INSEAD

The Business School for the World

Faculty & Research Working Paper

Component Modularity, Team Network Structure, and the Attention to Technical Interdependencies in Complex Product Development

Manuel E. SOSA
Martin GARGIULO
Craig M. ROWLES
2007/08/TOM/OB (revised version of 2006/43/TOM/OB)

Component Modularity, Team Network Structure, and the Attention to Technical Interdependencies in Complex Product Development

by

Manuel E. Sosa*

Martin Gargiulo**

and

Craig M. Rowles***

We appreciate the assistance of the engineers at Pratt & Whitney Aircraft for their collaboration during the data collection of this study. We also appreciate the comments from participants in the innovation forum at the Carnegie Bosch Institute conference (Stuttgart, Germany, Fall 2005) and the Wharton Technology mini-conference (April, 2006). We are grateful for insightful comments of Jonghoon Bae, Steven Eppinger, Gokhan Ertug, Christoph Loch, and Luk Van Wassenhove. We also appreciate the support of the INSEAD/Wharton Alliance Center for Global R&D.

* Assistant Professor of Technology Management at INSEAD, Boulevard de Constance, 77300 Fontainebleau, France; manuel.sosa@insead.edu. Tel: +33 (0)1 60 72 45 36. Fax: +33 (0)1 60 74 61 79.

** Associate Professor of Organisational Behaviour, Coordinator, Organisational Behaviour Area at INSEAD, 1 Ayer Rajah Avenue, Singapore, 138676

*** Pratt & Whitney Aircraft, East Hartford, Connecticut, USA

Component Modularity, Team Network Structure, and the Attention to Technical Interdependencies in Complex Product Development

Abstract

The development of complex products poses substantial operational and organizational challenges to established firms. Previous research has shown that coordinating technical interdependencies is vital for the successful development of complex products. We integrate research streams in product development and organizational theory to study the determinants of the capability of teams to attend to technical interdependencies in complex product development projects. We hypothesize that the lack of connectivity among product components (i.e., component modularity), and teams' communication network structure significantly influence their attention to critical technical interdependencies. We test our hypotheses by examining the network of components of a large commercial aircraft engine and the technical communication network structure of the organization that designs it. Our findings suggest that both component modularity and the density of the team's communication network play crucial roles in the capability of teams to attend to their critical interdependencies. Moreover, we found that communication network density is less beneficial for teams designing more modular components.

Keywords: Product Architecture; Modularity; Product Development Organizations; Social Networks.

1. Introduction

The development of complex products poses substantial operational and organizational challenges to firms. In the operational (or product) domain, these challenges are met by breaking down complex products into systems, which may be further decomposed into smaller components (e.g., Simon 1981, Suh 2001). The product decomposition determines the architecture of the product, which is defined by the way components interface with each other so that the product can fulfill its functional requirements (Ulrich 1995, Ulrich and Eppinger 2004). In the organizational domain, firms meet the challenges of complex product development by assigning each component to a team responsible for its design and for its integration with other components to ensure product functionality (e.g., Clark and Fujimoto 1991). The interfaces among product components define technical interdependencies among design teams, making effective collaboration across interdependent teams one of the most critical challenges in complex product development (Thompson 1967, Galbraith 1973, Smith and Eppinger 1997, Mihm et al. 2003).

Although attention to technical interdependencies is crucial for successful product development, teams typically ignore (or pay marginal attention to) a number of interdependencies during the development process (Sosa et al. 2004). Some level of neglect is perhaps unavoidable given the cognitive and resource limitations typically faced by teams (Simon 1947, Ocasio 1997). Lack of attention to non-critical or standardized interdependencies may not be ultimately significant (Sosa et al. 2004), but the neglect of critical interdependencies can have serious negative consequences for firms. For example, in a study of the semiconductor photolithography alignment equipment industry, Henderson and Clark (1990) found that novel interfaces between existing components were often neglected by design teams, causing established firms to lose their leading position in the market. In the auto industry, Ford and Firestone lost billions of dollars for poorly managing the interface between the tire design and the vehicle dynamics of the Ford Explorer (Pinedo et al. 2000). In the aerospace industry, Airbus' development of its A380 plane has suffered significant delays due to the lack of attention to some critical interfaces between the wiring systems and the fuselage (Gumbel 2006, Hollinger and Wiesmann 2006).

Despite the importance that critical technical interdependencies may have for the development of complex products, little is known about the factors that affect the capability of teams to attend to such interdependencies.[1] This is the topic addressed in this paper. *Why are some teams better than others attending to their critical technical interdependencies? Is it because of some attributes of the components they design, or because of their communication patterns with other teams in the organization, or both?* Building on existing research in product development (e.g., Ulrich 1995, Krishnan and Ulrich 2001) and in social network theory (e.g., Coleman 1990, Burt 1992, Ahuja 2000), we argue that both the product architecture and the structure of the team's communication network influence the team's ability to collaborate with other teams in the development process to attend critical technical interdependencies. A better understanding of how product architecture and communication networks affect successful product development can help managers make better decisions about which components require particular attention during the design process, as well as to consider the implications for other relevant design decisions such as outsourcing, off-shoring, and life-cycle management of product components.

Research in product development has mostly focused on understanding the structure of the development process and its impact on the organization. A stream of work focuses on minimizing and identifying the set of activities that are more likely to iterate and create rework so that they are managed with special attention (e.g., Eppinger et al. 1994, Smith and Eppinger 1997, Mihm et al. 2003). The minimization of interdependencies is central to the notion of "modularity," which focuses on reducing the connection across sets of physical components, development activities and/or organizational teams (e.g., Ulrich 1995, Baldwin and Clark 2000). Product modularity has been associated with flexibility to adapt and generate product variety (Ulrich 1995). It also facilitates the evolution of designs and industries by allowing teams to work independently on loosely coupled problems (Baldwin and Clark 2000). In the process domain, modularity has been proposed as a

---

[1] We broadly follow the definition of *attention* provided by Ocasio (1997: 189) who defines it as "the noticing, encoding, interpreting, and focusing of time and effort by organizational decision-makers on both (a) *issues*... and (b) *answers*." Hence, in the context of this paper, attention happens when design teams identify a design issue between the components they develop and interact about it.

mechanism to reduce cost in design testing and to mitigate design oscillations in complex engineering projects (Loch et al. 2001, Mihm et al. 2003). In the organizational domain, modularity has also been proposed as a source of flexibility to the organization (e.g., Sanchez and Mahoney 1996, Schilling 2000, Ethiraj and Levinthal 2004). Regardless of its focus, the research in product development coincides with earlier studies on R&D teams (e.g., Allen 1977, Brown and Eisenhardt 1995) in assuming that intense communication is necessary to execute interdependent design tasks.

The organizational literature inspired in social network theory has been mostly concerned with how communication networks can help or hinder an actor's ability to collaborate with other interdependent entities. A substantial body of evidence suggests that this ability is enhanced by mutual trust and willingness to help others, which are associated with densely connected communication networks (Coleman, 1990). The more collaboration is essential for task outcomes, the more likely that dense networks in which relationships are likely to be embedded in common third parties would be positively associated with such outcomes (e.g., Ahuja 2000; Obstfeld 2005).

In this paper, we build on these different literature streams to offer three important contributions. First, we take a network analytical perspective to integrate product development and organizational theories to explain a phenomenon that is equally important for both fields—namely, how product structure at the component level and informal communication structure at the team level affect the capability of teams to attend to critical technical interdependencies in complex product development. Second, we propose a network analytical approach to measure the degree of modularity of a given component based on the extent to which a component is interdependent with other components that are also interdependent. Third, and more importantly, we show that teams embedded in densely connected communication networks have greater capacity to attend to their critical interdependencies, especially when these teams are responsible for designing highly connected (i.e. non-modular) components.

2. Framework and Hypotheses

A complex product can be conceived as a network of components, with each component being a "node" and the interfaces between components being the "edges" (or ties) of the network. Similarly, the interactions among teams responsible for designing or for integrating such components can be viewed as a social network, with the teams being the nodes and the technical communication between them the edges of the network. Theoretically, an identified interface between two components should trigger some communication between the teams in charge of those components to address their technical interdependence. In some cases, however, communication between teams might also uncover previously unidentified interfaces between components. Yet, not all teams whose components are linked through an interface actually interact during the project implementation phase (Henderson and Clark 1990, Sosa et al. 2004), causing some interfaces to go unattended.

Superimposing the network that maps the interfaces that define the product architecture onto the network that portrays the observable technical communication structure of the design and integration teams, one can identify five types of interdependencies between teams (Figure 1). First, *attended interdependencies* occur when a design interface between two components is identified and the teams designing those components interact about it (interdependencies *a*, *b* and *c* in Figure 1). Second, *unattended interdependencies* occur when a design interface is identified and the corresponding teams do not interact (interdependence *d*). Third, *uncovered interdependencies* happen when the teams interact (for technical purposes) but no design interface between them has been identified by system architects (interdependence *e*). Fourth, *lack of interdependence* occurs when there is neither design interface nor team interaction such as the lack of ties between teams 2 and 4 in Figure 1. Note, however, that this last case can also be a "truly missed" interdependence that is neither identified by system architects nor attended through informal team interactions. These cases would be difficult to identify in practice, yet from a theoretical viewpoint we still need to recognize their possible existence. Finally, *external interdependencies* occur when teams which are not directly responsible for the design of a component interact with teams designing components. This may be the case with teams that are in charge of overseeing the integration among different aspects of the design but are not responsible for designing any specific component, as is the case with teams 5 and 6 in Figure 1.

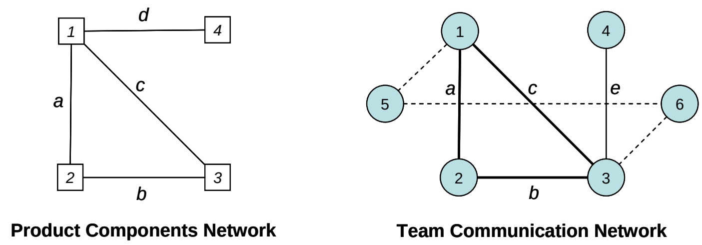

Figure 1. Hypothetical networks of components and teams

Prior research in product development has used a network analytical approach to examine the architecture of complex products (Sosa et al. 2003, Sosa et al. 2006) and to study the factors that explain the existence of unattended and uncovered interdependencies in design organizations (Sosa et al. 2004).[2] In the organizational theory domain, Gargiulo and Benassi (2000) have shown how the structure of an actor's communication network can affect this actor's ability to coordinate interdependencies. In this paper, we study how the "modularity" of each component and the structure of the design team's communication network affects the team's ability to attend to its critical technical interdependencies.

Effects of component modularity

We take a resource utilization perspective on complex product development (Adler et al. 1995, Loch and Terwiesch 1999) to study the effects of component modularity on teams' attention to their technical interdependencies. From this perspective, teams are considered as "workstations" responsible for designing a physical or functional component of the product, and a team's workload is largely determined by the characteristics of the component it designs. The capacity of the team to process its workload determines the team's "delay" in attending to the next "job" in the design task list (Adler 1995). Hence, the busier the team is, the longer design tasks "wait" to be processed, and the lower the probability that the team will be able to attend to all its critical technical interdependencies.

[2] Note that the focus and unit of analysis of this paper differ significantly from Sosa et al. (2003, 2004). In this paper the unit of analysis is the team and component, while the unit of analysis in Sosa et al. (2003, 2004) is the interdependence itself. As a result we study here how network structures of components and teams affect the ability of teams to coordinate with other teams in a complex product development effort. As for Sosa et al. (2006), such a paper defines component modularity based on the notion of centrality (Freeman, 1979) and relates it to component redesign decisions without considering at all the communication network of the organization. In this paper, on the other hand, the component modularity definition we operationalize is based on the notion of constraint (Burt 1992).

Experienced managers in mature companies like the one we studied are likely to anticipate their design teams' workload based on some key attributes of the components designed by the teams. Component complexity (as determined by the component's internal structure) and component redesign (i.e. the degree to which a component changes relative to a previous generation of the same product) are typically taken into account when managers assign team resources to prevent significant delays for processing design tasks. When developing complex products, however, anticipating fully the workload resulting from the direct and indirect connectivity among components may be difficult, even for experienced managers. Yet, unanticipated workload may significantly affect the capacity of teams to attend their critical dependencies during the design process. Empirical research on engineering changes in the aerospace and automotive industry provides evidence for this observation (e.g. Allen 1966, Loch and Terwiesch 1999, Terwiesch et al. 2002, Clarkson et al. 2004). The upshot of this research is that engineering changes in complex product development occur because engineers and managers cannot fully anticipate the impact that all their local design decisions have on other components in the product. This impact, in turn, is a function of how individual components are connected (directly and indirectly) to other components within the product.

Component modularity is the main structural property that captures the lack of connectivity of a component with other components in the product. The common notion associated with modularity is that it decouples otherwise interdependent groups of elements. Decoupling of elements generates flexibility to adapt to changes. Modular products are easier to change and upgrade, facilitating product variety (Ulrich 1995). In the process domain, modular development processes are less prone to design iterations and rework (Smith and Eppinger 1997, Mihm et al. 2003). In addition, the work structure associated with modular components is less complex, which reduces the effects of ambiguity and uncertainty in development projects (e.g., Pich et al. 2002, Ethiraj and Levinthal 2004).

Since complex products are hierarchically decomposed into systems and components for design purposes (Simon 1981), we can define modularity at the product, system and component level. Modular products include a one-to-one mapping of product functions to product components, resulting in "decoupled interfaces between components" (Ulrich 1995: 422). Within the product, previous research has focused on how systems (i.e., groups of components) share interfaces with other systems (without explicitly considering how components interact within systems). As a result, modular systems are defined as groups of similar components whose interfaces are clustered among few adjacent systems (Sosa et al. 2003). Finally, modularity can be also defined at the level of the components within a product as "the level of independency of a component from the other components within a product" (Sosa et al. 2006: 8). Building on such a general definition, we operationalize component modularity as *the degree to which a component lacks direct and indirect interfaces with other components in the product*. This is consistent with the original notion of modular and integral architectures introduced by Ulrich (1995, p. 422) in which the former ones specify de-coupled interfaces between components while the latter ones include coupled interfaces between them.

Our definition of modularity at the component level allows us to distinguish between "modular" components that fulfill one or a few functions due to their design independency from other components and "integral" components that depend on many others to fulfill their function in the product. From a network analytical perspective, a modular component has fewer and/or weaker *direct* and *indirect* links (i.e., technical design interfaces) with other components within the product. Modular components are less "connected" with other components and can be viewed as more flexible from a product design perspective because it has more degrees of freedom (Suh 2001). It is important to emphasize that since our unit of analysis is the component (and the team that designs it), we focus on modularity at the component level rather than at the system or product level.

The relative independence of modular components makes them less likely to generate highly coupled design tasks, which are the primary source of design rework and iterations during the development process (Smith and Eppinger 1997, Mihm et al. 2003). The more "modular" a component is, the more likely it is to generate a predictable workload structure (Ulrich 1995, Sanchez and Mahoney 1996, Baldwin and Clark 2000, Loch et al. 2001). Other things being equal, teams designing modular components should be more likely to handle a larger proportion of their critical technical interdependencies. Conversely, teams in charge of components that lack modularity due to their tight integration with many other components in the product should have less spare capacity, which in turn is likely to make them less likely to handle a large proportion of their critical technical interdependencies.

The negative effect of low component modularity on a team's ability to attend to technical interdependencies could be in part compensated with assignment of additional engineering resources. Yet, because the effects of low modularity often go beyond the interfaces with adjacent components, it is difficult to adequately assess *a priori* the effects on team workload (Pich et al. 2002, Mihm et al. 2003, Clarkson et al. 2004). These indirect interfaces between two or more adjacent components are an important source of the additional design iterations that consume most of the capacity of teams to attend technical interdependencies, rendering them less able to attend a larger proportion of their critical technical interdependencies. Teams in charge of modular components face a more predictable workload, whereas teams responsible for non-modular components are more likely to confront unforeseen design interactions that can strain the team resources. Thus, we expect that, holding team resources constant, component modularity should have a positive effect on the team's ability to handle technical interdependencies. Hence:

H1: The proportion of critical technical interdependencies attended by a team increases with the level of modularity of the component designed by that team.

Effects of the communication network structure

The very nature of the design process requires intense collaboration and information sharing among interdependent teams. Organization theory suggests that such interactions can be facilitated (or hindered) by the structure of the informal communication network among such teams. Research on the effects of communication networks on team performance dates back to early studies of R&D organizations that associated both intense and cohesive internal communication with high-performing organizations (Allen 1977) while also recognizing the critical role of "boundary spanners" (Tushman 1977) in linking the team with the rest of the organization.

The idea that cohesive communication networks can facilitate collaboration and exchange of information among interdependent actors has been more recently object of systematic inquiry in organization theory. Building on insights from sociological theory (Coleman 1990; Granovetter 1985), scholars have emphasized the role of the structure of the communication network on actors' ability to collaborate in complex tasks. Arguments have focused on the positive effects of densely connected networks on the production of social norms and sanctions that facilitate cooperative exchanges among individuals, groups, or organizations. In such networks, dyadic relationships between any two actors are likely to be "embedded" in common third parties—that is, for a pair of actors i,j linked by a relationship z such that $z_{ij} > 0$, there is one or more actors q such that $z_{iq}>0$ and $z_{jq} > 0$.

Three mechanisms are commonly invoked to account for the positive effects of densely connected networks on product development efforts: information sharing, fostering of common culture and norms, and reciprocity enforcement. First, when a focal actor is surrounded by a densely connected network, other actors in that network have both direct and indirect information (that is, through common third parties) on the motives and needs of the focal actor, removing barriers to knowledge sharing and facilitating collaboration in complex activities such as innovation (Ahuja 2000; Obstfeld 2005). Second, widespread interactions among members of the network may also facilitate the emergence of a common culture and norms that decreases the impact of competitive and motivational impediments to cooperation (Reagans and McEvily, 2003; Oh, Chung, and Labianca 2004). Such culture can promote the free flow of rich information and complex knowledge that is often essential to superior performance (Hansen, 1999). Third, the presence of common third parties can create additional pressures to collaborate (Gargiulo, 1993; Bae and Gargiulo 2004). In the presence of common third parties, failure to help one actor can have negative consequences on the help one might expect to obtain from those third parties, creating reputation effects that act as incentives to collaborate (Raub and Weesie 1990). Common third parties play thus a critical role in enforcing norms of collaboration and reciprocity within the network (Coleman, 1990).

The benefits associated with densely connected communication networks should be apparent in the context of complex product development in two different ways. First, because dense networks facilitate information sharing, a focal team embedded in a densely connected communication network should find it easier exchanging information about all potential design issues that might affect its component. This, in turn, should make it easier to coordinate the component interfaces with other teams, reducing uncertainty and unexpected design iterations. Second, because densely connected networks enforce reciprocity, a focal team embedded in a dense communication network is more likely to receive help from other teams in that network. This mutual help creates flexible capacity within the group, which can free up resources in the focal team. Indeed, flexible capacity has been found an efficient way to reduce delays in the presence of workload variability (Adler et al 1995, Loch and Terwiesch 1999). Thus, the more a design team is embedded in a cohesive, densely connected communication network, the more it will enjoy the benefits of a collaborative environment, and the more likely it will be to attend to the critical technical interdependencies that result from the interfaces with other components in the product design process.

H2: The proportion of critical technical interdependencies attended by a team increases with the density of the team's communication network.

The moderating effect of component modularity

Hypothesis 2 proposes a positive relationship between network density and attention to critical technical interdependencies. A growing organizational literature, however, suggest that the effects of network structures on outcomes can be contingent on task characteristics (e.g., Rowley, Behrens, and Krackhardt 2000, Ostbfeld 2005). Consistent with this idea, the benefits of network density on teams' attendance to critical technical interdependencies could be mitigated by the characteristics of the task facing the teams. If the benefits of dense networks result from their role in promoting a collaborative environment that helps teams attend their critical interdependencies, such benefits should be more apparent for teams whose components are more tightly integrated into the product architecture—that is, components with low modularity—and less so for teams in charge of highly modular components. By definition, teams in charge of modular components have fewer direct and indirect interdependencies with other teams in the development process. Since modular components are more independent from other components, their design teams are less affected by decisions taken by other teams. As such, they are less likely to experience capacity constraints resulting from unanticipated complications in the direct and indirect interfaces between their modular component and other components in the product. As modularity decreases, the likelihood of unanticipated complications in the design process is likely to increase and the benefits of being embedded in a densely connected communication network should be more apparent.

If the previous reasoning is correct, the benefits teams obtain from being embedded in densely connected networks should be contingent on the level of modularity of their respective components. As modularity increases, the returns to density in terms of enhanced ability to attend to technical interdependencies should decrease. In other words, while all teams benefit from the collaborative environment fostered by dense communication networks, teams in charge of modular components do so less than those designing highly integrated components. Thus:

H3: The effect of a team's communication network density on the team's attention to its critical technical interdependencies decreases with the modularity of the team's component.

3. Data and Methods

We test our hypotheses by studying the detailed design phase in the development of a large commercial aircraft engine at Pratt & Whitney, the PW4098 that equips the Boeing 777 two-engine aircraft. Several factors justified the selection of the 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 development team to each component facilitated the implementation of our research approach. Third, the model studied was the most recent engine program to complete design and development: almost all team members involved in the detail design phase were still accessible. Finally, the derivative engine studied was part of a family of engines with two new derivative engines already planned, whose development could gain directly from this study.

We captured the technical interface network of the 54 components that comprised the engine and the communication network of 60 teams (54 design teams and 6 integration teams) that developed the engine components and evaluated the overall engine performance (see Sosa et al. 2003 and Sosa et al. 2004 for details). We collected data from multiple sources. Product network data was obtained by interviewing several experienced engine architects. Data on communication among teams was gathered by interviewing and surveying key members of the design teams using standard sociometric questions (Marsden 1990). The use of multiple individual responses and multiple data sources to construct the network variables minimized the effect of response bias when testing our network hypotheses.

Product network data

Our product data capture the breakdown of the engine structure into eight systems and 54 components, as well as the five types of design dependencies among those components: spatial constraint, structural constraint due to transfer of loads, exchange of material, exchange of energy and exchange of information. Each type of design dependence x between components i and j was coded as either non-critical ($x_{ij,type}$ = 1) or critical ($x_{ij,type}$ = 2), depending on the impact on functionality of component i due to its dependence from component j. (See appendix A for further technical details).

We capture the total *technical interdependence* of the design interface between components i and j, m_{ij}, as the sum of their design dependencies $x_{ij,type}$ due to spatial, structural, material, energy and/or information functional requirements in both directions, standardized by the maximum technical interdependence detected between any two components in the engine:

$$m_{ij} = \frac{\sum_{type} x_{ij,type} + \sum_{type} x_{ji,type}}{\max\left(\sum_{type} x_{pq,type} + \sum_{type} x_{qp,type}\right)}, \qquad i \neq j; \quad p \neq q; \quad p=[1, \textit{Num. of comps.}]; \quad q=[1, \textit{Num. of comps.}]$$

This measure yields a valued symmetric network of product components where $0 \leq m_{ij} \leq 1$ and $m_{ij} = m_{ji}$, with reciprocal design dependencies producing comparatively stronger technical interdependencies. We found eight distinct pairs of components whose technical interdependence reached the maximum design interdependence strength (m_{ij} = 1). We will use m_{ij} to develop our measure of component modularity based on the direct and indirect technical interdependencies of each component.

Communication network data

The teams' communication network is defined by their task-related interactions with other teams during the design phase of the project. We capture task-related interactions between the 60 teams — that is, 54 design teams plus six integration teams—involved in the ten-month detailed design phase of 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).

Responses were obtained from team leaders and validated (and corrected, if necessary) by at least one key member of each design team surveyed. By the time of the data collection, all but one of the team leaders were still at P&W, yet eight of them had moved to different groups in the company. We were able to survey key members of 57 of the 60 design teams. The three teams whose direct responses were missing were the interface team at Boeing (a system integration team), one team in the high-pressure compressor system group, and one team in the externals and controls system group. These teams, however, were targets of information requests by other teams in the development process, so we kept them in the network.

Presentations to describe the overall objective of and terminology used during the data collection were made in two sessions to over two-thirds of the respondents, while the other ones were briefed individually afterwards. Specific definitions of survey terms (which had been previously discussed with the program managers) were reviewed and clarified as necessary. Each respondent was asked to identify (from the team roster for this project) the other design teams from which his team requested technical information. Then we asked respondents to assess in a single scale the frequency and importance of their "peak" technical interactions during the detail design period. 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"). Similar to previous research in technical communication (e.g., Allen 1977, Morelli et al 1995, Van den Bulte and Moenaert 1998), we focus on the presence or absence of *significant* information exchange as the binary variable of interest. We define *significant* information exchange as the technical communication that was relevant during the design phase due to their criticality and/or frequency (See additional details in the appendix B).

Based on these data, we reconstructed the informal communication network among the 60 teams involved in the project. To do so, we assumed that a communication tie z_{ij} between two teams i and j exist whenever either team reported *significant* requests for task-related information from the other team. In other words, we assume that the team that was the target of a request for information typically will respond to this request, hence establishing a communication tie with the requesting team. This is a reasonable assumption in the context we investigate (Morelli et al. 1995, Eppinger

2001). The resulting communication network contains 1,770 potential communication ties among the 60 teams, out of which 443 ties were observed—that is, 25 percent of the ties were present.

Dependent variable

The dependent variable in this study is the *proportion of critical technical interdependencies* attended by each of the 54 design teams. For a given team *i,* this proportion is the fraction between the weighted number of critical technical interdependencies attended by this team and the total weighted number of critical interdependencies facing the team. A technical interdependence between teams *i* and *j* is considered "attended" when a binary technical communication tie between the teams was observed (i.e. $z_{ij} = 1$). By counting only critical dependencies between components we seek to focus on interdependencies that, if left unattended, were highly likely to cause serious problems (such as rework and cost overruns) at later stages of the process. The weight of the interdependence between teams *i* and *j* is computed as a direct function of the number of critical design dependencies contained in the interface between their respective components. Thus, the proportion of attended technical interdependencies for team *i,* Y_i, is computed as follows:

$$Y_i = \frac{\sum_j \left[\left(\sum_{type} x_{ij,type,c} + \sum_{type} x_{ji,type,c} \right) z_{ij} \right]}{\sum_j \left(\sum_{type} x_{ij,type,c} + \sum_{type} x_{ji,type,c} \right)}$$

where $x_{ij.type,c}$ corresponds to a type of design dependence between the components and $x_{ij,type,c} = 1$ if x_{ij} is critical. Also, $z_{ij} = 1$ if there is a communication tie between teams *i* and *j* and 0 otherwise.

To compute the denominator of this fraction we proceeded as follows. First, we started with all 1,226 design dependencies x_{ij} between components (as identified by the systems architects) and retained only dependencies of various types (spatial, structural, energy, material, or information) that were classified as "critical" with respect to their the impact on functionality of the respective engine component. This yielded 1,118 critical design dependencies distributed across 327 component interfaces. Second, from this figure we discarded those dependencies that corresponded to components whose joint probability of redesign was less than 10%. (We provide an account of our measure of component redesign in our description of the control variables below.) In doing so, we assumed that interfaces between components with few changes from the prior generation of the engine were unlikely to require significant attention from the respective teams, regardless of their design dependencies. This further reduced our count to 1,110 critical design dependencies spread across 322 interfaces.

Finally, we added 46 additional interdependencies uncovered by the teams during the design process. In doing so, we assumed that cases of observed communication between two teams corresponded to the presence of *one* critical dependence between their respective components, even though that dependence was not identified by the system architects. We believe that this is the most reasonable assumption because communication ties retained only cases of significant and repeated technical interaction between teams. Hence, these ties were likely triggered by real critical dependencies not identified by the systems architects. In the absence of information about the real number of dependencies that triggered an observed communication tie between the teams, it is also reasonable to set this number to one because interfaces with many dependencies would have been less likely to be neglected by the systems architects. This results in a total of 1,156 critical design dependencies spread across 368 interfaces among the 54 engine components.

To compute the numerator of the fraction that defines our dependent variable, we assumed that all critical dependencies between two components were attended insofar as there was an observed communication tie between the respective teams (z_{ij} = 1). This, of course, includes the cases of uncovered critical interdependencies, since such interdependencies correspond by definition to cases of observed communication between the teams. Using this criterion, the teams attended 924 out of 1,156 technical dependencies between their components (80 percent). Yet, there was substantial variability across teams in their ability to attend to their technical interdependencies, and this variability is the focus of our paper. Figure 2 provides detailed counts of the technical interdependencies (as identified by the systems architects) and uncovered technical interdependencies (as identified through team communication), classified by criticality and attention status.

We assessed the validity of our measure of the dependent variable in two complementary ways. First, qualitative evidence gathered during our data collection suggests that the failure to attend critical interdependencies during the detailed design phase had important consequences for the project. For example, some of the unattended critical dependencies caused reductions in performance or durability of the affected components and sub-systems. Some of these problems (if not found and corrected in later phases of the product development project) would cause significant warranty or service expenses over the 25-30 year life cycle of the engine. For example, an unattended critical interface caused excessive loads on assembled hardware in early development tests, resulting in severely distressed hardware and special disassembly procedures. Another unattended critical interface caused excessive loads to—and reduced the life of—a critical engine component, but was not discovered until the first engines entered service (Sosa et al. 2004: 1687). Second, we tested the robustness of our results with alternative measures of the dependent variable. Specifically, we considered alternative interface redesign thresholds set at 0% and 20% respectively, hence varying the number of design interdependencies considered critical. We also assumed that the communication between teams in the absence of a previously identified design dependence was spurious or redundant, hence eliminating the 46 dependencies uncovered by team interactions from the measure of the dependent variable. Our statistical results were robust to these alternative definitions.

Observed team communication among 54 design teams?				
	NO (1,149)	**232 Unattended critical design dependencies**	64 Unattended non-critical design dependencies	1,032 Lack of interdependencies
	YES (282)	**878 Attended critical design dependencies**	52 Attended non-critical design dependencies	**46 Uncovered critical interdependencies**
		Critical Interfaces (322)	Non-Critical Interfaces (31)	
		YES (353)		NO (1,078)
		Design dependencies identified by systems architects?		

Figure 2. Classification of technical interdependencies

Independent variables

Component modularity. Engine components "lose" degrees of freedom (or modularity) to the extent to which they have direct and indirect technical interdependencies (i.e., design interfaces) with other components in the engine. We argue that there are two ways by which component *i* may lose modularity by interacting with component *j*. First, component *i* loses modularity by directly interacting with component *j*. The stronger the technical interdependence of component *i* with component *j*, m_{ij}, the fewer degrees of freedom for modifying the design of this component and the more it will be affected by changes that might occur in component *j*. Second, component *i* may lose modularity if it interacts with another component *q* that also interacts with component *j*. That is, component *i* is additionally constrained by indirect technical interdependencies with component *j* through component *q*, as changes in *j* can affect *i* directly ($m_{ij} > 0$) and indirectly through its effects on component *q* ($m_{iq}m_{qj} > 0$). These two mechanisms are captured by the following expression measuring component *i*'s loss of modularity (*lm*) due to its direct and indirect technical interdependencies with *j:*

$$(lm)_{ij} = \left[m_{ij} + \sum_q m_{iq} m_{qj} \right]^2 \ ; i \neq j \neq q$$

where m_{ij} corresponds to the standardized measure of identified technical interdependencies between components *i* and *j* discussed before.

It is worth recalling that m_{ij} captures the marginal strength of the technical interdependencies between components *i* and *j* with respect to the strongest technical interdependence within the product. By squaring the previous expression, our measure assumes that modularity is lost exponentially as direct and indirect technical interdependencies become stronger, and that there is an additional loss caused by the joint occurrence of direct and indirect technical interdependencies. The total loss of modularity for component *i* is simply the sum of the losses caused by each of its direct and indirect interfaces across all components *j* for which $m_{ij} > 0$, $\sum_j lm_{ij}$. Thus, component modularity diminishes with the number and with the strength of the direct and indirect technical interdependencies of that component. We reverse-coded the scores by multiplying each figure by negative 1 and dividing by 100 to facilitate the interpretation of the statistical analysis results. This is our measure of component modularity. See appendix A for a graphical illustration of the variation in component modularity captured by our measure.

Team's network density. We measure the density of team *i*'s network as the number of actual interactions among the teams interacting with team *i* over the number of potential interactions that could occur among such teams (excluding the direct interactions with team *i*). We use the general purpose network analysis software UCINET to calculate teams' network density (Borgatti et al. 2002).

Control variables

Team size. The effects of component modularity on a team's ability to attend to interdependencies and the team's ability to engage in communication with other teams might be affected by the resources available to the team. Because managers are likely to allocate resources to teams taking into account the workload that results from characteristics of the component and its interfaces with the other engine components, the effects of modularity and communication network structures may be confounded with the effects of the resources available to the team. To control for this possibility, we include a four-point discrete variable accounting for the manpower resources allocated to teams. Although we were unable to collect precise data on team size—which was also variable throughout different stages of the process—we obtained a qualitative assessment of team size based on the experience of one of the authors in the project, who is a design expert with substantial experience in similar engine programs and who also reviewed the design work for this particular project.

Team's network size. We control for the number of communication ties for each team for two reasons. First, the size of a team's communication network may directly affect the team's ability to attend to its critical technical interdependencies. The larger the number of communication ties a focal team has, the more likely it will be to communicate with interdependent teams, hence increasing the likelihood of attending to interdependencies. Second, network size is typically correlated with network density. The larger the team's network size, the lower the likelihood that any two teams in that network will communicate with one another and the lower the density of that network. By holding network size constant, we ensure that any observed effect for network density corresponds to the structure of the network and that is not an artifact of having a large number of direct contacts. Our measure of network size for team i is simply the number of direct communication ties between this team and all other teams in the project. In network analytical terms, this measure corresponded to the "degree" of the team's communication network (Freeman 1979).

Communication through integration teams. Although the two communication network variables—team network density and network size—are measured on the communication network for the 60 teams in the project (that is, the 54 design teams and the 6 integration teams), design teams may differ in the extent to which they communicate indirectly through integration teams. Indeed, the role of such teams is to collect and pass technical information among design teams to determine engine-level performance in areas such as rotor dynamics and aerodynamics. In some cases, indirect communication through integrating teams may allow two design teams to attend to critical interdependencies without engaging in direct communication. If there is heterogeneity in the team's reliance on indirect communication to attend to technical interdependencies, and if this heterogeneity is related to any of our other independent variables, estimates for the effect of these variables would not be accurate. To eliminate this possibility, for each team *i* we control for the number of teams *j* that team *i* reaches through one or more integration teams in two steps in the communication network. Specifically, a two-step communication tie between design teams *i* and *j* occurs whenever team *i* communicates with integration team *k* and this team in turn communicates with team *j,* but there is no direct communication tie between *i* and *j*.

Component redesign. An important factor that impacts both a team's workload and the need to coordinate interfaces with other engine components is the novelty of its component. The more novel the component with respect to a prior generation, the more likely it will generate extra work for the team and the more likely it will affect the interfaces with adjacent components. We capture this by measuring component redesign. We define component redesign as the percentage of actual novel design content in a component, relative to the design of this component in the previous version of the product. As it is physically impossible to determine the exact amount of redesign in a component, we relied on percentage estimates by the design teams of the amount of redesign for their respective components in comparison to the prior existing engine. Specifically, we asked team informants to provide an estimate of the level of redesign required for "your parts or system for the PW4098, as a percentage of the prior existing engine design." This question provided a clear reference point to capture planned redesign due to new design content of the component itself as well as redesign to adapt the component to the new engine.

Component complexity. Another component attribute that may impact the team's workload and therefore the capacity to attend technical interdependencies with other design teams is the internal complexity of the component itself. We measure the complexity of components by estimating the number of distinct parts included on each component. Here, we rely on the experience of one of the authors in this project as well as other similar engine programs.

Non-critical design dependencies. Although non-critical dependencies were explicitly excluded from our measure of the dependent variable, a team's workload might still be affected by the number of non-critical design dependencies facing a team. In total, systems architects identified 276 non-critical design dependencies throughout the 353 design interfaces among the 54 engine components. Other things being equal, one can expect that the larger the number of non-critical design dependencies facing a team, the higher its workload and the more likely the team may fail to attend to some of its critical interdependences. To control for this possibility, we include a variable that captures the number of non-critical design interdependences for each team.

Model estimation

Two characteristics of our data impose specific constraints on model estimation. First, because our dependent variable is a proportion such as $0 \leq y \leq 1$, standard ordinary least squares (OLS) estimates may not be appropriate, for two reasons. First, the predicted values from an OLS regression can never be guaranteed to fall within the unit interval. Second, the coefficient of a linear model assumes that the effect of x is constant across all levels of y, which again may not be accurate. A standard solution to this problem is to estimate a linear model for the log-odds ratio of the dependent variable. This approach, however, cannot be applied to observations where y_i takes on values of 0 or 1, which forces the introduction of an adjustment before computing the log-odds ratio. A better alternative, proposed by Papke and Wooldridge (1996), is to use the following model: $E(y_i \mid x_i) = G(\beta x_i)$ where $G(.)$ is the logit function. This approach does not require any adjustment of the data for extreme values of 0 and 1: The conditional expectations of y given x is estimated directly using non-linear least squares (see Papke and Wooldridge 1996 for details). To estimate this model, we rely on the procedure implemented in *Stata-SE 9,* estimating a generalized linear model (GLM) with a binomial distribution of the dependent variable, a logit function and robust standard errors.

A second characteristic of the data is that the components of the PW4098 engine were architected into eight systems, with their corresponding design teams organized in a similar way. Typically, interfaces within systems were significantly stronger than interfaces across systems, and the same occurred with communication ties between teams (Sosa et al. 2004). This suggests that observations within a given system may not be independent. To account for this condition, we estimate the model adjusting standard errors for intra-group correlation using the cluster procedure implemented in *Stata*.

4. Analysis and Results

Table 1 presents descriptive statistics of the variables in the analysis. As expected, team size is positively correlated with the team's ability to attend to technical interdependencies. Team size is significantly and negatively correlated to component modularity (r = -.390, p < .01) and positively correlated with both the number of non-critical design dependencies and component complexity (as measured by the number of distinct parts). The simple correlation with redesign, however, is not significant. Yet, when the effects of modularity, number of non-critical dependencies, and component complexity are held constant, redesign has the expected positive association with team size (1.99 *t*-value), with these four variables explaining over 20% of the variance in team size. This confirms the expectation that managers did consider both component attributes when assigning resources to teams and provides legitimacy for our qualitative estimate of team resources. Team size is also positively associated with communication network size, suggesting that the ability to sustain a communication network increased with the resources available to the team. Team network size is negatively correlated with team network density, confirming the well-known relationship between these two network variables. Finally, a team in charge of more modular components was more likely to have a smaller communication network, which is consistent with the fact that such a network resulted from technical interactions due to interdependencies. Because teams in charge of modular components had fewer interdependencies to attend, they did not need to maintain large communication networks.

Table 2 presents GLM estimates of the effects of the independent variables on the team's ability to attend to its critical technical interdependencies. Models 1 to 4 introduce the control variables; Models 5-6 enter the two key independent variables, modularity and network density, whereas Model 7 introduces the interaction between these two variables. Following the standard practice, we centered the predictors on their mean to facilitate the interpretation of the interaction results and to mitigate the effects of multicollinearity between the main variable and the interaction term. Since our data is clustered in eight groups we cannot use likelihood ratio test to make statistical inference of the coefficients because it does not fully account for the fact that individual observations are no longer independent. In these cases, we use the "likelihood" only for the computation of the point estimates and not for variance estimation. Instead, we use Wald tests and Bonferroni tests to assess the significance of the coefficients (Korn and Graubard, 1990, Williams 2000). Because our sample size is limited to 54 cases, the inclusion of nine independent variables in Model 7 may raise concerns about the small number of observations per estimate. While the display of hierarchical models attests to the stability of the main results across model specifications, we took an additional precaution and estimated a model where all non-significant controls were excluded from the equation (Model 8). The estimates for the variables in this model are essentially similar to the ones reported in Model 7, attesting to the stability of the results.

Table 1. Descriptive statistics and correlations of variables†

Variables	Mean	S.D.	1	2	3	4	5	6	7	8
1. Fraction of attended critical interdependencies	0.79	0.15								
2. Team size	2.44	0.93	0.265**							
3. Comm through integration teams	37.78	8.28	-0.236*	-0.201						
4. Non-critical dependencies	10.22	7.68	-0.179	0.352**	-0.064					
5. Component complexity	41.59	66.19	-0.048	0.308**	-0.160	0.288**				
6. Component redesign	0.48	0.33	0.194	0.142	0.081	0.028	-0.334**			
7. Component modularity	-0.83	1.05	-0.078	-0.390***	0.445***	-0.418***	-0.583***	0.233*		
8. Team network size	13.17	7.13	0.506***	0.549***	-0.550***	0.239*	0.388***	-0.085	-0.636***	
9. Team network density	0.64	0.17	-0.337**	-0.612***	0.338**	-0.373***	-0.245*	-0.129	0.363***	-0.793***

**p* < .1; ** *p* < .05; ****p* < .01

†Team degree and team network density are calculated on the communication network of the 60 teams

Team size has a significant positive effect on the team's ability to attend to its critical technical interdependencies in Models 1, 2, and 3, but it becomes non-significant once communication network size is included in the equation (Model 4). This suggests that the effects of team size on the ability to attend critical interdependencies operates mainly through the positive effect of a team's size on its ability to maintain a sufficiently large communication network through which to address interdependencies.

The effect of indirect communication through the integrators teams is not significant. This, however, cannot be interpreted as an indication that integration teams did not play an important role in the project. Rather, it seems that there was little variance in the number of teams reached through integrators in our sample and that variance was mostly associated with component modularity. Indeed, an inspection of the correlations in Table 1 suggests that the tendency to rely on indirect communication through integrators was not associated with any of the workload characteristics of the team except for component modularity. We also tested the robustness of this interpretation by computing measures of communication network size and density after excluding the integrator teams from the network and re-estimating the models. The results were analogous to the ones presented in Table 2.

Table 2. Effects on attendance to technical interdependence†

Variables	Model 1	Model 2	Model 3	Model 4	Model 5	Model 6	Model 7	Model 8
Team size	0.248*** (0.091)	0.341** (0.155)	0.293* (0.177)	-0.038 (0.129)	-0.076 (0.117)	-0.034 (0.103)	-0.045 (0.106)	
Comm through integration teams		-0.023 (0.019)	-0.025 (0.021)	0.012* (0.006)	0.002 (0.009)	0.007 (0.104)	0.008 (0.010)	
Non-critical dependencies		-0.033** (0.014)	-0.034*** (0.002)	-0.041*** (0.007)	-0.033*** (0.009)	-0.019** (0.009)	-0.017** (0.008)	-0.017** (0.008)
Component complexity		-0.0014* (0.0008)	-0.0005 (0.0016)	-0.0023** (0.0010)	-0.0004 (0.0010)	0.0007) (0.0007)	0.0005 (0.0007)	
Component redesign			0.441 (0.420)	0.543 (0.406)	0.480 (0.409)	0.621* (0.326)	0.600* (0.327)	0.546** (0.301)
Team network size				0.121*** (0.029)	0.158*** (0.027)	0.230*** (0.031)	0.241*** (0.032)	0.228*** (0.024)
Component modularity					0.429*** (0.156)	0.608*** (0.158)	0.585*** (0.156)	0.568*** (0.145)
Team network density						2.780*** (0.354)	3.268*** (0.388)	3.137*** (0.429)
Modularity x Density							-0.772** (0.346)	-0.718*** (0.248)
Deviance	7.568	6.667	6.513	4.510	4.148	3.710	3.661	3.700
N	54	54	54	54	54	54	54	54

* $p < 0.1$ ** $p < 0.05$ *** $p < 0.01$ (*two-tailed tests*)

† Coefficients significance estimated with robust standard errors adjusted for eight clusters in the system.

As expected, the effect of non-critical dependencies is negative in the partial models, as these dependencies are expected to create additional workload for the team. The positive effect of redesign in the full models (6 and 7) is less intuitive. One would have expected that, insofar as redesign is likely to increase workload, it should strain the team resources and contribute to a higher failure in attention to critical interdependencies. The opposite seems to be true. A possible explanation for this effect is that high redesign forces the teams to focus the attention on their shared interfaces, making it more likely that they will establish a communication tie through which attend their critical interdependencies. Finally, team network size has the expected positive effect on team's ability to attend critical interdependencies. The larger the team's communication network, the larger the fraction of interdependencies attended by this team. Although this effect is intuitive, it is also partly driven by our measure of attention to interdependencies. Indeed, a design team that communicates with all the other 53 design teams in the project would, by definition, attend a 100% of its critical interdependencies.

Models 5 to 7 report the effects that are directly relevant to testing our hypotheses. Consistent with *H1*, component modularity has a significant positive effect on the proportion of critical interdependencies attended by teams. As expected, teams in charge of modular components were more likely to attend a larger proportion of their technical interdependencies. The effect of network structure was also positive and significant as predicted by *H2*. Holding workload and network size constant, teams embedded in a densely connected communication network were significantly more likely to attend a larger proportion of their critical technical interdependencies than teams that communicated with sparsely connected others.

Consistent with the network and organizational literature, we attribute the positive effect of communication network density on team's ability to attend to critical interdependencies to the collaborative environment created by densely connected communication networks. We also argued (*H3*) that if this mechanism is true, the effects of network density should be more apparent for teams whose components shared more direct and indirect interfaces with other components in the product—that is, for teams in charge of less modular components—and less so for teams in charge of relatively independent (i.e., modular) components. The significant negative coefficient for the interaction between component modularity and network density (Model 7) supports *H3*. It suggests that the benefits of network density are contingent on the degree of modularity of the team's component. As modularity increases, the likelihood of unanticipated complications in the design process is likely to decrease and the benefits of being embedded in a densely connected communication network are less apparent. It should be noted, however, that the net effect of communication network density remains positive at all levels of component modularity, although they are stronger for teams with low modularity components.

5. Discussion

The main question we address in this paper is: *What makes some teams better than others on attending to their critical technical interdependencies?* This is particularly relevant in complex product development, where one main challenge is coordinating and integrating the effort of many design teams developing the many sub-systems and components that form the product.

We found that both the modularity of the components and the density of the communication network of the design teams significantly influence the ability of such teams to attend larger proportion of their critical technical interdependencies. Our data clearly suggest that component modularity helps: components with less direct and indirect interfaces with other components in the product are less likely to generate unpredicted design iterations. This, in turn, allows design teams to use their assigned resources to attend larger proportions of their critical interdependencies. This result has important implications for other design decisions that depend (or should depend) upon structural properties of the components, as determined by their connectivity within the product. For example, the decision of whether to outsource or not a component should consider its modularity, because critical interfaces of less modular components have a high probability of going unattended than those of more modular components.

Our results also highlight the importance of the structure of the informal communication network—and specifically, a densely connected network– in explaining the team's ability to attend to critical technical interdependencies. The effect of network density is independent of the intuitive "coverage" effect of having a large communication network. Being directly connected to many other design teams obviously help a team to attend to its technical interdependencies, but communication among those design teams foster a climate of trust and mutual collaboration that helps the focal team to process information and solve problem faster, as well as to gain flexible capacity to manage the variability inherent to the design process. In other words, a dense communication network gives a boost to the resources of the team, allowing it to attend a larger proportion of its critical interdependencies. Consistent with this reasoning on the mechanisms that make communication network density beneficial for teams in attending interdependencies, the effects of network density are stronger for teams likely to experience higher capacity constraints, as it is the likely case for teams designing less modular components.

The observed effects of network density on the teams' ability to attend technical interdependencies add to the growing literature on the effects of network structures on a variety of outcomes. Sparse communication networks rich in what Burt (1992) dubbed "structural holes" between the parties connected to a focal player can help this player to access diverse information and to facilitate the circulation of that information among otherwise disconnected players (Burt 1992). Sparse networks, however, may become a liability in contexts where the active collaboration among the actors is essential, as it is the case with the setting of this study (Coleman 1990; Burt 2005:93-166; Ahuja 2000; Obstfeld 2005; Oh, Chung, and Labianca 2004, Reagans and McEvily, 2003). In our study, it was essential to disentangle the effects of network size and network density in order to uncover the positive effect of network density. If we had measured communication network structures using an aggregated measure of network constraint (as defined by Burt 1992), we would find that teams with sparse networks (spanning structural holes) would attend larger proportions of critical interdependencies. Such a result, however, is driven by the effect of network size while the effect of network cohesion (captured by indirect network constraint) is significantly positive.

The research stressing the positive effects of network density on collaboration (and hence on the outcomes facilitated by that collaboration) has also warned about the pitfalls of extremely dense networks. Such pitfalls are associated with the risk of having redundant information (Burt 1992) and excessive rigidity (Burt 2005:167-223; Gargiulo and Benassi 2000) which can fuel inertia and hinder innovation. In contexts where the benefits of density should be apparent, extreme levels of network density may result in costs that offset at least in part the alleged benefits. This possibility has led scholars to propose a curvilinear association between network density and benefits (e.g., Uzzi 1997; Oh, Chung, and Labianca 2004). We tested for the possibility of a curvilinear association between network density and the teams' ability to attend to their interdependences by using a quadratic specification for density. As expected, the coefficient for the squared term is negative, but not significant (-1.07 *t*-value), failing to provide clear support for this conjecture.

The effects of network density on the teams' ability to attend technical interdependencies also have implications for important design decisions associated with product components. First, it is important for managers to identify critical interdependencies in advance and to provide the coordination mechanisms and incentives to foster the formation of dense networks around such critical interfaces, especially for teams whose components are more tightly integrated within the product architecture. Second, and going back to the outsourcing decision, managers need to consider that outsourcing less modular components may require significant investments to foster communication networks among the groups in charge of those components, which may weight negatively on the decision to outsource them.

Previous research in product development typically considers collaboration and coordination as the result of direct interaction between interdependent actors (e.g. Allen 1977, Brown and Eisenhardt 1995, Krishnan and Ulrich 2001). Yet, we have shown that indirect connections both in the product and organizational domains matter significantly in distinct ways. Indirect design interfaces decrease the modularity of the product components and therefore increase the risk of unforeseeable design iterations which in turn increases the likelihood of failing to attend critical interdependencies. On the other hand, indirect communication through common third parties facilitates trust, knowledge sharing, and collaboration, providing teams with additional capacity that makes it easier for them to handle larger proportions of critical interdependencies.

Although fieldwork evidence from the project studied validates our quantitative results (see appendix C for details), the nature of our data suggests care at the time of generalizing our findings. We believe that similar results would be found in other complex design projects where the product structure is well understood and whose development organization maps directly to the architecture of the product. Nonetheless, external validation of our findings is needed, which opens interesting opportunities for future research in this area.

6. Conclusions and Future Work

As Simon (1981) suggested, complex systems are difficult to understand because the behavior of the whole depends in non-trivial ways on how its elements interact. Similarly, understanding how technical information flows in social networks within organizations is an elusive task that requires managers' active intervention. A network approach focused on capturing the elements of both product and organizational systems and their linkages can be very fruitful in understanding how these complex systems work. We have applied a network approach to study the interactions across the network structure of a complex product and the complex organization that designed it. We examine the impact of product and organizational network structures on an outcome that is critical for complex product development: attention to critical interdependencies. This has important implications for both operations management and organizational theory. For operations management, we show how network attributes of product components such as modularity can be measured so that the impact of components' connectivity can be studied. In addition, we also learn that although component modularity can help teams attend to a larger proportion of their critical interdependencies, the benefits of modularity reach a limit that can be surpassed only by examining the organizational structure. For organizational theory, we show that it is important to understand not only which network structure provides more benefits to design teams but also what determinants of design tasks can moderate the impact of social networks. We found that the benefits of dense networks are mitigated by the modularity of product components.

While our paper is successful in addressing the questions posed in our introduction, it also raises a number of other interesting questions that can stimulate future research on the topic. How does the communication network structure of teams mitigate design iterations (or design oscillations) typically observed in complex product development? How does component modularity and team network structure relate to important design decision such as outsourcing or off-shoring? Further research is also required to understand the dynamics of these complex systems. How do product structures evolve over time? Which organizational network structures are better to cope with such evolution? By fruitfully combining insights from operations management and network analysis, we believe that we have delineated a path that can help researchers address these important questions. Appendices are included in an online supplement.

7. References

Adler, P., A. Mandelbaum, V. Nguyen, E. Schwerer. 1995. From project to process management: An empirically based framework for analyzing product development time. *Mgmt. Sci*. 41(3): 458–484.

Ahuja, G. 2000. Collaboration networks, structural holes, and innovation: A longitudinal study. *Admin. Sci. Quaterly*. 45: 425–455.

Allen, T.J. 1966. Studies of the problem-solving process in engineering design. *IEEE Trans. Engrg. Management*. EM-13(2) 72-83.

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

Bae, J., and M. Gargiulo. 2004. Partner substitutability, alliance network structure, and firm profitability in the telecommunications industry. *Academy of Management Journal* 47: 843–859.

Baldwin, C.Y., and K.B. Clark. 2000. *Design rules: Volume 1: The power of modularity*. Cambridge, Mass.: MIT Press.

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

Borgatti, S.P., Everett, M.G. and Freeman, L.C., 2002, *UCINET for Windows: Software for Social Network Analysis*. Harvard: Analytic Technologies , Cambridge, MA.

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

—, 2005. Brokerage and Closure. An Introduction to Social Capital. Oxford University Press.

Clark, K.B., and T. Fujimoto. 1991. Product Development Performance: Strategy, Organization and Management in the World Auto Industry. Cambridge, Mass.: Harvard Business School Press.

Clarkson, P.J., C.S. Simons, and C.M. Eckert. 2004, Predicting change propagation in complex design, *Journal of Mechanical Design*, **126**(5):765-797.

Coleman, J.S. 1990, *Foundations of social theory*. Cambridge, Mass.: Belknap Press.

Eppinger, S. D., D.E. Whitney, R.P. Smith, D.A. Gebala. 1994. A model-based method for organizing tasks in product development. *Research in Engineering Design* 6(1): 1–13.

—. 2001. Innovation at the speed of information. *Harvard Business Review*. 3-11.

Ethiraj, S.K., and D. Levinthal. 2004. Modularity and innovation in complex systems. *Mgmt. Sci*. 50(2), 159–173.

Freeman, L. 1979. Centrality in social networks: Conceptual clarification. *Social Networks* 1:215–239.

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

Gargiulo, M. 1993. Two-step leverage: Managing constraint in organizational politics. *Administrative Science Quarterly* 39: 1-19.

—, and M. Benassi. 2000. Trapped in your own net? Network cohesion, structural holes, and the adaptation of social capital. *Org. Sci.* 11(2): 183–196.

Granovetter, M.S. 1985. Economic action and social structure: The problem of embeddedness. *American Journal of Sociology* 91: 481–510.

Gumbel, P. 2006. Trying to untangle wires. *Time*. October 16 (European edition), p. 36-37.

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.

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

Hollinger, P., G. Wiesmann. 2006. House of conflict: How rivalries have disrupted the steep climb of Airbus. *Financial Times*, November 14, p. 13.

Korn, K.L. and B.I. Graubard.1990. Simultaneous testing of regression coefficients with complex survey data: use of Bonferroni t statistics. *The American Statistician*, 44: 270-276.

Loch, C.H., and C. Terwiesch. 1999. Accelerating the process of engineering change orders: Capacity and congestion effects. *J. Prod. Innov. Mgmt*. 16: 145-159.

—, —, and S. Thomke. 2001. Parallel and sequential testing of design alternatives. *Mgmt. Sci*. 45(5): 663–678.

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

Mihm, J., C. Loch, 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. Management* 42(3): 215-222.

Obstfeld, D. 2005. Social networks, the *Tertius iungens* orientation, and involvement in innovation. *Admin. Sci. Quart.* 50: 100–130.

Ocasio, W.1997. Towards an attention-based view of the firm. *Strat. Mgmt. J*.18:187-206.

Oh, H., M. Chung, and G. Labianca. 2004 Group social capital and group effectiveness: The role of informal socializing ties. *Academy of Mgmt. J*. 47 : 860-875.

Papke, L.E., and J.M. Wooldridge. 1996. Econometrics methods for fractional response variables with an application to 401(k) plan participation rates. *Journal of Applied Econometrics* 11(6): 619–632.

Pich, M., C. Loch, A. De Meyer. 2002. On uncertainty, ambiguity, and complexity in project management. *Manag. Sci*. 48(8): 1008–1023.

Pinedo, M., S. Sehasdri, and E. Zemel. 2000. The Ford-Fireston case. *Teaching case*. Department of Information, Operations, and Managment Sciences, Stern School of Business, NYU.

Raub, W., and J. Weesie. 1990. Reputation and efficiency in social interactions: An example of network effects. *American Journal of Sociology* 96: 626–654.

Reagans, R. and B. McEvily. 2003. Network structure and knowledge transfer: The effects of cohesion and range. *Admin. Sci. Quart.* 48: 240–267.

Rowley, T., D. Behrens, D. Krackhardt. 2000. Redundant governance structures: An analysis of structural and relational embeddedness in the steel and semiconductor industries. *Strat. Mgmt. J*.. 21**:** 369–386.

Sanchez, R., and J.T. Mahoney. 1996. Modularity, flexibility, and knowledge management in product and organization design. *Strategic Management Journal* 17: 63–76.

Schilling, M. 2000. Toward a general modular systems theory and its application to interfirm product modularity. *Academy of Management Review* 25(2): 312–334.

Simon, H.A. 1947. *Administrative Behavior: A Study of Decision-making Processes in Administrative Organizations*. Macmillan, Chicago, IL.

—, 1981. *The Science of the Artificial* (2nd ed.). Cambridge, Mass.: MIT Press.

Smith, R.P., and S.D. Eppinger. 1997. Identifying controlling features of engineering design iteration. *Management Science* 43(3): 276–293.

Sosa, M.E., S.D. Eppinger, and C.M. Rowles. 2003. Identifying modular and integrative systems and their impact on design team interactions. *Journal of Mechanical Design* 125(2): 240–252.

—. 2004. The misalignment of product and organizational structures in complex product development. *Mgmt. Sci.,* 50(12): 1674–1689.

—. 2006. A network approach to define modularity of components in complex products. *ASME Journal of Mechanical Design (forthcoming).*

Suh, N.P. 2001. *Axiomatic Design: Advances and Application*. New York: Oxford University Press.

Terwiesch, C., C. H. Loch, 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.

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

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

—, S.D. Eppinger. 2004. *Product Design and Development* (3rd ed.). New York: McGraw-Hill.

Uzzi, B. 1997. Social structure and competition in the interfirm networks : The paradox of embededdness. *Admin. Sci. Quat*. 42 : 35-67.

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.

Williams, R.L. 2000. A note on robust variance estimation for cluster-correlated data. *Biometrics* 56.

Online supplement for:

Component Modularity, Team Network Structure, and the Attention to Technical Interdependencies in Complex Product Development

Appendix A: Additional details about the product network data

As mentioned, each type of design dependence x between components i and j was coded as either non-critical ($x_{ij,type}$ = 1) or critical ($x_{ij,type}$ = 2), depending on the impact on functionality of component i due to its dependence from component j. In both cases we capture both positive and negative design dependencies. Non-critical dependencies were those that were either "beneficial, but not absolutely necessary for component functionality" (a positive dependency) or "cause negative effects, but do not prevent component functionality" (a negative dependency). Critical dependencies, on the other hand, were those that were either "necessary for component functionality" (a positive dependency) or "must be prevented to achieve component functionality" (a negative dependency). For example, we determined that the outer air seals and transition ducts (OAS-Duct) of the low-pressure turbine (LPT) impose a critical, one-directional, negative, energy dependence on the LPT blades, driven by geometry and clearances between the components, which make it difficult for the blades to maintain adequate vibration margin (See Sosa et al. 2003 and Sosa et al. 2006 for further technical details).

Figure A1 shows a cross-sectional view of the engine studied as well as the network diagram of the 54 components of the engine. The nodes are colored to illustrate the eight systems that comprised the engine: Fan, Low-Pressure Compressor (LPC), High-Pressure Compressor (HPC), Combustion Chamber (CC), High-Pressure Turbine (HPT), Low-Pressure Turbine (LPT), Mechanical Components (MC) and External and Controls (EC).

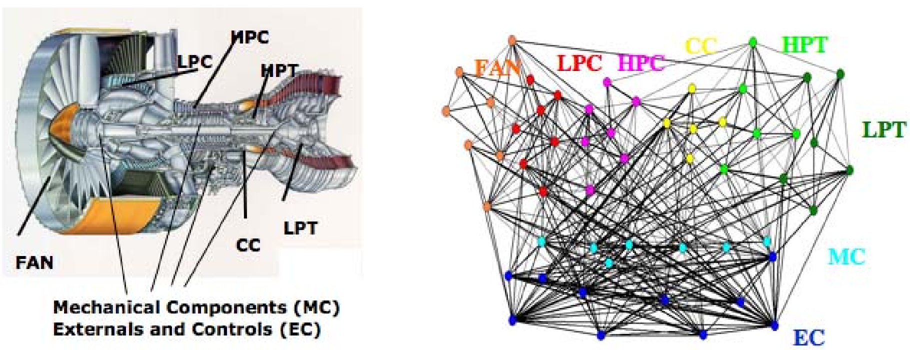

Figure A1. Cross-sectional view and network diagram of the PW4098 aircraft engine

To illustrate the variation in component modularity captured by our measure, Figure A2 shows the ego network of the most and least modular components of the engine studied.

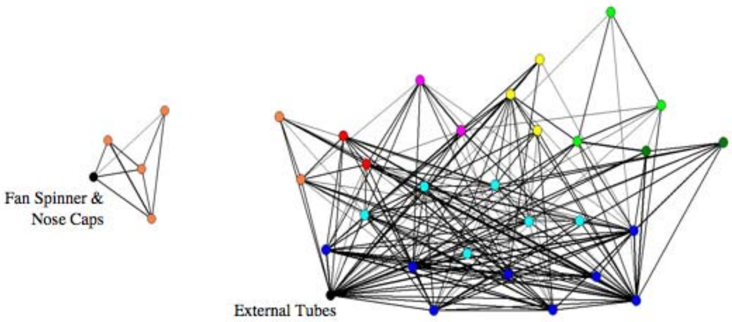

Figure A2. Network structure for two components with the lowest and highest modularity

Appendix B: Additional details about the communication network data

In order to collect the communication data we used a single discrete metric to capture both *criticality* and *frequency* of the *peak* interaction exchange during the detailed design period. The criticality dimension ranges from *routine* to *critical* (*routine*: the information requested was important but could be generated by the team alone with minimal risk of delay; *important*: the information requested could be generated by the team alone with some risk and effort; *critical*: the information could not be generated by the team alone) while the *frequency* dimension ranges from *never* to *very frequent* (*never*: no information requested during detailed design phase; *infrequent*: monthly or less; *frequent*: weekly or biweekly; *very frequent*: daily or almost daily). Hence, non-zero interaction intensity ranges from "routine and frequent" to "critical and very frequent." Our scale excluded interactions that were routine *and* infrequent because they were likely to be spurious or redundant information exchanges. We conducted follow-up interviews with at least one team member to validate our communication data and excluded few interactions that were erroneously reported as they were purely social or consultative in nature rather than project-related interactions. In total, we identified 680 *significant* technical interactions in which team *i* requested technical information from team *j* during the detailed design phase of the development project.

Appendix C: Additional qualitative evidence from the project studied

We also gathered additional fieldwork evidence from the project studied to validate our quantitative results. Teams designing less-modular components had significantly more cross-component and cross-system interfaces to attend to, but had less engineering work to do per component than the teams designing more modular but more engineering-intensive components, with correspondingly less time to manage their interfaces outside of their systems. Given this, each type of team had to develop tools that made their work more efficient within the time allotted. Some teams developed communication-enabling tools that allowed them to establish network cohesion in large networks, while other teams developed more automated engineering analytical tools that could be used within a small dense network with other teams. The design of some components, such as the fan hub or the high-pressure compressor [HPC] fixed stators, whose lead times were limited by engineering analysis and by part definition leading to long production cycles (e.g., major castings and custom forgings of super-alloys) were managed the “old way”—that is, with less frequent communication but sufficient and efficient enough to provide them the information they needed when they needed it. For these teams, component modularity and network density clearly impacted their attention to critical interdependencies. For example, although both the fan hubs and the HPC fixed stators had proportional team resources to handle their levels of component redesign and component complexity, both the component modularity and the team network density of the latter was significantly lower than the former, resulting in significantly less proportion of attention to critical design dependencies.

Europe Campus
Boulevard de Constance,
77305 Fontainebleau Cedex, France
Tel: +33 (0)1 6072 40 00
Fax: +33 (0)1 60 74 00/01

Asia Campus
1 Ayer Rajah Avenue, Singapore 138676
Tel: +65 67 99 53 88
Fax: +65 67 99 53 99

www.insead.edu