones). Hence, a logit formulation for *p** models that does not depend on the normalizing constant has been developed for single dichotomous relation networks (Strauss and Ikeda 1990, Wasserman and Pattison 1996). As a result, Wasserman and Pattison (1996) define the log of conditional odds as follows:

$$\varpi_{ij} = \log\left\{\frac{\Pr(X_{ij}=1 \mid X_{ij}^c)}{\Pr(X_{ij}=0 \mid X_{ij}^c)}\right\} = \theta'\left[z(x_{ij}^+) - z(x_{ij}^-)\right] = \theta'\delta(x_{ij}) \qquad (2)$$

This expression is discussed at length by Wasserman and Pattison (1996). Of interest is the term $\delta(x_{ij})$, which is the vector of explanatory variables that surfaces when the tie *ij* changes from 1 to 0. As indicated by Wasserman and Pattison (1996, p. 407), "to specify a *logit p** model, one chooses *a priori* a collection of network statistics that is supposed to affect the log odds of a tie being present to absent". Hence, the model depends on the network effects that one believes to have a significant

tendency of being present in the network. For each network effect (such as expansiveness, reciprocation, or transitivity), there is a corresponding network statistic and a corresponding explanatory variable in the logit model. It is important to emphasize that the explanatory variable is the change in the network statistic when the tie from element *i* to element *j* (X_{ij}) changes from being present to absent. Wasserman and Pattison (1996) describe how by explicitly assuming a dependence structure between the ties in a network the independence dyad assumption is no longer needed. Extensions of the *logit p** model for multivariate and valued relations are presented by Pattison and Wasserman (1999) and Robins *et al.* (1999), respectively.

Having introduced the *logit p** model for single dichotomous relation, our next task is to build specific members of the *logit p** family to properly model our alignment matrix and test our hypotheses. We carry out this task in two stages. First, based on Pattison and Wasserman (1999), we rewrite equation (2) for two binary relations by considering the dichotomous random variable X_{ijm} which records whether tie *ij* of type *m* is present. In our case, *m*=1 corresponds to design interfaces while *m*=2 corresponds to team interactions. With such models, we test hypotheses H1, H2, H5, and H6. We can express our *logit p** for two binary relations as follows (Pattison and Wasserman 1999):

$$\varpi_{ijm} = \log\left\{\frac{\Pr(X_{ijm}=1 \mid X^{c}_{ijm})}{\Pr(X_{ijm}=0 \mid X^{c}_{ijm})}\right\} = \theta'\left[z(x^{+}_{ijm}) - z(x^{-}_{ijm})\right] = \theta'\delta(x_{ijm}) \qquad (3)$$

In the second stage, we extend our formulation to incorporate the effects of design interface strength (to test hypotheses H3 and H4) based on Robins *et al.* (1999). When considering trichotomous design interfaces, our original X_{ijm} array will have *1s* and *2s* for *m*=1. As a result, we need to transform our newly defined trichotomous X_{ijm} array into a three-way binary array, $Y_{ij,m}$, in which the third dimension has three states (*m*=1w, *m*= 1s, and *m*= 2). That is, *m*=1w corresponds to WEAK design interfaces, while *m*=1s corresponds to STRONG design interfaces, and *m*=2 corresponds to team interactions. Hence, the transformation takes the following form:

$Y_{ij,2} = X_{ij2}$

If $X_{ij1} = 0$, then $Y_{ij,1w} = Y_{ij,1s} = 0$

If $X_{ij1} = 1$, then $Y_{ij,1w} = 1$ and $Y_{ij,1s}$ is undefined

If $X_{ij1} = 2$, then $Y_{ij,1s} = 1$ and $Y_{ij,1w}$ is undefined

As a result of such a transformation, we can write the *logit p** model using our newly defined $Y_{ij,m}$ which purposely excludes "response" variables for which $Y_{ij,m}$ is undefined. Hence, we write the expression for the *logit p** for the $Y_{ij,m}$ array as follows:

$$\varpi_{ij,m} = \log\left\{\frac{\Pr(Y_{ij,m}=1 \mid Y^{c}_{ij,m})}{\Pr(Y_{ij,m}=0 \mid Y^{c}_{ij,m})}\right\} = \gamma'\left[z(y^{+}_{ij,m}) - z(y^{-}_{ij,m})\right] = \gamma'\delta(y_{ij,m}) \qquad (4)$$

We complete the rest of our *logit p** analysis in three steps: (1) defining substantial network effects to include in our models; (2) estimating parameters corresponding to each network effect by fitting our *logit p** models to the observed data using standard logistic regressions; (3) interpreting parameters obtained from logistic regressions.

6.2.1. Hypothesized network effects

Our *logit p** formulation includes the basic dyadic and triadic effects typical of network data (Anderson *et al.* 1999, p. 46). Table 4 describes the network effects for each type (i.e., m=1 for design interfaces, and m=2 for team interactions) with their corresponding parameters and their associated network statistic. We also include network effects that result from considering the two networks simultaneously; these are the bivariate effects captured by our alignment matrix.

In order to test our hypotheses we define structural variables as $ACROSS_{ij}$ and $MODULAR_{ij}$ to capture whether tie ij is across boundaries and between modular systems, respectively. Using these structural variables and the bivariate network effects described in Table 4, we define formal tests for our hypotheses as follows:

H1: $\theta_{ACROSS,12} < 0$ H2: $\theta_{ACROSS,MODULAR,12} < 0$ H3: $\theta_{STRONG,2} > \theta_{WEAK,2}$

H4: $\theta_{ACROSS,STRONG,2} > \theta_{ACROSS,WEAK,2}$ H5: $\tau_{221} > 0$ H6: $\tau_{112} > 0$

6.2.2. Parameter estimation

Fitting a *logit p** to data can be done (albeit approximately) by adopting the pseudo-likelihood estimation strategy originally proposed by Strauss and Ikeda (1990) and further discussed by Wasserman and Pattison (1996), Pattison and Wasserman (1999), and Robins *et al.* (1999). This fitting approach assumes that the logits, ϖ_{ijm}, of the conditional probabilities are statistically independent. Hence, maximizing the pseudo-likelihood function is equivalent to fitting a logistic regression model to the logits, ϖ_{ijm}, using standard computing packages. (We used SPSS 11.0.) It is important to recall that the explanatory variables in the logistic regressions are the *difference* network statistics, that is, the change in network statistic from tie (ij,m) being present to being absent. Hence, before fitting any of the models, we need to pre-process the observed data (documented in our team interaction and design interface matrices) to calculate the change statistic for each relational tie X_{ijm}.[3]

[3] We modified SPSS codes available at http://kentucky.psych.uiuc.edu/pstar/index.html to calculate the change statistics of interest for the models corresponding to Equations (3) and (4). The codes are available from the authors upon request.

Following Anderson *et al.* (1999, p. 49), we assess (approximately) the statistical importance of any explanatory variable by evaluating the difference in pseudo-likelihood ratio statistics (G_{PL}^2) by referring it to the appropriate χ^2 distribution. We also evaluate (approximately) the significance of each parameter by comparing their pseudo-Wald statistics (Wald_{PL}) to the appropriate χ^2 distribution.

Table 4. Network effects included in *logit p models**

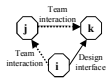
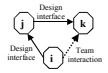
Network effect label	Description	Graphical configuration	Parameters	Network statistic $z(\mathbf{x})$
Dyadic effects				
Choice	For m=1, propensity of components to have a design interface with others For m=2, propensity of teams to report interactions with others		$\theta_1 \quad \theta_2$	$L_m = \sum_{ij} X_{ij,m}$
Reciprocation (or mutuality)	Configurations in which ties ij and ji are both present for each network of type m		$\rho_1 \quad \rho_2$	$M_m = \sum_{i<j} X_{ij,m} X_{ji,m}$
Triadic effects				
2-In-star (overall attractiveness)	Configurations in which two separate ties are directed towards the same element. This effect captures the overall tendency for each element to attract ties of type m		$\sigma_{I,1} \quad \sigma_{I,2}$	$S_{I,m} = \sum_{i,j,k} X_{ji,m} X_{ki,m}$
2-Out-star (overall expansiveness)	Configurations in which two separate ties are directed away from the same element. This effect captures the overall tendency for each element to generate ties of type m		$\sigma_{O,1} \quad \sigma_{O,2}$	$S_{O,m} = \sum_{i,j,k} X_{ij,m} X_{ik,m}$
2-mixed-star	Configurations in which a tie is directed away from an element to which another tie is directed to		$\sigma_{M,1} \quad \sigma_{M,2}$	$S_{M,m} = \sum_{i,j,k} X_{ji,m} X_{ik,m}$
Transitivity	Configurations in which three elements form a transitive triad (such as ij, jk, and ik) of type m		$\tau_1 \quad \tau_2$	$T_{T,m} = \sum_{i,j,k} X_{ij,m} X_{jk,m} X_{ik,m}$
Cyclicity	Configurations in which three ties of type m form an intransitive cycle such as ij, jk, and ki		$\zeta_1 \quad \zeta_2$	$T_{C,m} = \sum_{i,j,k} X_{ij,m} X_{jk,m} X_{ki,m}$
Bivariate effects				
Exchange	Configurations in which a design interface is present and it is reciprocated by a team interaction, or vice versa		ρ_{12}	$M_{1,2} = \sum_{i<j} X_{ij,1} X_{ji,2}$
Association	Configurations in which both design interface and team interactions are present		θ_{12}	$L_{1,2} = \sum_{ij} X_{ij,1} X_{ij,2}$
Indirect team interactions	Configurations in which a transitive triad is formed by two subsequent team interactions with a design interface		τ_{221}	$T_{221} = \sum_{i,j,k} X_{ij,2} X_{jk,2} X_{ik,1}$
Indirect design interfaces	Configurations in which a transitive triad is formed by two subsequent design interfaces with a team interaction		τ_{112}	$T_{112} = \sum_{i,j,k} X_{ij,1} X_{jk,1} X_{ik,2}$

Table 5 shows the results of fitting six dichotomous bivariate *logit p** models to test H1, H2, H5, and H6. For each model, we show parameter estimates, number of parameters included in the model, and the model's G_{PL}^2 statistic. Table 5 exhibits three sets of parameters. The first two sets correspond to network effects of the design interface matrix and team interaction matrix, respectively, whereas the last set of parameters corresponds to network effects of the alignment matrix which are the ones whose statistical significance we want to infer.

Effects of design interface strength

Table 6 shows the results of fitting four trichotomous *logit p** models for testing H3 and H4. For brevity, Table 6 exhibits parameters corresponding to alignment network effects only. However, similar to the models shown in Table 5, each of the models in Table 6 also includes dyadic and triadic effects for weak and strong design interfaces and for team interactions.

Table 5. Results of *logit p analysis (dichotomous relations)**

Parameters	Model 1 Independent	Model2 Association	Model 3 Across	Model 4 Modular	Model 5 Indirect	Model 6 Indirect Within
Design Interface effects						
θ_1	-2.446 (91.259)	-3.046	-2.490	-2.701	-2.719	-1.877
ρ_1	3.246 (360.035)	2.871	2.846	2.819	2.804	2.711
$\sigma_{I,1}$	.021 (.890)	.000	.002	.003	-.030	-.028
$\sigma_{O,1}$	.020 (.754)	-.017	-.015	-.013	.034	.043
$\sigma_{M,1}$	-.129 (35.304)	-.084	-.074	-.072	-.084	-.089
τ_1	.267 (111.052)	.262	.259	.259	.283	.280
ζ_1	-.134 (3.903)	-.176	-.195	-.192	-.168	-.144
Team Interaction effects						
θ_2	-3.319 (177.838)	-4.178	-3.056	-3.753	-3.662	-3.948
ρ_2	3.815 (362.516)	3.231	3.021	2.972	2.964	2.944
$\sigma_{I,2}$	-.004 (.028)	-.023	.000	.012	-.072	-.079
$\sigma_{O,2}$	.063 (15.965)	.090	.111	.121	.148	.152
$\sigma_{M,2}$	-.092 (27.383)	-.072	-.045	-.038	-.033	-.029
τ_2	.334 (113.162)	.249	.212	.212	.227	.222
ζ_2	-.389 (20.849)	-.366	-.421	-.408	-.385	-.442
Alignment effects						
ρ_{12}		**-.242** (2.124)	-.224	-.198	-.220	-.166
θ_{12}		**2.769** (329.614)	1.917	1.807	2.091	2.227
Effects across boundaries (H1)						
$\theta_{ACROSS,1}$			**-.803** (9.864)	-.771	-.819	-1.738
$\theta_{ACROSS,2}$			**-1.895** (35.314)	-1.816	-1.819	-1.462
$\theta_{ACROSS,12}$			**1.013** (10.681)	1.375	1.376	1.265
Effects between modular systems (H2)						
$\theta_{MODULAR,1}$				**.230** (1.359)	.201	.196
$\theta_{MODULAR,2}$				**.681** (6.866)	.658	.701
$\theta_{MODULAR,12}$				**.117** (0.086)	.003	-.089
$\theta_{ACROSS,\ MODULAR,12}$				**-.749** (3.003)	-.704	-.687
Effects of indirect team interactions and indirect design interfaces (H5, H6)						
τ_{221}					**.003** (0.002)	.000
τ_{112}					**-.081** (2.641)	-.059
$\tau_{WITHIN,112}$						**.258** (7.183)
$\tau_{WITHIN,122}$						**-.337** (12.004)
Number of parameters	14	16	19	23	27	29
G_{PL}^2	2451.659	2021.192	1984.473	1974.203	1946.754	1934.936

$Wald_{PL}$ statistics are shown between parentheses. For approximate statistical inference we compare $Wald_{PL}$ against χ^2. Hence, $p < 0.1$ if $Wald_{PL} > 2.706$
Models 5 and 6 also include lower order parameters $\sigma_{I,12}$ and $\sigma_{O,12}$. For model 6, we define $WITHIN_{ij}$ to capture whether tie ij is within boundaries.

6.2.3. Parameter interpretation

In general, a significantly positive parameter indicates a tendency for the associated configuration to occur in the network, whereas negative parameters suggest a lack of presence of such an effect in the network. In Table 5, model 1 (independent) considers both team interaction and design interface matrices independently of each other. As expected, both networks exhibit strong tendencies for reciprocation. In addition, the tendency for transitivity is strongly exhibited by both networks, whereas the tendency for cyclicity is not likely to occur in either network. Model 2 (association) includes an

insignificant exchange parameter (ρ_{12}), which indicates that design interfaces are not likely to be reciprocated by team interactions (nor vice versa). Most of the improvement in fit of model 2 over model 1 is due to the significantly positive association parameter (θ_{12}), which indicates a strong general tendency for design interfaces and team interactions to be aligned.

Table 6. Results of *logit* p* analysis with trichotomous design interface strength

Parameter	Model 1 (Exchange)	Model 2 (Clustering)	Model 3 (Association)	Model 4 (Association across)
$\rho_{WEAK,2}$	**1.105** (55.724)	.950	-.264	-.187 (1.116)
$\rho_{STRONG2}$	**1.305** (54.383)	1.017	-.470	-.404 (3.234)
$\theta_{ACROSS,WEAK}$		**-1.151** (23.029)	-.663	-1.090 (14.743)
$\theta_{ACROSS,STRONG}$		**-1.445** (20.305)	-.876	-1.317 (10.223)
$\theta_{ACROSS,1}$		**-1.409** (37.008)	-1.193	-1.860 (35.149)
$\theta_{WEAK,2}$			**2.625** (259.769)[a]	1.718 (29.792)
$\theta_{STRONG,2}$			**3.078** (231.840) [a]	2.403 (46.674)
$\theta_{ACROSS,WEAK,2}$				**1.138 (**10.857) [b]
$\theta_{ACROSS,STRONG,2}$				**.877 (**4.938) [b]
Number of parameters	28	31	33	35
G_{PL}^2	2833.541	2755.417	2347.819	2335.305

$Wald_{PL}$ statistics are shown between parentheses. For approximate statistical inference we compare $Wald_{PL}$ against χ^2. Hence, $p < 0.1$ if $Wald_{PL} > 2.706$

a: To test that $\theta_{STRONG,2} > \theta_{WEAK,2}$ we estimate a reduced model with a single parameter ($\theta_{1,2}$) whose $G_{PL}^2 = 2352.179$. Hence, $\Delta G_{PL}^2 = 4.360$, $\Delta df = 1$, $p < 0.05$

b: To test that $\theta_{ACROSS,STRONG,2} > \theta_{ACROSS,WEAK,2}$ we estimate a reduced model with a single parameter ($\theta_{ACROSS,1,2}$) whose $G_{PL}^2 = 2335.637$. Hence, $\Delta G_{PL}^2 = 0.332$, $\Delta df = 1$, $p > 0.1$

We add the effects of group boundaries in model 3 (across). Model 3 shows significantly negative clustering parameters ($\theta_{ACROSS,1}$ and $\theta_{ACROSS,2}$) indicating, as expected, a strong tendency for both design interfaces and team interactions to be clustered within boundaries. The third-order parameter, $\theta_{ACROSS,12}$, is significantly positive, which indicates, contrary to hypothesis H1, that the tendency for design interfaces and team interactions to be aligned is stronger across boundaries. Note that model 3 still shows a negative overall effect due to group boundaries, that is, model 3 predicts that the overall probability for design interfaces and team interactions to be aligned is lower across boundaries than within boundaries. However, this is due to strong clustering effects rather than to the pure alignment effect. We discuss this apparently clashing result in the next section.

Model 4 (modular) includes the effects of system modularity. The model suggests that there is a significant concentration of team interactions between modular systems ($\theta_{MODULAR,2} > 0$), yet modularity does not directly influence the alignment of design interfaces and team interactions as indicated by the insignificant $\theta_{MODULAR,12}$ parameter. Model 4 also includes a significantly negative $\theta_{ACROSS,MODULAR,12}$ parameter, which indicates that when considering the cases across boundaries, the pure propensity of design interfaces and team interactions to be aligned is significantly lower between modular systems (in line with H2).

Model 5 (indirect) shows not statistically significant parameters associated with indirect effects. However, model 6 (indirect within) shows that indirect effects are statistically significant within boundaries. More specifically, the significantly positive $\tau_{WITHIN,221}$ parameter indicates that there is a

strong propensity for indirect team interactions configurations to be present within boundaries (in line with H5.) Contrary to H6, the significantly negative $\tau_{WITHIN,112}$ parameter indicates that indirect design interface configurations are not likely to occur within boundaries.

Effects of design interface strength

In Table 6, model 1 (exchange) includes basic dyadic and triadic parameters in addition to significantly positive exchange parameters ($\rho_{WEAK,2}$ and $\rho_{STRONG,2}$). Model 2 (clustering) includes clustering parameters resulting in significantly negative parameters, confirming that both team interactions and design interfaces (at both levels) tend to be clustered within boundaries. Model 3 (association) includes statistically significant association parameters. As hypothesized by H3, we found that strong design interfaces are more likely to be aligned with team interactions than are weak design interfaces (i.e., $\theta_{STRONG,2}$ is significantly greater than $\theta_{WEAK,2}$.) Model 4 includes third-order parameters to capture whether there is a significant difference of the association effect across boundaries. We found that $\theta_{ACROSS,WEAK,2}$ and $\theta_{ACROSS,STRONG,2}$ are not significantly different, indicating that design interface strength does not moderate the alignment of design interfaces and team interactions across boundaries (H4 is not supported).

7. Discussion of Results

In order to rigorously test our hypotheses, we built several *logit p** models of the alignment matrix. One of the key advantages of the *p** formulation is that it explicitly controls for dyadic and triadic tendencies embedded in network data. In our problem, controlling for *clustering* effects was particularly important to disentangle such an effect from the *pure alignment* effect we wanted to investigate. It is important to note that the results of the *logit p** analysis, detailed in the previous section, largely coincide with our *log-linear* p_1 analysis (available from the authors upon request).

As expected, we found a strong tendency for design interfaces and team interactions to be aligned throughout the network. Even this most basic result is tremendously relevant. It suggests that managers should be able to explicitly examine the product architecture in order to plan for cross-team interactions when organizing design teams. However, our results also suggest that managers need to be wary of various factors which may prevent a perfect alignment of design interfaces and design team interactions.

First, when including the effects of group boundaries in our statistical models, we found that, as expected, clustering effects (i.e., concentration of ties within boundaries) are very strong, both in the product and organizational domains. Surprisingly, group boundaries do not hinder the alignment of

design interfaces and team interactions (as hypothesized in H1) but significantly strengthen such an alignment. Does this mean that we should observe a significantly larger proportion of aligned cases across boundaries? No, it does not. Indeed, both p_1 and p^* models predict a significantly lower probability of encountering aligned cases across boundaries. The reason for this apparently clashing result is that the effects of organizational/system boundaries have two components, a *clustering* component and a *pure alignment* component which, when considered jointly, result in smaller probabilities of finding aligned ties across boundaries (Robin *et al.* 1999). However, when disentangling both clustering and pure alignment effects, we observe that the pure alignment tendency is significantly stronger across boundaries. In order words, due to clustering effects, the expected level of aligned cases across boundaries is significantly lower than within boundaries. That is, in our study, having 52% of the cross-boundary design interfaces unmatched by team interactions and 25% of the cross-boundary team interactions unmatched by design interfaces, is actually better than what we should expect, considering such strong clustering effects. Hence, the main hindering effect of organizational/system boundaries is to lower the expected level of alignment across boundaries. As a result, although team interactions and design interfaces exhibit a strong tendency to be aligned across boundaries, managers should still expect (and prepare for) a significantly lower proportion of aligned cases across boundaries. This result also suggests that we need to understand better which factors moderate the alignment of design interfaces and team interactions across (and within) boundaries, which is precisely what the rest of our results focus on.

When studying the effects of system modularity, we first found no direct effect on the alignment of design interfaces and team interactions. However, and more interestingly, when considering the cases across boundaries, we found that the alignment of design interfaces and team interactions is weaker between modular systems than with integrative systems. Hence, the teams that do particularly well at addressing cross-boundary design interfaces are the ones that design integrative systems. Similarly, system architects seem to do a better job identifying design interfaces with integrative systems. This suggests an important consideration for managers planning development projects that involve modular systems. Although Sanchez and Mahoney (1996) suggest that system modularization and component standardization facilitate coordination of development activities, our results indicate that, during the design of complex products (which typically are not perfectly modular), the effects of modularization may hinder managers' handling of interdependence across group boundaries.

While analyzing the effects of design interface strength on the association of design interfaces and team interactions, we found that the stronger the design interface, the greater the likelihood that teams would interact (H3), which is in line with previous research about team interdependence. Even though

this result could be considered as "good news" for managers who might believe design interface strength drives the alignment of design interfaces and team interactions across organizational/system boundaries (H4), we did not find empirical support for this latter hypothesis. Indeed, this latter result can be interpreted as "bad news" for managers because even if design interfaces across boundaries are critical, the likelihood that they are unmatched by the corresponding design teams is the same as if they were non-critical design interfaces. Therefore, managers need to pay particular attention to identifying and managing critical cross-boundary design interfaces without relying on the level of criticality of the interface as a mechanism to improve the alignment of design interfaces and team interactions across organizational or system boundaries.

When studying the effects of indirect team interactions (H5) and indirect design interfaces (H6) on the overall network, we found no statistical evidence. However, when testing for such effects within boundaries, we found a significant tendency for indirect team interactions to occur. This result suggests that design teams may indeed use other trustful intermediary teams (which are most likely to be within their groups) to obtain relevant technical information. As suggested by recent research in knowledge management (e.g., Hansen 2002), indirect team interaction is an effective mechanism for acquiring knowledge when the intermediary unit does not significantly distort the information exchanged, which, according to our results, is likely to occur within groups. The benefits of indirect interactions rest on the underlying hypothesis that indirect interactions free up teams to focus on "more expensive to maintain" direct interactions which are truly necessary for complex knowledge transfer (Hansen 1999, 2002). Hence, managers should foster trust and rich knowledge sharing between design teams so that more effective indirect interactions take place and teams make better use of "more expensive" direct interactions to address critical interdependences. Based on our results, doing so could be particularly influential across boundaries where the use of indirect team interaction as a coordination mechanism appears to be scarce.

As for the effects of indirect design interfaces (H6), our results indicate that it is very unlikely that unmatched team interactions address indirect design interfaces. Since this result does not support the hypothetical existence of system-level design dependencies due to propagation of known direct design interfaces, the notion that unmatched team interactions echo the existence of unidentified or tacit (direct) design interfaces is not challenged. Hence, managers should pay particular attention to understanding the reasons for unmatched team interactions to update their architectural knowledge.

In summary, although the alignment of interfaces and interactions appears to be strong across boundaries, such "good news" needs to be interpreted cautiously. First, although the "pure" alignment is stronger across boundaries, clustering effects overcome such a tendency, resulting in a significantly

lower proportion of aligned cases across boundaries. Second, there is a significant difference across boundaries between modular systems and integrative systems. The alignment across boundaries is actually significantly weaker between modular systems. Third, although design criticality significantly influences the alignment of interfaces and interactions in general, such an effect is not particularly relevant across boundaries. Hence, strong cross-boundary design interfaces are equally likely to be unmatched by team interactions as weak design interfaces. Fourth, the apparently weak pure tendency for alignment within boundaries is compensated by indirect team interactions with other teams within boundaries, which is a coordination mechanism that does not appear to occur across boundaries.

7.1. Threats to Validity

Although we have rigorously studied the significance of some effects to explain the mismatches between design interfaces and team interactions, this paper has not included all contributing factors. Of particular concern would be omitted factors that correlate with one or more of the independent variables included in our statistical models. We briefly discuss the possible impact of two omitted effects: team interactions with system integration teams and design interface carry-over.

We expect the effects of interactions with system integration teams (i.e., the last six teams in the team interaction matrix) to be insignificant because these teams interact with almost every other team in the organization. Moreover, we found no significant empirical evidence of indirect team interactions through system integration teams (Sosa 2000). This is consistent with our results which suggest that design teams tend to use indirect interactions to exchange technical information only within group boundaries.

In cases of incremental innovation such as the development of derivative products, many design interfaces may not change from one generation of a product to the next. This “carry-over effect” could result in unaddressed design interfaces as long as the current organization “remembers” the state of those unchanged interfaces and needs no interaction to verify them. Even though we believe that this is an important factor to consider when planning team interactions, we found, during follow-up data collection with the high-pressure turbine and low-pressure turbine design teams, that some unchanged design interfaces were still addressed by team interactions due to the high level of criticality of the interface and/or the presence of new design participants. (Sosa 2000.) Furthermore, to rigorously study the design interface carry-over effect, we need to consider the evolution and impact of a change in the design interface as well as knowledge dynamics of the design teams involved. Given the various factors involved in the carry-over effect, we do not expect it to significantly impact the hypothesized effects studied in this paper.

8. Conclusions and Implications

Previous research has studied product architecture decisions and technical communication patterns in product development from separate viewpoints. Rather, this paper takes an integrated and structured approach to investigate the alignment (and misalignment) of product and organizational structures during the detailed design phase of complex development efforts. Our approach involves four important steps: 1) capturing the product architecture by documenting product decomposition and design interfaces, 2) capturing the integration effort of the development organization by documenting technical interactions between design teams, 3) mapping the product architecture onto the development organization by merging design interfaces and team interactions into the alignment matrix, and 4) analyzing the patterns exhibited in the alignment matrix to uncover the factors that systematically prevent design interfaces and team interactions from being aligned.

The analysis presented in this paper has focused on explaining mismatches between design interfaces and team interactions. This work contributes to the product innovation literature, in both product architecture and organizational perspectives, by uncovering factors that impact the likelihood that 1) product-related interdependences are not addressed by team interactions, and 2) design teams interact despite of absence of a product-related interdependence between them. Our results not only show how organizational/system boundaries impact the likelihood of misalignment of interfaces and interactions but also how boundary effects may be contingent upon system modularity, strength of the design interface, and indirect interactions. These results are particularly important when managing knowledge in organizations, as highlighted in a recent review by Argote *et al.* (2003, p. 578) who concluded that "where organizational boundaries are drawn has important implications for knowledge transfer and subsequent organizational performance… More research is needed on the conditions under which organizational members value internal versus external knowledge. " Our findings begin to shed some light on this new research area.

From an analysis viewpoint, we illustrate how to formally build statistical models, based on social networks methods, for rigorous hypothesis testing using DSM-type data. We introduce a novel and powerful statistical technique (the *logit p*,* developed by Wasserman and Pattison) to estimate statistical models that control for dyadic and triadic network effects. The important advantage of these models over traditional statistical analysis is that they control for embedded effects typical of network data. As shown, only by conducting this type of analysis were we able to disentangle the clustering effects from the pure alignment effect across boundaries (which is something that we could not have done with a descriptive or QAP analysis). We also illustrated how the use of *p** models opens up new

avenues for researchers interested in testing the effects of network structures that involve three players. Although the *p** formulation is very robust from a statistical modeling viewpoint, fitting these models to data is still done approximately. As mentioned, we tested the robustness of our results by completing similar analysis using a *log-linear* p_1 approach. Yet, future research will benefit from ongoing efforts focusing on alternative fitting strategies of *p** models.

As in many other empirical studies that collected data in a single organization (e.g., Morelli *et al.* 1995, Van den Bulte and Moanaert 1998, Terwiesch *et al.* 2002), we cannot claim the generality of our findings before completing similar studies in other types of products in different industries. Yet, we would expect to obtain analogous results in other projects developing complex systems and where the design teams are organized according to the product architecture as is the case in many development projects in the automobile and aircraft industries (e.g., McCord and Eppinger 1993, Robertson and Allen 1992). Certainly, this study is descriptive in nature and as such, we avoid drawing explicit prescriptive conclusions from it. Moreover, one could presume that a significant proportion of misalignment observed is simply due to abnormalities of the project studied. However, considering the critical nature of the product under development and the proven record of the organization involved, we strongly doubt that this project exhibited more inefficiencies than a typical development project of this nature.

8.1. Managerial Implications

In addition to the managerial repercussions of each result, as discussed in the previous section, this research has important implications for managers, from three different perspectives. From a strategic viewpoint, Henderson and Clark (1990) highlighted the fact that "learning about changes in the architecture of the product is unlikely to occur naturally. Learning about changes in architecture –about new interactions across components (and often across functional boundaries) – may therefore require explicit management and attention." (p. 28). By documenting the architecture of the product in a design interface matrix for every generation of a product family, managers can quickly identify novel architectures. More importantly, by building the alignment matrix, managers have a compact and visual representation that allows them to diagnose how their organization addresses the design interfaces of the product under development. This provides a systematic way to evaluate how development organizations manage system-level (architectural) knowledge, a critical issue for firms facing architectural innovation. Furthermore, an alignment matrix would help managers to pinpoint where to focus their efforts to realign their organization to develop distinct product architectures.

From a project planning perspective, our approach helps managers plan better technical interactions between design teams across organizational and functional boundaries. This is particularly beneficial in projects of incremental and modular innovation, in which the architecture of the product is well understood. Our analysis suggests that managers should focus their efforts on understanding the causes of *unmatched* design interfaces (the "W"s or "S"s of the alignment matrix) and *unmatched* team interactions (the "O"s of the alignment matrix) across modular systems. These are the design interfaces most difficult to identify, and (if identified), most likely to be unaddressed by the corresponding design teams, even if they are critical design interfaces. For example, in the organization where we applied our approach, some of the 25 unmatched team interactions between modular systems were critical design interfaces that had not been previously identified by design experts. As a result of applying our method, managers learned about these interdependencies and established a dedicated design team to explicitly handle these critical cross-boundary design interfaces during the development project of the next-generation engine.

From a project execution perspective, our approach can also be used to actively monitor complex development projects. Nowadays, development organizations are capturing technical interdependences between their development tasks in order to plan for design iterations (Smith and Eppinger, 1997). One way to monitor whether all relevant interdependences have been captured and whether communication patterns follow the original planning is by comparing the task structure of the project with the actual communication patterns that occur during the project. By uncovering any misalignment between project structure and actual communication while it happens, managers can identify the source of such a misalignment and act upon it.

8.2. Research Implications

This paper opens up a new stream of research on the interface of product architecture and organizational structure by studying whether and how they fit together. In this paper, we have studied the static effects that influence the (mis)alignment of design interfaces and team interactions. An interesting challenge (methodological and statistical) for future research is to extend this method to explore the evolution over time of such alignment for several generations in a product family.

In projects with a significantly large proportion of outsourced components, our approach could be extended to study the alignment of product architecture and the network of suppliers to investigate the effects of factors such as enterprise boundaries, contracts, information technology compatibility, and organizational culture disparity. Doing so would result in a novel study that would not only consider the impact of OEM-supplier relations but also cross-supplier relations on product innovation.

Although understanding the factors that affect the "fit" of product and organizational structures is important, future research is also needed to understand the circumstances in which "fit" influences performance. Extending this method to study various mappings of product architectures and development organizations would be a challenge (and an opportunity) for future research efforts. This study is based on the assumption of a direct mapping of product architecture and development organization. What if this were not the case? What types of barriers are more severe (organizational or system barriers)? Under what circumstances is an organizational design that mirrors the architecture of the product a good one?

This paper combines the innovation literature with social network literature for statistical analysis purposes. However, in the organizational domain, there exists literature that relates social network properties such as cohesion and structural holes to organizational performance (e.g., Burt 1992). Important contributions to the product innovation literature can be offered by defining product network properties and studying their relation to both product and process performance.

9. Acknowledgements

Funding of this research was provided by the MIT Center for Innovation in Product Development. We appreciate the assistance of the engineers and managers at Pratt & Whitney Aircraft. We appreciate insightful comments by Christophe Van den Bulte, Dawn Iacobucci, William Lovejoy, Christoph Loch, Luk Van Wassenhove, the associate editor, and three anonymous reviewers. We also thank Pip Pattison for insights in data analysis issues, and François-Xavier Priour and Jean Cropper for editorial assistance.

10. References

Adler, P.S. 1995. Interdepartmental interdependence and coordination: the case of the design/manufacturing interface. *Org. Sci.* 6 147-167.

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

Allen, T.J. 1977. *Managing the Flow of Technology*. MIT Press, Cambridge, MA.

Anderson, C., S. Wasserman, B. Crouch. 1999. A *p** primer: Logit models for social networks. *Social Networks. 21,* 37-6.

Argote, L., B. McEvily, and R. Reagans. 2003. Managing knowledge in organizations: an integrative framework and review of emerging trends. *Manag. Sci.* 49 (4), 571-582.

Baldwin, C. Y., K. B. Clark. 2000. *Design rules: Volume1: The power of modularity*. Cambridge, MA: MIT Press.

Brown, S. L., K. M. Eisenhardt. 1995. Product development: Past research, present findings, and future directions. *Academy of Manag. Review*. 20, 343-378.

Burt, R. 1992. *Structural Holes. The Social Structure of Competition*. Harvard University Press, Cambridge, MA.

Eisenhardt, K., B. Tabrizi. 1995. Accelerating adaptive processes: Product innovation in the global computer industry. *Admin. Sci. Quart*. 40(1) 84-110.

Eppinger, S. D., D.E. Whitney, R.P. Smith, D.A. Gebala. 1994. A model-based method for organizing tasks in product development. *Res. in Eng'g. Des.* 6(1) 1-13.

Fienberg, S.E., M.M. Meyer, S. Wasserman. 1985. Statistical analysis of multiple sociometric relations. *J. Amer. Statist. Assoc*. 80 51-67.

—, S. Wasserman. 1981. Categorical data analysis of single sociometric relations. S. Leinhardt, ed. *Sociological Methodology* 1981. Jossey-Bass, San Francisco, CA. 156-192.

Frank, O., and D. Strauss. 1986. Markov graphs. *J. Amer. Statist. Assoc*. 81, 832-842.

Galbraith, J. R. 1973. *Designing Complex Organizations*, Reading: Adison-Wesley Publishing.

Gatignon, H., M.L. Tushman, W. Smith, P. Anderson. 2002. A structural approach to assessing innovation: Construct development of innovation locus, type, and characteristics. *Manag. Sci*. 48(9) 1103-1122.

Gonzalez-Zugasti, J., K.Otto, J. Baker. 2000. A method for architecting product platforms. *Res.in Eng'g. Des.* 12, 61-72.

Granovetter, M. 1973. The strength of weak ties. *Amer. J. Soc.* 6 (6) 1360-1380.

Griffin, A., J.R. Hauser.1992. Patterns of communication among marketing, engineering and manufacturing: A comparison between two new product teams. *Manag. Sci*. 38,3, 360-373.

Gupta, A. K., Govindarajan. 2000. Knowledge flows within multinational corporations. *Strat Manag. J*. 21(4) 473-496.

Gupta, S., V. Krishnan. 1999. Integrated component and supplier selection for a product family. *Productions and Operations Manag.* 10, 291-308.

Hansen, M. 1999. The search-transfer problem: The role of weak ties in sharing knowledge across organization subunits. *Admin. Sci. Quart.* 44(1) 82-111.

—. 2002. Knowledge networks: Explaining effective knowledge sharing in multiunit companies. *Org. Sci.* 13(3) 232-248.

Henderson, R., K. Clark. 1990. Architectural innovation: the reconfiguration of existing product technologies and the failure of established firms. *Admin. Sci. Quart*. 35(1) 9-30.

Hirtz J., R. B. Stone, D. A. McAdams, S. Szykman, K. L. Wood. 2002. A functional basis for engineering design: Reconciling and evolving previous efforts. *Res Eng Design.* 13: 65-82.

Holland, P.W., S. Leinhardt. 1981. An exponential family of probability distributions for directed graphs. *J. Amer. Statist. Assoc*. 76 33-51.

Kirschman, C. F. and G. M. Fadel. 1998. Classifying functions for mechanical design. *J. of Mech. Design*. 120 475-482.

Krackhardt, D. 1988. Predicting with networks: nonparametric multiple regression analysis of dyadic data. *Social Networks*. 10, 359-381.

Krishnan, V., R. Sinngh, and D. Tirupati. 1999. A model-based approach for planning and developing a family of technology-based products. *Manufacturing & Service Operations Manag.* 1(2) 132-156.

—, and K. Ulrich. 2001. Product development decisions: a review of the literature. *Manag. Sci*. 47 (1) 1-21.

Loch, C., C. Terwiesch, S. Thomke. 2001. Parallel and sequential testing of design alternatives. *Manag. Sci.*, 47(6)663-678.

Marsden, P. V. 1990. Network data and measurement. *Annual Review of Sociology*. 16, 435-463.

—, and K. E. Campbell. 1984. Measuring tie strength. *Social Forces*. 63: 482-501.

McCord, K.R., S.D. Eppinger. 1993. Managing the integration problem in concurrent engineering. *Sloan Working Paper # 3594*. MIT Sloan School of Management. Cambridge, MA.

Mihm, J., C. Loch, and A. Huchzermeier. 2003. Problem-solving oscillations in complex engineering projects. *Manag. Sci*. 46 (6) 733-750.

Morelli, M.D., S.D. Eppinger, R.K. Gulati. 1995. Predicting technical communication in product development organizations. *IEEE Trans. Eng'g. Manag.* 42(3) 215-222.

Novak, S., S. Eppinger. 2001. Sourcing by design: product complexity and the supply chain. *Manag. Sci*. 47(1) 189-204.

Pattison, P., and Wasserman, S. 1999. Logit models and logistic regressions for social networks: II. Multivariate relations. *British J. of Math. and Stat. Psychology. 52*, 169-193.

Pimmler, T.U., S.D. Eppinger. 1994. Integration analysis of product decompositions. *ASME Conference on Design Theory and Methodology*. Minneapolis, MN. 68, 343-351.

Robertson, D. C. and T. J. Allen. 1992. Managing CAD systems in mechanical design engineering. . *IEEE Trans. Eng'g. Manag.* 39(1) 22-31.

Robertson, D. and Ulrich, K. 1998. Planning for Product Platforms. *Sloan Manag. Review,* 39(4) 19-31.

Robins, G., Pattison, P., and Wasserman, S. 1999. Logit models and logistic regressions for social networks, III. Valued relations. *Psychometrika. 64,* 371-394.

Rowles, C.M. 1999. *System Integration Analysis of a Large Commercial Aircraft Engine*. Master's Thesis. MIT.

Sanchez, R., J. T. Mahoney. 1996. Modularity, flexibility, and knowledge management in product and organization design. *Strategic Manag. J.* 17 (Winter special issue), 63-76.

Simpson, T., J. Maier, F. Mistree. 2001.Product platform design: method and application *Res Eng Design. 13: 2-22.*

Smith, R., S. Eppinger. 1997. Identifying controlling features of engineering design iteration. *Manag. Sci.* 43(3) 276-293.

Sosa, M.E. 2000. *Analyzing the Effects of Product Architecture on Technical Communication in Product Development Organizations*. Doctoral Thesis. MIT. Cambridge, MA.

—, S.D. Eppinger, C.M. Rowles. 2003. Identifying modular and integrative systems and their impact in design team interactions. *Journal of Mech. Design,* 125(2), 240-252.

—, —, M. Pich, D. McKendrick, S. Stout. 2002. Factors that influence technical communication in distributed product development: an empirical study in the telecommunications industry. *IEEE Trans. Eng'g. Manag*. 49(1) 45-58.

Steward, D. 1981. The design structure matrix: a method for managing the design of complex systems. *IEEE Trans. Eng'g. Management* EM-28(3) 71-74.

Strauss, D. and M. Ikeda. 1990. Pseudolikelihood estimation for social networks. *J. Amer. Statist. Assoc*. 85, 204-212.

Terwiesch, C., C. H. Loch, and A. De Meyer. 2002. Exchanging preliminary information in concurrent engineering: Alternative coordination strategies. *Org. Sci*. 13 (4) 402-419.

Thompson, J. D. 1967. *Organizations in Action*, New York: McGraw Hill.

Thomke, S. H. 1998. Managing experimentation in the design of new products. *Manag. Sci.* 44(6) 743-762.

Tsai, W. 2002. Social structure of "coopetition" within a multiunit organization: Coordination, competition, and intraorganizational knowledge sharing. *Org. Sci*. 13(2) 179-190.

Tushman, M. L. 1977. Special boundary roles in the innovation process. *Admin. Sci. Quarterly*. 22, 587-605.

Ulrich, K.T. 1995. The role of product architecture in the manufacturing firm. *Res. Policy*. 24 419-440.

—, D. J. Ellison. 1999. Holistic customer requirements and the design-select decision. *Manag. Sci.,* 45() 641-658.

—, S.D. Eppinger. 2000. *Product Design and Development*. 2nd. Edition. McGraw Hill, Inc.

Van den Bulte, C., R.K. Moenaert. 1998. The effects of R&D team co-location on communication patterns among R&D, marketing, and manufacturing. *Manag. Sci.* 44: S1-S19.

Wasserman, S. and D. Iacobucci. 1986. Statistical analysis of discrete relational data. *British J. of Math. Stat. Psychology*. 39, 41-64.

—, D. Iacobucci. 1988. Sequential social network data. *Psychometrika* 53 261-282.

—, P. Pattison. 1996. Logit models and logistic regression for social networks: I. an introduction to Markov graphs and p*. *Psychometrika* 61 401-425.

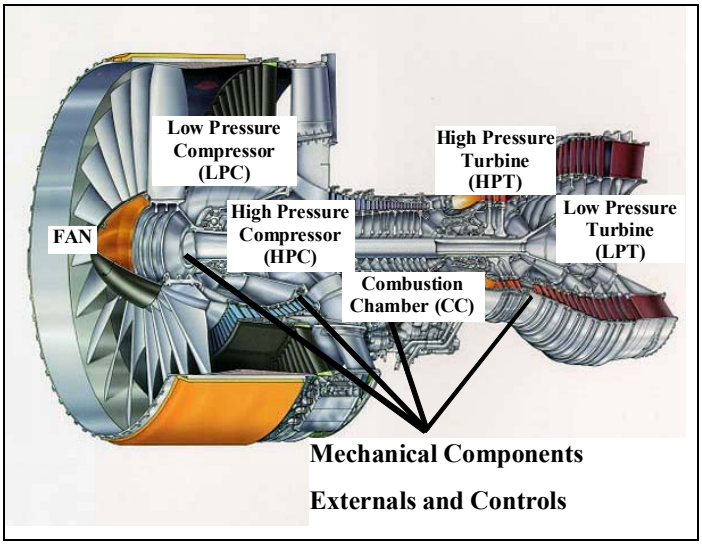


Figure 4. PW4098 Commercial Aircraft Engine Studied

		Modular Systems						Integrative Systems	
		FAN system	LPC system	HPC system	CC system	HPT system	LPT system	Mech. Components	External and Controls
Modular Systems	**FAN system (7 components)**								
	LPC system (7 components)								
	HPC system (7 components)								
	CC system (5 components)								
	HPT system (5 components)								
	LPT system (6 components)								
Integrative Systems	**Mech. Components (7 components)**								
	Externals and Controls (10 components)								

W= WEAK design interface S=STRONG design interface

Figure 5. Design Interface Matrix

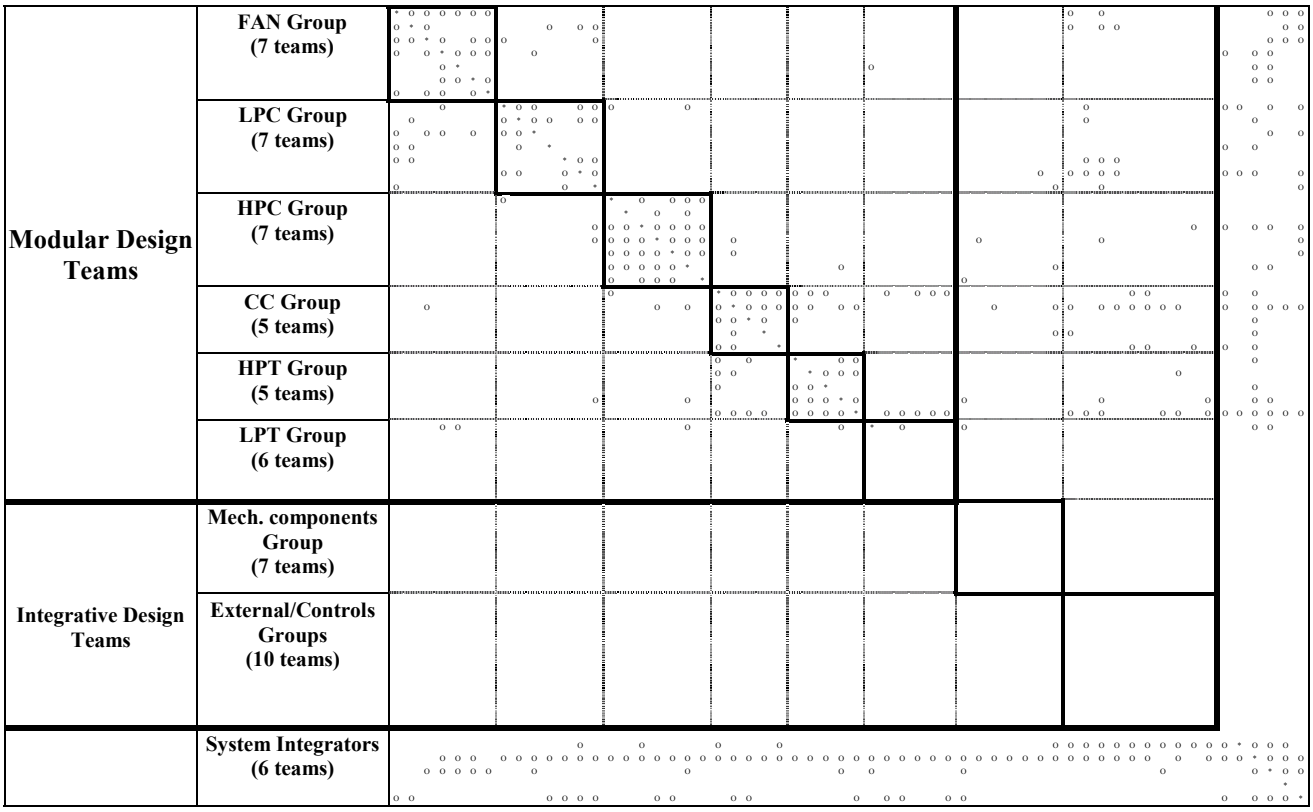


Figure 6. Team Interaction Matrix (Binary)

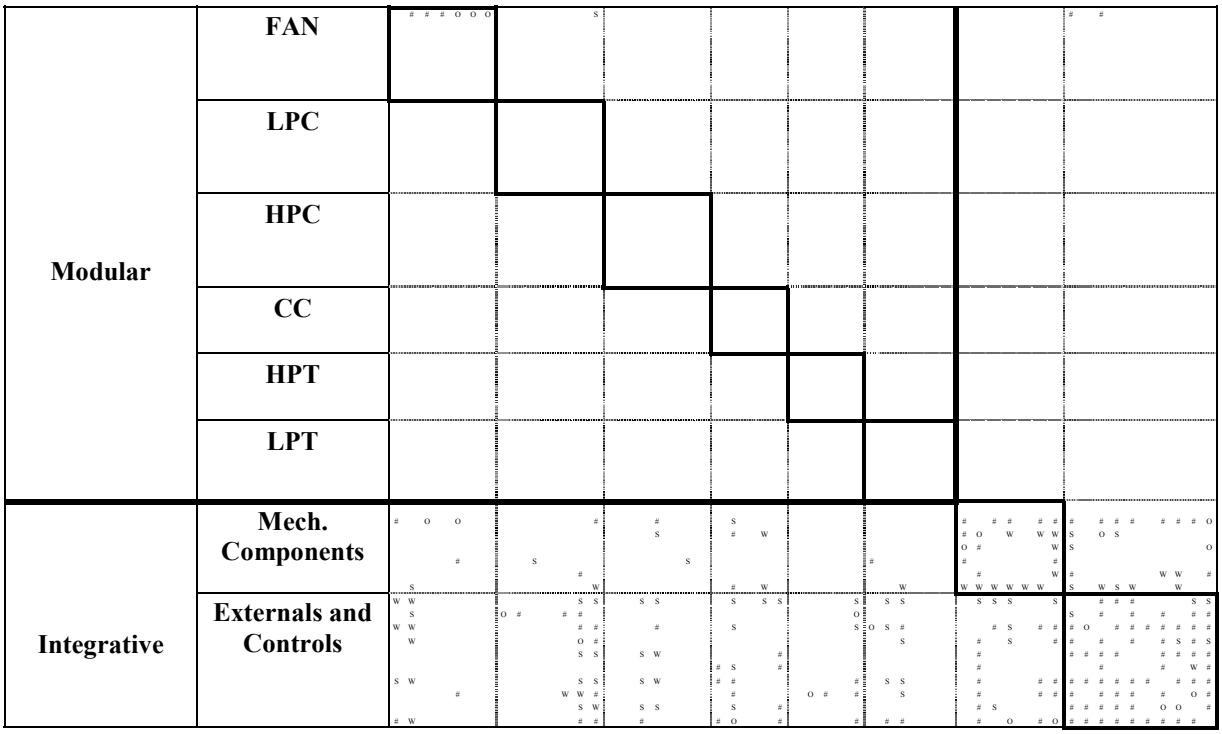


Figure 7. Alignment Matrix

W: Unmatched WEAK design interface S: **Unmatched** STRONG design interface **O: Unmatched** Team interaction
#: Aligned presence of design interface and team interaction **(Blank): Aligned** absence of design interface and team interaction
***:** Diagonal elements (meaningless)

(Optional) Appendix A. *Log-linear p_1* statistical analysis

The statistical models described in this appendix are based on the p_1 distribution introduced by Holland and Leinhardt (1981). In order to introduce the p_1 distribution, let us consider the four-dimensional **Y**-array whose component Y_{ijkl} describes the interaction between element *i* and element *j*. The third and fourth dimensions of the **Y**-array are binary and describe the state of dyad *ij* of a network. Hence, k=1 if element *i* interacts with element *j*, and l=1 if element *j* interacts with element *i*. Fienberg and Wasserman (1981) show that Holland and Leinhardt's distribution, p_1, can be expressed in a log-linear format, as follows:

$$\ln \mathrm{P}\{Y_{ijkl}=1\} = \lambda_{ij} + (k+l)\theta + k\cdot\alpha_i + l\cdot\beta_i + l\cdot\alpha_j + k\cdot\beta_j + (kl)\rho \qquad \text{(A1)}$$

The $\{\alpha_i\}$ parameters measure the *expansiveness* of the elements of the network, indicating how likely an element is to generate relational ties (non-zero cells in row *i* of the matrices). The $\{\beta_j\}$ parameters measure the *attraction* of the elements of the network, indicating how likely an element is to receive relational ties (non-zero cells in column *j* of the matrices). The "reciprocity" parameter, ρ, measures the overall tendency in the network to reciprocate interactions. The θ parameter indicates the overall volume of interaction in the network. Finally, the λ_{ij} parameters are "dyadic" effects that ensure that the probabilities sum to one for each dyad (equation A1); they have no substantive meaning. For a more detailed description of these parameters, refer to Holland and Leinhardt (1981).

Similar to our *logit p** approach, we build our log linear models in two stages. First, we model our alignment matrix considering binary design interfaces to test H1 and H2. Then, we extend our log-linear models to consider trichotomous design interfaces in order to test H3 and H4.

1. Log-linear p_1 models of alignment matrix with binary design interfaces

Similar to Van den Bulte and Moenaert (1998) we build log-linear models of our alignment matrix (with binary design interfaces) in five steps:

1.1. Build a p_1 model of the alignment matrix

Fienberg *et al* (1985) and Wasserman and Iacobucci (1988) extend p_1 to multiple sociometric relations. Based upon these results we develop a log-linear model of the alignment matrix. We consider the joint distribution of both design interfaces and team interactions for a given dyad. That is, each dyad (*i,j*) of the alignment matrix has 16 states. Four (2 x 2) states are associated with the dyad's design interface relation, and four (2 x 2) states are associated with its team interaction relation, resulting in 16 states. Following the definitions of the variables k and l introduced before, we assign the subscripts (k_1,l_1) to describe the four states associated with the design interface relation of dyad (*i,j*), while the subscripts (k_2,l_2) refer to the four states associated with the team interaction relation of dyad (*i,j*). Hence, the redefined **Y**-array has now six dimensions 54 x 54 x (2 x 2) x (2 x 2), and its characteristic element can be defined as follows:

$Y_{ij\ k1,l1\ k2,l2}$ = 1 if dyad (*i,j*) behaves as described by (k_1,l_1) for their design interfaces AND by (k_2,l_2) for their team interactions.
$Y_{ij\ k1,l1\ k2,l2}$ = 0 otherwise.

Considering the joint distribution of design interfaces and team interactions yields a log-linear model which describes simultaneously the behavior of the elements of our network according to two

independent relations (design interfaces and team interactions). Hence, the first base log-linear model can be written as follows:

$$\ln P\{Y_{ij\,k_1 l_1\,k_2 l_2}=1\}=\lambda_{ij}+(k_1+l_1)\theta_1+k_1\alpha_{1i}+l_1\beta_{1i}+l_1\alpha_{1j}+k_1\beta_{1j}+(k_1 l_1)\rho_1+ (k_2+l_2)\theta_2+k_2\alpha_{2i}+l_2\beta_{2i}+l_2\alpha_{2j}+k_2\beta_{2j}+(k_2 l_2)\rho_2 \quad \text{(A2)}$$

The parameters of this model have the same meaning as in the original p_1 model, but applied to either design interfaces (subscript 1) or team interactions (subscript 2).

1.2. Aggregate physical components and design teams into groups

Fienberg and Wasserman (1981) introduced the approach of placing actors into subsets using relevant actor characteristics such that actors within a subset are assumed to behave similarly. Based on this approach we aggregate the 54 elements of the **Y**-array into 8 subsets according to the system boundaries of the product and the organizational boundaries of the development organization, respectively. By doing so, we obtain a much smaller **W**-array whose dimensions are 8 x 8 x (2 x 2) x (2 x 2), with elements $\{W_{rs\,k1,l1\,k2,l2}\}$ to be equal to the number of dyads between groups r (G_r) and s (G_s) whose design interfaces are described by (k_1,l_1) and whose team interactions are described by (k_2,l_2). Hence,

$$w_{rs\,k_1 l_1\,k_2 l_2}=\sum_{i\in G_r}\sum_{j\in G_s} y_{ij\,k_1 l_1\,k_2 l_2} \quad \text{(A3)}$$

Therefore, we can rewrite the model in equation (2) to specify the expected number of dyads between groups r and s that behave as (k_1,l_1, k_2,l_2) as follows:

$$\ln E(W_{rs\,k_1 l_1\,k_2 l_2})=\lambda_{rs}+(k_1+l_1)\theta_1+k_1\alpha_{1r}+l_1\beta_{1r}+l_1\alpha_{1s}+k_1\beta_{1s}+(k_1 l_1)\rho_1+ (k_2+l_2)\theta_2+k_2\alpha_{2r}+l_2\beta_{2r}+l_2\alpha_{2s}+k_2\beta_{2s}+(k_2 l_2)\rho_2 \quad \text{(A4)}$$

It is important to note that even though we have grouped components and teams into groups to facilitate the estimation and statistical inference of the models, the unit of analysis is still the dyad.

1.3. Capture the alignment of design interfaces and team interactions

The base model specified in equation (A4) assumes that design interfaces and team interactions are two independent relations of the same network of elements. However, we need to consider second-order interaction effects between design interfaces and team interactions to capture the association between the design interface matrix and the team interaction matrix. Following the notation of Wasserman and Iacobucci (1988), we define the following two association parameters, $\theta_{1,2}$ and ρ_{12}. $\theta_{1,2}$ measures any tendency toward conformity across relationships. That is, component i depends on component j, AND team i reports interaction with team j (the "#" cells of the alignment matrix). On the other hand, ρ_{12} measures tendency toward flow reversal. That is, component i depends upon component j, AND team j reports interaction with team i. These two parameters correspond to *association* and *exchange* parameters included in our *logit p** formulation. By including these two parameters, we extend our base model (equation A4) as follows:

$$\ln E\,(W_{rs\,k_1 l_1\,k_2 l_2}) = \lambda_{rs} + (k_1 + l_1)\theta_1 + k_1\alpha_{1r} + l_1\beta_{1r} + l_1\alpha_{1s} + k_1\beta_{1s} + (k_1 l_1)\rho_1 + (k_2 + l_2)\theta_2 + k_2\alpha_{2r} + l_2\beta_{2r} + l_2\alpha_{2s} + k_2\beta_{2s} + (k_2 l_2)\rho_2 + (k_1 k_2 + l_1 l_2)\theta_{1,2} + (k_1 k_2 + l_1 l_2)\rho_{1,2} \quad \text{(A5)}$$

1.4. Extend the model with structural parameters: *ACROSS* and *MODULAR*

To explicitly represent organizational and system boundary effects, we define the following indicator variable:

ACROSS = 1 if elements (i.e. component and team) *i* and *j* are in the different groups ($r \neq s$)
ACROSS = 0 if $r=s$

By expanding the dimension of the **W**-array with *ACROSS* as the seventh dimension, we can estimate parameters associated with the second-order interaction terms *ACROSS* x k_1, and *ACROSS* x k_2, due to symmetry of the **W**-array identical to *ACROSS* x l_1 and *ACROSS* x l_2, respectively. These terms capture the within-boundary effects exhibited in both the design interface matrix and team interaction matrix. Indeed, we expect these terms to be significantly negative indicating that it is less likely to encounter design interfaces across system boundaries and team interactions across organizational boundaries.

In order to include the effects due to system modularity into the model we define another indicator variable, *MODULAR*, as follows:

MODULAR=1 if both components of a dyad belong to modular systems (r<7 and s<7)
MODULAR=0 if one of the components of a dyad belongs to integrative systems (r≥7 or s≥7)

Having defined the indicator variables *ACROSS* and *MODULAR*, we proceed to define third-order and fourth-order interaction effects that allow us to test the effects due to group boundaries (H1) and the moderating effects due to system modularity (H2).

We estimate the parameter associated with the third-order interaction effect *ACROSS* x k_1 x k_2 (due to symmetry of the **W**-array identical to *ACROSS* x l_1 x l_2). Thus, $\theta_{ACROSS,1,2}$ captures whether the occurrence of dyads across boundaries with design interfaces matched by team interactions is significantly less than the occurrence of those dyads within boundaries. Hence, a formal hypothesis testing for H1 can be specified as follows:

H1: $\theta_{ACROSS,1,2} < 0$

We also estimate the parameter associated with the fourth-order interaction effect *MODULAR* x *ACROSS* x k_1 x k_2 (due to symmetry of the **W**-array identical to *MODULAR* x *ACROSS* x l_1 x l_2). Thus, $\theta_{MODULAR,ACROSS,1,2}$ captures whether the effect due to organizational/system boundary is significantly different for modular systems than for integrative systems. We expect this effect to be significantly negative, which corresponds to fewer cross-boundary design interfaces (matched by team interactions) between modular systems than to integrative systems (H2). Hence, a formal hypothesis testing for H2 is expressed as follows:

H2: $\theta_{MODULAR,ACROSS,1,2} < 0$

After extending the model with indicator variables, *ACROSS* and *MODULAR*, to include the high-order interaction effects, we write our final log-linear model as follows:

$$\begin{aligned}
&\ln E\,(W_{rs\,k_1 l_1\,k_2 l_2}) = \lambda_{rs} + (k_1 + l_1)\theta_1 + k_1\alpha_{1r} + l_1\beta_{1r} + l_1\alpha_{1s} + k_1\beta_{1s} + (k_1 l_1)\rho_1 + \\
&(k_2 + l_2)\theta_2 + k_2\alpha_{2r} + l_2\beta_{2r} + l_2\alpha_{2s} + k_2\beta_{2s} + (k_2 l_2)\rho_2 + (k_1 k_2 + l_1 l_2)\theta_{1,2} + (k_1 k_2 + l_1 l_2)\rho_{1,2} + \\
&(ACROSS \cdot k_1 + ACROSS \cdot l_1)\theta_{ACROSS,1} + (ACROSS \cdot k_2 + ACROSS \cdot l_2)\theta_{ACROSS,2} + \\
&(ACROSS \cdot k_1 \cdot k_2 + ACROSS \cdot l_1 \cdot l_2)\theta_{ACROSS,1,2} + \\
&(MODULAR \cdot k_1 + MODULAR \cdot l_1)\theta_{MODULAR,1} + (MODULAR \cdot k_2 + MODULAR \cdot l_2)\theta_{MODULAR,2} + \\
&(MODULAR \cdot k_1 \cdot k_2 + MODULAR \cdot l_1 \cdot l_2)\theta_{MODULAR,1,2} \\
&(MODULAR \cdot ACROSS \cdot k_1 \cdot k_2 + MODULAR \cdot ACROSS \cdot l_1 \cdot l_2)\theta_{MODULAR,ACROSS,1,2}
\end{aligned} \tag{A6}$$

1.5. Fit models to data and test hypotheses H1 and H2

Fitting a model to data means finding the best (maximum likelihood) estimates of all parameters in the model that could produce the interaction data represented in the aggregated alignment matrix (**W**-array). To test the significance of the parameters we ask how much the expected and observed matrices differ. We do so by using conventional rules for likelihood-ratio and conditional likelihood-ratio tests for log-linear models for categorical data[1]. Based on results presented by Fienberg and Wasserman (1981) we use standard iterative proportional fitting computer programs for contingency tables (We used SPSS) to fit the models to data (**W**-array).

Table A1 shows the estimates of the parameters for five log-linear models with their respective likelihood-ratio statistics, G^2, and their numbers of degrees of freedom. The first model (independent) does not include the association parameters between design interfaces and team interactions. (This model corresponds to equation A4.) The second model (base, which corresponds to equation A5) includes $\theta_{1,2}$ which substantially improves the goodness-of-fit of the independent model (ΔG^2 = 943.71, Δdf =1) indicating, as expected, that there is significantly strong association of design interfaces and team interactions. Model 2 also includes the exchange parameter, ρ_{12}, which is not statistically significant. Including the second-order interaction effects with *ACROSS* greatly improves the goodness of fit of the base model (G^2 = 3719.18, df =5688, model not shown). The inclusion of these effects resulted in significantly negative parameters indicating, as expected, that significantly smaller portions of design interfaces and team interactions take place across boundaries.

Consistent with our *logit p** analysis, model 3 (across) includes a significantly positive $\theta_{ACROSS,1,2}$ parameter indicating that the "pure" alignment of design interfaces and team interactions are more likely to take place across boundaries. Yet, due to strong clustering effects model 3 still predicts a lower probability of finding aligned design interfaces and team interactions across boundaries.

When adding second-order and third-order interaction effects with *MODULAR,* the log-linear model does not significantly improve its goodness-of-fit (see Model 4), which indicates that system modularity does not have a direct effect on the alignment of design interfaces and team interactions. Finally, Model 5, corresponding to equation A6, includes the fourth-order interaction parameter $\theta_{MODULAR,ACROSS,1,2}$, whose value is significantly negative, indicating that cross-boundary design interfaces matched by team interactions are less likely to occur between modular systems (supporting H2).

[1] Bishop, Y. M., Fienberg, S. E., and Holland, P. W. (1975), *Discrete Multivariate Analysis: Theory and Practice*. Cambridge, MA: The MIT Press.

2. Log-linear p_1 models of the alignment matrix with valued design interfaces

In order to test the effects of design interface strength (H3 and H4) we need to consider non-binary design interfaces. By using the metric $STRENGTH_{ij}$, defined in section 5.1, a cell of the design interface matrix can have three possible states (i.e. NULL design interface, WEAK design interface, or STRONG design interface). Therefore, a dyad of the design interface matrix would have nine possible states and a dyad of the alignment matrix would have 36 possible states (nine states corresponding to the design interface dyad times four states corresponding to the binary team interaction dyad).

Wasserman and Iacobucci (1986) extended p_1 models for statistical analysis of discrete relational data. Using their notation, we can write a p_1 model that estimates the probability that dyad (i,j) of the design interface matrix will behave as the newly defined (k_1,l_1) as follows:

$$\ln P\,(Y_{ij\,k_1 l_1}) = \lambda_{ij} + \theta_{k_1} + \theta_{l_1} + \alpha_{i(k_1)} + \alpha_{j(l_1)} + \beta_{j(k_1)} + \beta_{i(l_1)} + \rho_{k_1,l_1} \qquad \text{(A7)}$$

The most important difference between this model and the one described by equation (A1) is that this model includes *expansiveness* $\{\alpha_{i(k1)}\}$, *popularity* $\{\beta_{j(k1)}\}$, and *reciprocity* $\{\rho_{k1,\,l1}\}$ parameters associated with each non-zero design interface strength (i.e. WEAK and STRONG).

Following the same rationale as in the previous sub-section, we extend the model described by equation (A7) to a model that describes the joint probability distribution of trichotomous design interfaces and binary team interactions including higher-order interaction effects at specific strengths. Hence, the equivalent model to equation (A5) using trichotomous design interfaces can be expressed as follows:

$$\begin{aligned}\ln E\,(W_{rs\,k_1 l_1 k_2 l_2}) = {} & \lambda_{rs} + \theta_{k_1} + \theta_{l_1} + \alpha_{r(k_1)} + \alpha_{s(l_1)} + \beta_{s(k_1)} + \beta_{r(l_1)} + \rho_{k_1,l_1} + \\ & \theta_{k_2} + \theta_{l_2} + \alpha_{r(k_2)} + \alpha_{s(l_2)} + \beta_{s(k_2)} + \beta_{r(l_2)} + \rho_{k_2,l_2} + \theta_{1,2} + \rho_{1,2}\end{aligned} \qquad \text{(A8)}$$

The $\{\theta_{1,2}\}$ parameters are associated with the second-order interaction effects k_1 x k_2 (which due to symmetry are identical to l_1 x l_2). Since k_1 and l_1 have two non-zero states (WEAK and STRONG), including these effects results in estimating two $\theta_{1,2}$ parameters, $\theta_{WEAK,2}$ and $\theta_{STRONG,2}$. $\theta_{WEAK,2}$ captures the level of association of the design interface matrix and team interaction matrix at k_1=*WEAK* whereas $\theta_{STRONG,2}$ captures the level of association at k_1=*STRONG*. Hence, a formal hypothesis testing of H3 can be expressed as follows:

H3: $\theta_{STRONG,2} - \theta_{WEAK,2} > 0$

In order to test the moderating effect of design interface strength on the effects of organizational boundaries (H4) we extend our model with structural parameter *ACROSS* to examine the third-order interaction effects *ACROSS* x k_1 x k_2 (due to symmetry identical to *ACROSS* x l_1 x l_2). As before, including this interaction effect means estimating two parameters (one for each non-zero strength). Hence, we estimate $\theta_{ACROSS,WEAK,2}$ and $\theta_{ACROSS,STRONG,2}$. However, we are interested in testing whether the level of association is stronger (across boundaries) for cases with strong design interfaces (H4). Hence, a formal hypothesis testing of H4 is formulated as follows:

H4: $\theta_{ACROSS,STRONG,2} - \theta_{ACROSS,WEAK,2} > 0$

In order to test H3 and H4 we follow a similar procedure to the one described in step 5 of the previous sub-section, fitting the newly defined log-linear models to data. For the purpose of brevity, Table A2 shows only the estimates of relevant parameters of the models used to test H3 and H4 with their respective likelihood-ratio statistics, G^2, and their numbers of degrees of freedom.

Model 1 (exchange) includes second-order exchange parameters and serves as our base model. Model 2 (clustering) includes parameters to capture tendencies to cluster design interfaces and team interactions within boundaries which, as expected, are all significantly negative. Model 3 (association) includes association parameters between design interfaces and team interaction for both WEAK and STRONG design interfaces. Note that both $\theta_{WEAK,2}$ and $\theta_{STRONG,2}$ are significantly positive (ΔG^2 = 224.70, Δdf = 2, p <0.001), indicating that there is a strong association between design interfaces and team interaction at both levels. To test whether the difference of these parameters is significantly positive we estimate a reduced model that includes a single parameter to capture the association of design interfaces and team interactions. Since this reduced model exhibits a significantly worse goodness-of-fit than model 3, we can conclude that $\theta_{STRONG,2}$ is significantly greater than $\theta_{WEAK,2}$ (supporting H3).

Model 4 includes third-order interaction effects with *ACROSS* (i.e. $\theta_{ACROSS,STRONG,2}$ and $\theta_{ACROSS,WEAK,2}$). Consistent with our binary log-linear models, these parameters are significantly positive (ΔG^2 =17.89, Δdf = 2, p <0.001) indicating that the *pure* tendency for alignment is stronger across boundaries for both WEAK and STRONG interface strengths. Yet, we are interested in testing whether the difference of the newly included parameters is significantly positive. To do so, we estimate a reduced model that captures, with a single parameter, the association of design interfaces and team interactions across boundaries. Since the reduced model does NOT exhibit a significantly worse goodness-of-fit than model 4, we cannot conclude that $\theta_{ACROSS,STRONG,2}$ is significantly different than $\theta_{ACROSS,WEAK,2}$, hence H4 is not supported.

Finally, we ran additional models including the indicator variable *MODULAR* to test whether there was a moderating effect of system modularity on the effect of design interface strength and found no evidence of such effect (models not included in Table A2).

Table A1. Results of Log-Linear Analysis (Binary Relations)

Parameters	Model 1 (Independent)	Model 2 (Base)	Model 3 (ACROSS)	Model 4 (MODULAR)	Model 5 (FINAL)
Parameters for the design interface matrix					
α_{1FAN}	0.4321	0.3632	0.3743	0.2598	0.3128
α_{1LPC}	0.2090	0.3632	0.3161	0.2603	0.2562
α_{1HPC}	-0.0120	0.1061	0.0517	0.0108	-0.0129
α_{1BD}	-0.0172	-0.2137	-0.1407	-0.2924	-0.2042
α_{1HPT}	-0.5653	-0.5731	-0.5401	-0.6408	-0.6033
α_{1LPT}	-0.0771	-0.0533	-0.0621	-0.1451	-0.1301
α_{1MC}	-0.2567	-0.1496	-0.1596	0.0862	0.0280
α_{1EC}	0.2869	0.1575	0.1606	0.4612	0.3535
β_{1FAN}	-0.7416	-0.6587	-0.6993	-0.7507	-0.7918
β_{1LPC}	-0.0672	0.1355	0.0896	0.0263	0.0284
β_{1HPC}	0.0509	0.2169	0.1834	0.1045	0.1177
β_{1BD}	-0.0957	-0.4262	-0.3557	-0.5249	-0.3993
β_{1HPT}	0.4363	0.3360	0.3757	0.2159	0.3253
β_{1LPT}	-0.3862	-0.2229	-0.2226	-0.3289	-0.3016
β_{1MC}	0.3176	0.1773	0.1667	0.4843	0.3522
β_{1EC}	0.4856	0.4420	0.4624	0.7738	0.6691
θ_1	-2.1306	0.2063	0.1198	1.0958	0.3526
ρ_1	3.9891	3.5224	3.4502	3.5083	3.4532
Parameters for the team interaction matrix					
α_{2FAN}	0.2779	0.2312	0.2269	0.0776	0.1370
α_{2LPC}	0.0062	-0.2327	-0.2088	-0.3657	-0.3203
α_{2HPC}	-0.0313	-0.1437	-0.1321	-0.2783	-0.2500
α_{2BD}	0.0009	0.2323	0.2945	0.0904	0.2538
α_{2HPT}	-0.3079	-0.0678	-0.0237	-0.1989	-0.1104
α_{2LPT}	-0.0197	0.0706	0.1023	-0.0797	-0.0017
α_{2MC}	-0.3880	-0.3491	-0.4667	0.0249	-0.1738
α_{2EC}	0.4620	0.2595	0.2076	0.7297	0.4655
β_{2FAN}	-0.5182	-0.2337	-0.3480	-0.4008	-0.4945
β_{2LPC}	-0.1838	-0.3553	-0.3426	-0.5244	-0.4493
β_{2HPC}	-0.1618	-0.3120	-0.3185	-0.4837	-0.4220
β_{2BD}	0.2767	0.5694	0.6579	0.3802	0.6202
β_{2HPT}	0.3069	0.2591	0.3355	0.0683	0.2941
β_{2LPT}	-0.5070	-0.3638	-0.3993	-0.5360	-0.5222
β_{2MC}	0.4625	0.4057	0.4521	0.9271	0.6903
β_{2EC}	0.3249	0.0307	-0.0371	0.5690	0.2834
θ_2	-2.1239	-0.4895	-0.8373	0.9959	-0.3599
ρ_2	3.5191	2.5124	2.206	2.4583	2.1749
Second-order, third-order and fourth-order interaction parameters					
$\theta_{1,2}$		**3.2314**[a]	3.2902	3.2318	2.9670
ρ_{12}		-0.2101[a]	-0.3290	-0.2453	-0.3603
$\theta_{ACROSS,1}$			**-0.2081**		-0.6137
$\theta_{ACROSS,2}$			**-1.0284**		-1.4850
$\theta_{ACROSS,1,2}$			**0.9016**[b]		0.5645
$\theta_{MODULAR,1,2}$				**0.0547**[c]	-0.4831
$\theta_{MODULAR,ACROSS,1,2}$					**-0.3487**[d]
Goodness-of-fit					
G^2 †	5243.00	4298.70	3705.60	4134.61	3670.23
df	5692	5690	5687	5687	5683

† G^2 provided by statistical packages is incorrect because the unit of analysis is still the dyad rather than the group of dyads. We calculate G^2 as follows (Fienberg and Wasserman 1981):

$$G^2 = 2\sum_{i<j}\sum_{k_1,l_1,k_2,l_2} y_{ijk_1l_1k_2l_2}\log(y_{ijk_1l_1k_2l_2}/\hat{y}_{ijk_1l_1k_2l_2}) = -2\left[\sum_{r<s}\sum_{k_1,l_1,k_2,l_2} w_{rsk_1l_1k_2l_2}\log(\hat{w}_{rsk_1l_1k_2l_2}/[G_rG_s]) + \sum_{r}\sum_{k_1,l_1,k_2,l_2} w_{rrk_1l_1k_2l_2}\log(\hat{w}_{rrk_1l_1k_2l_2}/[G_r(G_r-1)])\right]$$

a: The unconstrained model against which significance of $\theta_{1,2}$ was assessed is model 1. Hence, $\Delta G^2 = 943.71$, $\Delta df = 1$, $p < .001$. Model 2 does not significantly improve its goodness-of-fit when the term ρ_{12} is added (ΔG^2=0.59, $\Delta df = 1$, hence $p > 0.1$)

b: The unconstrained model against which the hypothesis (H1) is tested includes second-order parameters with ACROSS (G^2 =3719.18, $df = 5688$). Hence, $\Delta G^2 = 13.58$, $\Delta df = 1$, $p < 0.001$

c: The unconstrained model against which the significance is assessed includes the second-order parameters with MODULAR (G^2 = 4134.90, df = 5688). Hence, $\Delta G^2 = 0.29$, $\Delta df = 1$, $p > 0.1$

d: The unconstrained model against which the hypothesis (H2) is tested includes second and third-order interaction terms with both ACROSS and MODULAR (G^2 = 3706.68, df = 5684). Hence, $\Delta G^2 = 36.45$, $\Delta df = 1$, $p < .001$)

Table A2. Results of Log-linear Analysis with Valued Design Interfaces

Parameters	Model 1 (Exchange)	Model 2 (Clustering)	Model 3 (Association)	Model 4 (Association across)
Second order exchange parameters				
$\rho_{WEAK,2}$	2.0508 [a]	1.8575	-0.3786	-0.3099
$\rho_{STRONG,2}$	2.6590 [a]	2.2256	-0.5228	-0.4705
Second-order interaction parameters with ACROSS				
$\theta_{ACROSS,WEAK}$		-0.8299 [b]	-0.7179	0.0204
$\theta_{ACROSS,STRONG}$		-1.3483 [b]	-1.2106	-0.6157
$\theta_{ACROSS,2}$		-1.2892 [b]	-1.2596	-1.1904
Second-order association parameters				
$\theta_{WEAK,2}$			2.9187 [c,c1]	3.1348
$\theta_{STRONG,2}$			3.5711 [c,c1]	3.6950
Third-order interaction parameters with ACROSS				
$\theta_{ACROSS,WEAK,2}$				1.1398. [d,d1]
$\theta_{ACROSS,STRONG,2}$				0.74.18 [d,d1]
Goodness-of-fit				
G^2 †	5399.20	4739.95	4515.25	4497.36
df	8535	8532	8530	8528

† G^2 is determined as indicated in Table A1.

a: The unconstrained model against which the significance is assessed is the independent model (equation A7) whose G^2 =6112.07, df = 8537. Hence, ΔG^2 =712.87, Δdf = 2, $p < 0.001$

b: The unconstrained model against which the significance is assessed is Model 1. Hence, ΔG^2 = 659.25, Δdf = 3, $p < 0.001$

c: The unconstrained model against which the significance is assessed is Model 2. Hence, ΔG^2 = 224.70, Δdf = 2, $p < 0.001$

c1: The model against which the significance of the parameters difference is assessed is a reduced model with a single association parameter, G^2 = 4522.63, df = 8531. Hence, ΔG^2 = 7.38, Δdf = 1, $p < 0.01$

d: The unconstrained model against which the significance is assessed is Model 3. Hence, ΔG^2 = 17.89, Δdf = 2, $p < 0.001$

d1: The model against which the significance of the parameters difference is assessed is a reduced model with a single association parameter across boundaries, G^2 = 4498.85, df = 8527. Hence, ΔG^2 =1.49, Δdf = 1, p >0.1