The Impact of Communication Structure on New Product Development Outcomes

Marcelo Cataldo
Carnegie Mellon University
Pittsburgh, PA, USA
mcataldo@cs.cmu.edu

Kate Ehrlich
IBM Research
Cambridge, MA, USA
katee@us.ibm.com

ABSTRACT
New product development teams face important challenges to their performance due to the novelty of the work and the need to rapidly develop shared knowledge and goals. However, little is known about the relationship between the structural properties of communication and performance in these teams. This study examined the effect of communication structure, specifically hierarchical and small-world structures on the delivery performance and quality outcomes of a large-scale new product development project. Our longitudinal analyses revealed that structuring communication patterns in a hierarchical manner significantly improves delivery performance. However, hierarchical communication has a detrimental effect on quality while small-world communication structures improved the quality outcomes of development teams. We discuss the implications of these results for collaborative tools that support communication tradeoffs.

Author Keywords
Communication; New Product Development; Hierarchy; Small-World Networks; Social Network Analysis.

ACM Classification Keywords
H.5.3 [Group and Organization Interfaces]: Asynchronous Communication, Computer-supported Cooperative Work;

General Terms
Human Factors, Management, Measurement, Performance.

INTRODUCTION
Informal communication is a topic of long standing for project performance in product development teams [1, 6, 22]. As projects become geographically dispersed [9, 31, 32], communication as well as coordination and awareness can be impeded by the spatial and temporal barriers [17, 34], adversely impacting project productivity and quality outcomes [32, 35].

New product development (NPD) teams face additional challenges to their performance as a result of the novelty of the work, and new designs and processes [30]. Moreover, the need to ensure that the innovations are successful can result in different patterns of communication than for more routine or well-understood tasks [16, 40]. Research has established that team performance is associated with communication structure [27]. However, these studies have not addressed NPD issues and have been mostly cross-sectional. Further, ample evidence exists that interaction patterns evolve over the life cycle of a project (e.g. [1, 10, 16]).

The current study seeks to extend existing research on the effect of informal communication on performance, to new product development teams. We examined the effect of communication structure – specifically hierarchical and small-world structures – on: (a) planning and its execution and (b) the quality of the outcomes, in NPD projects. We depart from previous studies using surveys or retrospective reports, by relying on analysis of communication traces from to tasks performed in the project, collected as the work was being done in order to examine communication over time rather than at a particular snapshot.

Our work makes two valuable contributions to the new product development and CHI/CSCW literatures. Firstly, by examining the impact of communication structure on two distinct aspects of new product development performance, planning and quality, we begin to address the question of whether it is sufficient for a team to evolve a single communication structure or whether teams need to become more adept at trading off between communication styles. Secondly, we shed light on how the structure of informal communication evolves over the life cycle of new product development projects and how those patterns impact those projects, particularly geographically distributed ones.

The rest of the document is organized as follows. We first discuss the theoretical background leading to our hypotheses. Then, we describe the research setting and our empirical design followed by a discussion of the results. We conclude with a discussion of limitations of our study and the implications of our results for future research and tools.
INFORMAL COMMUNICATION AND OUTCOMES IN NEW PRODUCT DEVELOPMENT

In product development, project success or performance is often measured by outcomes such as product quality, time, cost and customer-oriented metrics [39]. When development projects focus on one outcome, e.g. high product quality, they face the need to make trade-offs in project length or functionality in order to deliver on time [31, 39]. A key question for researchers and practitioners alike is whether it is in fact necessary for new product development (NPD) projects to trade one outcome for others, for instance, can projects achieve high levels of quality without impacting or minimally impacting product delivery plans? Research results are ambivalent in this regard. On the one hand, recent research showed that reducing defects and rework improves quality [24] suggesting that both quality and time could be optimized together. On the other hand, studies of geographically distributed software development projects has shown that team configurations that produce above average quality outcomes are different from those team configurations that exhibit high levels of productivity [13, 35].

One critical driver of new product development outcomes, is informal communication [6]. Most early research focused on the general aspects of communication within a project team such as adequate interaction among the key members or departments involved in the project and between the projects and external stakeholders [2, 6, 28, 38]. More recent work showed that the general patterns of interaction resemble the structure of the product being developed [37]. However, limited attention has been given to the relationship between the specific structural properties of the communication networks and different outcomes of NPD projects. Work in the areas of geographically distributed project teams and innovation suggests two particular communication structures that have been linked to project success: hierarchies and small-world networks [27, 40].

Hierarchical communication networks emphasize efficiency of information. In the context of communication networks, hierarchy considers the extent to which team communication flows outward from one person or a small group without flowing back [29]. Small-world networks are characterized by small densely interconnected groups of individuals which are themselves loosely interconnected through a small number of people. Unlike hierarchical networks, small-world networks emphasize the richness of the information exchange through the dense and redundant paths that link individuals. In this paper, we examine the impact of these two types of communications structures on two distinct outcomes of NPD projects: delivery performance and quality.

The Role of Communication in Delivery Performance

Delivery performance refers to planning development activities and executions of such plans in order to complete the tasks on time, and ultimately, deliver the final product on time. Poor delivery performance may have significant implications in a development organization such as failing to meet customer expectations, impact release of interdependent products and efficiently manage development resources. NPD teams face the challenge of structuring their communication across the various phases of the development process so that team members are well informed and engaged while the amount of communication required does not interfere with finishing the work in time.

Hierarchy

By centralizing communication responsibility to a few individuals, hierarchical structures can reduce the cost of establishing and maintaining communication particularly, in the context of geographically distributed teams [27]. Then, hierarchical communication may be especially well suited to the operational aspects of new product development by enabling efficient distribution of the relevant information to the team members who carry out the tasks.

H1a: NPD teams that have evolved a hierarchical communication structure will exhibit better delivery performance than those teams, which do not have that communication structure

Small-World Communication

On the other hand, small-world communication structures are an important source of shared knowledge, collaborative efforts and innovation [40], factors of particular importance in new product development. However, considering the work involved in carrying out a set of development tasks, having communication clustered in several small, closely knit groups is likely to be detrimental to getting the work communicated, understood by the relevant people and, consequently, completed on time. Firstly, dense communication patterns can increase the amount of time spent on interactions, which can impact the time available for the work. Secondly, it moves communication away from project and technical leads who have responsibility for directing the work. Thirdly, it reduces the consistency of communication and messages to the whole team by reducing the role and voice of the project and technical leads.

H1b: NPD teams that have evolved a small-world communication structure will exhibit poorer delivery performance than teams, which do not have that communication structure

The Role of Communication in Quality Outcomes

The quality of the outcomes is also a key factor in product development teams. Quality relates to the capabilities of the product development team to identify, integrate and bring to bear the right knowledge and information to make the right design or implementation decisions such that defects and mistakes are minimized [12, 19]. Despite the potential benefits of lean and efficient ways of conveying information there are important challenges associated with integrating, and leveraging such information in the context of novel
development tasks [3]. Project members might struggle to make sense of new knowledge and effectively integrate it into the relevant context because of differences in perspectives, lack of common understandings and lack of familiarity working together [5]. Failure to adequately make sense of the knowledge and information can result, for instance, in unidentified dependencies, which lead to coordination breakdowns and ultimately, lower product quality [12, 26].

Hierarchy
Hierarchical communication centralizes communication to a few individuals, which reduces the amount of communication amongst people at the same level of the hierarchy. Lower levels of communication are associated with higher levels of coordination breakdowns [18] which ultimately results in poorer quality [26].

\[ H2a: \text{NPD teams that have evolved a hierarchical communication structure will exhibit poorer quality outcomes than teams, which do not have that communication structure.} \]

Small-World Communication
Small world communication structures may be especially well suited to development tasks, such as the need to coordinate the development of a crosscutting product feature, that are challenging and highly interdependent. The richness of the communication makes it more likely for those people to share similar attitudes and norms and to trust each other and have strong relationships [7]. These commonalities may help reduce team problems due to miscommunication and conflict especially in distributed teams. They also make members more likely to give each other preferential access to important resources and knowledge, consider each other’s information, integrated knowledge, and combine information effectively when making decisions [7, 40].

\[ H2b: \text{NPD teams that have evolved a small-world communication structure will exhibit better quality outcomes than teams, which do not have that communication structure.} \]

RESEARCH SETTING
We examined the proposed hypotheses on data collected from the development of the first release of the Jazz software development environment on which the IBM® Rational® Team Concert™ (RTC) products are based. Jazz is a development environment designed to make product development more collaborative through integration of information and tasks across the phases of the lifecycle [20]. RTC itself was developed using the Jazz environment; IBM made the data from project’s software repositories for the first release available to us.

Development activities started as early as May 2006. However, we focused our analysis on the 13 months prior to the release of the product’s first version, which was the period of time between July 2007 and August of 2008 which corresponded to the main development period. A total of 161 developers, worked full time on the project during the period analyzed. There were 26 functional teams distributed in 21 different sites across U.S., Canada and Europe. Each team planned its own development work for the next iteration during the first week of the iteration and iterations began and ended at the same time for all teams. There were 14 iterations during the time covered by our data. See [19] for additional details.

Description of the Measures
In addition to the data from the software repositories, we also collected information about the project member’s location from personnel records and conducted semi-structured interviews with 4 developers who were members of the project throughout the development, to validate our assumptions about the interpretation of the measures. The rest of the section is organized as follows. First, we describe our outcome measures. Second, we describe how we captured the interaction patterns as well as measures to assess the hierarchical and “small-worldness” nature of those communication networks.

Measuring Delivery Performance
In order to examine the role of the structure of communication patterns, we chose the pair <iteration, team> as the unit of analysis for our study. We measured Delivery Performance of an iteration as the number of tasks that were planned for the <iteration, team> pair but were rescheduled to the following iteration because they were not completed. This measure focuses on how well the team members that participated in the iteration were able to execute the planned set of activities as well as capturing aspects of the development process such as adequate planning and adequate handling of unexpected situations. Interviews confirmed that rescheduling was an adequate measure since it was an integral part of the development process. As one interviewee said, “inevitably things didn’t fit and then we moved them” referring to moving work items to the next iteration.

Measuring Quality
In order to assess the quality outcomes of an iteration, we examined the defects reported in each iteration with the following procedure. For a given iteration \(i\), we first selected all the defects reported in the following iteration \(i_{t+1}\). Our measure of Quality captured the number of defects that either impacted components changed during iteration \(i\) and took 1 or more days to get resolved or impacted components not changed in iteration \(i\) nor in iteration \(i_{t+1}\). We validated this measure by analyzing a random sample of 527 defects corresponding to 10% of all defects across all 14 iterations. We inspected the reports of those defects.
in order to understand if there were indicators that could assist us in the process of linking a defect to the iteration that introduced it. The analysis revealed that many defects consisted of work items that were created and resolved in a matter of hours. The comments in the reports suggested that there were issues found while working on other development tasks and the defects were blocking the other activities so they were resolved quickly but limited information existed to link the defect to prior changes in the source code except for the information about what files were changed.

**Construction of the Communication Network**

The RTC development environment has features that provides project members with contextualized interaction for work items. These interactions took the form of comments in a work item report and were stored in the tool’s database, allowing us to identify the work items, the individuals that made the comment(s) as well as the content of the comment. A stated goal of the project was to make the development as transparent as possible to outsiders as well as internally, which led developers to use comments as the main channel of communication even with collocated colleagues. Moreover, good tracking reinforced developers’ preference to use comments over other forms of interaction.

We constructed iteration-level communication networks as follows. We considered only those comments made in the work item reports that were within the time frame of the iteration. The first comment in the work item linked the person that made the comment with the owner of the task at the time the comment as made. If the owner and the commenter were the same person, the link was ignored. For the subsequent comments in the work item report, a link in the communication network between two project members was made if the comments were adjacent in the sequence of comments and they were associated with the task at hand. These comments also resulted in a link between the commenter and the owner of the work item at the time it was made. Additional details of the construction of the networks can be found in our previous work [19].

**Measuring Hierarchical Communication Patterns**

We computed the Hierarchical Communication Pattern measure based on the hierarchy metric proposed by Krackhardt [29]. Network hierarchy measures the direction of flow in the network. Krackhardt’s approach counts the number of times there is a path from a node in the network to another but not a path back. In order for this measure to be computed, we consider the communication network as a dichotomous directed graph. We used the ORA tool [8] to compute the measure. There are other approaches to measure a central communication structure. A more traditional approach would have been to consider network centralization, which captures the extent to which one or a few individuals are highly connected to the rest of the individuals in the network. Although network centralization and network hierarchy share some similarities, as discussed by Hinds and McGrath [27], the two measures refer to two distinct constructs. Mainly, hierarchical network captures the flow of information. Furthermore, a network in which communication is centralized is not necessarily hierarchical because communication could be reciprocal between the central individuals and those in the periphery. On the other hand, hierarchical networks might not be centralized since information could be delivered through multiple nodes without reciprocal links.

**Measuring Small-World Communication Patterns**

Small world networks are those characterized by short average path lengths among any pair of nodes and high levels of a clustering coefficient [42]. Watts and Strogatz [42] proposed such a definition by comparing a class of networks to Erdos-Renyi random graphs. Then, establishing that a network is a “small world” requires a comparison between the focal network and random graphs. In order to measure the “small worldness” of our communication networks, we build on the approach proposed by Uzzi and Spiro [40] which is based on Watts and Strogatz’s method [42] and utilizes two separate metrics: the clustering coefficient ratio (CC ratio) and path length ratio (APL ratio).

Both ratios are determined by comparing the clustering coefficient and the average path length measures of our communication networks against the corresponding measures computed from random graphs with the same number of nodes and density. We computed 100 random graphs and then computed the average clustering coefficient and path length measures across those 100 graphs. Those averages were used in the comparison to the communication network metrics. The computation of the network metrics was done using the ORA tool. Once the clustering coefficient and average path length ratios were computed, the Small World Communication Pattern variable was calculated as $CC \text{ ratio} = \frac{CC}{\text{APL ratio}}$ as proposed in Uzzi and Spiro [40].

**Additional Control Factors**

Past research work suggests a wide range of factors that may impact outcomes such as performance and quality in software development (e.g. [5, 12, 25]). We organized our control variables into five groups:

**Controls for team-related factors**: We computed several measures that capture attributes of the teams we studied. Since comments on work items associated with a pair (iteration i, team t) could also come from people outside the team we computed two measures: Number of Project Members Involved captures the total of team and non-team members that participated in the work associated with each pair (iteration, team) and Number of Multi-Team Project Members captures the subset of those individuals that were part of more than one formal team.

The project involved engineers in 21 different development locations, so geographical dispersion may play an important role in terms of communication and, consequently, in terms of delivery iteration performance and software quality. GSD
is a dichotomous variable where 1 indicates that the project members involved in the work associated with a pair (iteration, team) were geographically distributed; otherwise it was set to 0.

We collected measures of experience based on the work by Boh and colleagues [5], which utilize the data in software repositories as the basis for assessing experience. We measured Average Level of Technical Experience as the average number of work items that the team member worked on prior to the focal iteration. Unfortunately, we did not have access to data to capture experience levels prior to the Jazz project. Average Level of Shared-work Experience measures the average number of times that team members worked together in work items prior to the beginning of the focal iteration. Finally, we measured Average Level of Team Workload as the average number of work items owned per team member during the iteration.

Controls for communication-related factors: Since our independent measures focus on patterns of communication, it is important to control for factors related to the propensity of the project member to communicate more or less with other project members or with particular project members. We also computed two measures that captured the project member’s amount of interaction at the dyad level. Amount of Intra-Team Communication captured the number of dyads from the communication networks where both members belonged to the same team. On the other hand, Amount of Inter-Team Communication captured the number of dyads where at least one of the persons in the pair also belonged to a different team. However, a variance inflation factor revealed that the amount of intra-team communication measure was collinear with other control factors so we did not include it in our analysis.

Controls for task-related factors: Since iterations vary in the amount of work they generate and in the importance of the work, we measured the Ratio of High Priority Work Items that each team faced in each iteration as well as the Average Size of Work Items as the average number of files modified in the commits associated with the work items.

Controls for technical-related factors: Past work has shown that the technical dependencies that exist among the technical elements of a system under development are an important driver of dependencies among development tasks and consequently, of interaction among interdependent project members (e.g. [14, 37]. Building on past work [13] on the effect of technical dependencies on quality, we constructed an iteration-wide matrix containing the technical dependencies among system’s components based on the commit information associated with the work items. Since the ownership of each architectural component was allocated to a single team, teams defined a relevant technical boundary. We combined such information with the dependency matrices to construct control measures that capture the nature of the project’s technical dependencies. The Internal Technical Coupling measure refers to the density of the sub-matrix based on the set of relationships defined in the technical dependencies matrix that are contained within a team’s area of responsibility. External Technical Coupling was operationalized as the number of components not under the responsibility of the focal team that had technical dependencies with those components changed by the focal team.

Controls for unobserved effects: Finally, we included a few additional controls for unobserved, team- and iteration-specific factors. These dummy variables effectively control for any factors, which we could not explicitly measure such as the cultural or organization aspects of the teams as well as temporal effects of the iterations. These were recorded as Dummy A through D for the teams and Dummy Iteration A through E for the iterations.

Finally, it is worth pointing out that we computed and evaluated other network measures such as density, which did not have any impact in our models. Therefore, we did not include them in the final results reported in the paper.

Description of the Model
Our two dependent variables, Delivery Performance and Quality, are count variables. Then, we used a multi-level Negative Binomial regression model. We examined our hypotheses by creating several negative binomial models, which included the different control variables as well as our independent measures as fixed effects. In addition, we added iteration, team and individuals as random effects. In order to assess the fit of each model, we report the log-likelihood of each model as well as the percentage of deviance explained by the model. The deviance of a model is defined as “-2 * log-likelihood of the model” and lower values are associated with a better fit of the model to the data. The percentage of the deviance explained is a ratio of the deviance of the null model (containing only the intercept) and the deviance of the final model. Although, regression coefficients are often reported as part of the results, we opted for reporting the incidence rate ratios (IRRs) associated with the fixed effect factors of the regression models because they simplify the interpretation of the results. IRRs indicate the rate at which the dependent variable would change for one unit change in one independent variable keeping the other factors constant.

RESULTS
Preliminary Analyses
In order to develop a robust dataset, we only included those teams who had at least 20 work items in an iteration and had work items planned in at least 4 iterations. We performed sensitivity analyses to examine the impact of such decision and we found that running our regression models including all the data resulted in overfitted estimates. Then, we considered that the decision to remove those data points was adequate. This resulted in data for 16 out of the 26 teams. The 14 iterations and 16 teams resulted in 197 pairs
<iteration, team>. Note that we did not have 224 pairs because each team did not necessarily have work items planned in all of the iterations.

We first computed descriptive statistics of the measures described in the previous section. All variables with the exception of our two independent measures and the GSD measure had skewed distributions so they were log-transformed. We also performed various collinearity diagnostics. We first ran a variance inflation factors (VIF) analysis and it showed that all the measures included in our models did not exhibit collinearity problems since they had VIFs below 5 as recommended guidelines suggest [23]. A pair-wise correlation analysis among those measures also showed levels of correlation that were not problematic.

**The Impact on Delivery Performance**

In this section we discuss the results pertaining to hypotheses 1a and 1b, which focused on the relationship between the hierarchical and “small-worldness” nature of communication patterns on Delivery Performance. Table 1 reports the results of the regression analyses. Models I and II show the incident rate ratios (IRRs) associated with the control factors. We separated the control factors in two different models because we wanted to assess the relative impact of the technical dependencies, a factor that has been shown to have major impact on coordination needs [11] and therefore, a major impact on software development productivity [11] and software failures [12]. The results reported by models I and II are consistent with past research (e.g. [5, 12, 25]). We also observe that several of the measures for unobserved factors associated with the teams and the iterations are statistically significant, suggesting there are idiosyncratic attributes of the teams as well as of the iterations that impact the performance outcomes of an iteration. Models III and IV test hypothesis 1A by introducing the Hierarchical Communication Pattern variable into the analysis. The IRR associated with our independent variable is statistically significant and between 0 and 1 suggesting that higher levels of hierarchy reduce the number of work items that are rescheduled. In other words, increased hierarchical communication is associated with improved iteration performance providing support for H1a. Model IV emphasizes the robustness of that result since the effect remains significant after considering the role of the technical dependencies. It is also worth pointing out that the effect size of hierarchy (2.7% of deviance explained) is similar to the effect size of technical dependencies (3.1% of deviance explained).

Finally, models V and VI in Table 1 assessed hypothesis 1B that posited that small-world communication patterns would be associated with lower delivery performance in our specific case, associated with higher number of rescheduled work items. Our results show no clear support for the hypothesis. The IRR associated with the variable Small-World Communication Patterns are above 1 but the estimates are only marginally significant.

**The Impact on Quality**

We now turn our attention to hypotheses 2A and 2B, which posited that hierarchical communication patterns would be detrimental to quality outcomes while small-world communication patterns would be beneficial. Table 2 reports the results of our analyses. As before, Models I and II show the incident rate ratios (IRRs) associated with the control factors and, again, we separate the control factors in the two models to assess the specific impact of logical technical dependencies. The results are consistent with past research (e.g. [5, 12, 25]). Models III and IV examined hypothesis 2A by introducing into the models our independent variable, Hierarchical Communication Patterns. Our results provide strong support for hypothesis 2A suggesting that hierarchical communication patterns in computer-mediated communication are detrimental to the quality outcomes of iterations. An IRR above 1 indicates that increases in the hierarchical structure of the communication patterns are associated with increases in the number of defects, therefore, associated with poorer quality outcomes. In fact, when we compare models II and III, we see the negative effect of hierarchy is significantly larger than technical dependencies, which prior research suggested as a primary driver for software failures (e.g. [12, 13]). Finally, Models V and VI introduced the Small-World Communication Pattern variable into the analysis, which found support for H2b. Higher levels of “small-worldness” in the communication patterns reduces the estimated number of defects associated with the iteration since the IRR is below 1. It is worth pointing out that the effect size of the variable is not as high as our other independent variable (hierarchy) since it only explained 4.2% of the deviance.

**DISCUSSION**

This paper examined the effect of communication structure, specifically hierarchical and small-world structures on new product development delivery performance and quality. We measured delivery performance by the number of development tasks that needed to be rescheduled to a later iteration. There was a strong positive effect on performance for hierarchy providing evidence that structuring communication patterns in a hierarchical manner improves delivery performance (H1A). However, we found only a marginal negative impact for small-worlds communication structures on delivery performance, providing weak support for hypothesis 1B. We assessed quality by the number of defects that resulted from the work done during an iteration. We found that hierarchical communication structures have a significant detrimental effect on quality (H2A supported) while small-world communication structures improved the quality outcomes of an iteration providing support for hypothesis 2B. We especially draw attention to the significance of the findings after taking technical dependency structures into account. A number of research studies have established that dependencies can drive communication structure [14, 41]. The current study confirmed a positive effect of both intra- and inter-team dependencies on itera-
In a hierarchical structure, there is less overall communication because the communication flows in a particular direction, which minimizes the communication between people at the same level of the hierarchy. This structure is suited to tasks associated with planning and executing against a plan. In contrast, there is a lot of localized communication within each small-world cluster although not necessarily much communication between clusters. This structure is better suited to tasks associated with individual development tasks that can benefit from the contextualization and trust that comes from high communication with a small group of people.

Our results highlight a potentially important trade-off between high delivery performance and high quality outcomes. An analysis of the evolutionary patterns of the communication networks suggested that certain teams were able to overcome the trade-off and achieve high performance and quality by combining hierarchical and small-world communication structures. More specifically, we examined the teams in the top 33 percentile in our outcome variables. These teams, also achieve high levels of both hierarchical and small-world communication. We observed this pattern

<table>
<thead>
<tr>
<th>Controls for unobserved effects</th>
<th>Model I</th>
<th>Model II</th>
<th>Model III</th>
<th>Model IV</th>
<th>Model V</th>
<th>Model VI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Team Dummy A</td>
<td>1.291**</td>
<td>1.357**</td>
<td>1.085</td>
<td>1.364**</td>
<td>1.413**</td>
<td>1.327**</td>
</tr>
<tr>
<td>Team Dummy B</td>
<td>1.069</td>
<td>1.108</td>
<td>1.039</td>
<td>1.119</td>
<td>1.128</td>
<td>1.168</td>
</tr>
<tr>
<td>Team Dummy C</td>
<td>0.841+</td>
<td>0.921**</td>
<td>0.707**</td>
<td>0.920**</td>
<td>0.840*</td>
<td>0.928**</td>
</tr>
<tr>
<td>Team Dummy D</td>
<td>0.624**</td>
<td>0.934*</td>
<td>0.823*</td>
<td>0.911*</td>
<td>0.661**</td>
<td>0.752*</td>
</tr>
<tr>
<td>Iteration Dummy A</td>
<td>0.966</td>
<td>0.862</td>
<td>1.133</td>
<td>0.887</td>
<td>1.137</td>
<td>1.052</td>
</tr>
<tr>
<td>Iteration Dummy B</td>
<td>1.046</td>
<td>0.592</td>
<td>0.995</td>
<td>0.997</td>
<td>0.954</td>
<td>0.817</td>
</tr>
<tr>
<td>Iteration Dummy C</td>
<td>0.591+</td>
<td>0.514**</td>
<td>0.575**</td>
<td>0.515**</td>
<td>0.517**</td>
<td>0.462**</td>
</tr>
<tr>
<td>Iteration Dummy D</td>
<td>0.717+</td>
<td>0.388</td>
<td>0.872</td>
<td>0.400</td>
<td>0.702</td>
<td>0.465</td>
</tr>
<tr>
<td>Iteration Dummy E</td>
<td>0.139**</td>
<td>0.097**</td>
<td>0.101**</td>
<td>0.062**</td>
<td>0.116**</td>
<td>0.085**</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Controls for Team-related factors</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Project Members Involved</td>
<td>2.823**</td>
<td>5.801**</td>
<td>2.398**</td>
<td>5.710**</td>
<td>3.249**</td>
<td>4.263**</td>
</tr>
<tr>
<td>Number of Multi-team Project Members</td>
<td>0.568**</td>
<td>0.816*</td>
<td>0.740*</td>
<td>0.809*</td>
<td>0.590**</td>
<td>0.726*</td>
</tr>
<tr>
<td>GSD</td>
<td>2.631**</td>
<td>2.486**</td>
<td>2.238**</td>
<td>2.509**</td>
<td>2.280**</td>
<td>2.061**</td>
</tr>
<tr>
<td>Average Level of Technical Experience</td>
<td>0.937</td>
<td>1.013</td>
<td>0.992</td>
<td>1.041</td>
<td>0.983</td>
<td>1.011</td>
</tr>
<tr>
<td>Average Level of Shared-work Experience</td>
<td>0.857</td>
<td>0.675*</td>
<td>0.879*</td>
<td>0.601*</td>
<td>0.603*</td>
<td>0.618*</td>
</tr>
<tr>
<td>Average Level of Team Workload</td>
<td>1.643**</td>
<td>1.754**</td>
<td>1.792**</td>
<td>1.734**</td>
<td>1.382**</td>
<td>1.694**</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Controls for Communication-related factors</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Amount of Inter-Team Communication</td>
<td>1.004**</td>
<td>1.022*</td>
<td>1.003**</td>
<td>1.012*</td>
<td>1.002**</td>
<td>1.003*</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Controls for Task-related factors</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Ratio of High Priority Work Items</td>
<td>0.226**</td>
<td>0.701**</td>
<td>0.362**</td>
<td>0.639**</td>
<td>0.363**</td>
<td>0.331**</td>
</tr>
<tr>
<td>Average Size of Work Items</td>
<td>1.152</td>
<td>1.066</td>
<td>1.101</td>
<td>1.049</td>
<td>1.203</td>
<td>1.262</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Controls for Technical-related factors</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Internal Technical Coupling</td>
<td>1.302</td>
<td>1.094</td>
<td>1.036</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>External Technical Coupling</td>
<td>2.331**</td>
<td>2.277**</td>
<td>1.869**</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Independent Variables</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Hierarchical Communication Pattern</td>
<td>0.669*</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Small World Communication Pattern</td>
<td>0.843*</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Log-Likelihood</td>
<td>-523.41</td>
<td>-502.08</td>
<td>-498.41</td>
<td>-480.71</td>
<td>-518.59</td>
<td>-500.63</td>
</tr>
<tr>
<td>Deviance Explained</td>
<td>24.8%</td>
<td>27.9%</td>
<td>28.4%</td>
<td>31.1%</td>
<td>25.5%</td>
<td>28.1%</td>
</tr>
</tbody>
</table>

(+ p < 0.10, * p < 0.05, ** p < 0.01)
in at least 1 iteration for 13 teams, however, the pattern was more prevalent for 3 teams where they exhibited the pattern in more than 3 iterations. On the other hand, other teams developed only one particular pattern or they flip-flopped between the hierarchical and small-world patterns. These teams were never able to develop both high levels of delivery performance and quality within an iteration. Consequently, they faced a trade-off between performance and quality or both outcomes were not particularly high. Furthermore, in several instances those teams (9 out of the 26 teams) exhibited both low levels of delivery performance and quality in at least 1 iteration of the project.

**Limitations**

Our study has several limitations worth highlighting. First, we studied a single project, which might raise external validity concerns. Although we recognize such a risk, we think our research setting represents an unique opportunity to perform a longitudinal study of a new product development project and the communication patterns that emerge over time through computer-mediated-communication means. A second limitation of the study is the focus on the first release of the product. The project has continued to work together and produce other releases of the product. Then, we could expect maturation effects that might impact the patterns of communication and their evolution. Unfortunately, the data associated with subsequent releases are not currently available. Finally, several teams had collocated members and in our study we were not able to collect data to control for face-to-face interactions that might impact how the communication patterns in the work items reports evolve. However, our interviews confirmed that comments were the primary form of technical communication as befits a project that had transparency as a goal.

**Implications for Tool Design**

Although our research takes place in an NPD context, the relevance of this domain for the design of collaborative tools is well established in the CSCW literature (e.g. [18, 19, 34, 36]). The effect of distributed work on awareness and collaboration has been extensively studied in software...
engineering (e.g. [18, 36]) as well as other contexts (e.g. [34]). Our results provide two key contributions to the design of collaborative tools. First, current collaborative tools have been shown to be successful in small teams, however the current generation of designs is inadequate for large-scale and distributed projects [36]. For instance, existing technologies inform developers of a relatively large number of events of possible interest. However, in large-scale projects, such an approach will quickly exhaust attentional resources. The results of our study suggest an alternate approach in which collaborative tools target specific pairs or groups of people, to reduce the problem of information overload. Data on communication patterns – who talks to whom – can be used to focus presence, activity and other types of information relevant to a particular user as a way of reducing information load.

Second, and arguably more importantly, our work shows that two communication structures have different impacts on different product development outcomes, pointing to an important trade-off that collaborative tool designers need to understand in order to develop effective tools. The approach we used to construct the communication networks and our measures point to a potential technique that collaborative tools could leverage in order to better support the evolving nature of NPD projects.

Implications for Additional Future Research
The results of our study have important implications for future research. First, our results point to a potential trade-off that development teams may face in terms of the structural properties of the communication patterns that emerge during an iteration and how those patterns impact two important outcomes in new product development: delivery performance and quality. Arguably, development teams could reap benefits in both performance and quality if they are able to modulate their communication in accordance with the type of work being done, as well as in alignment with the requirements imposed by technical dependencies [11]. However, a closer look at the teams in the dataset indicated that some of the teams were able to maintain above average delivery performance and quality. In those cases, we observe that the evolution of the hierarchy and small-worldness measures alternated between above average levels and average levels suggesting that those teams do seem to vary in their ability to change communication structures. Then, future research should examine these aspects and their implications on NPD outcomes.

Second, and in relation to the previous point, the trade-off challenge is augmented significantly in the context of geographically distributed product development. Ramassubu and colleagues [35] showed that the team configurations that are beneficial for productivity in geographically distributed development projects tend to be highly detrimental in terms of quality. Those results combined with those reported in the paper suggest that the trade-off between performance and quality could stem from the relationship of those outcomes with the structure of communication patterns as well as other important organizational factors such as the team configuration and the dispersion of expertise across the project’s locations. Future research should examine the relative role of all those factors on NPD outcomes to gain insight into the specifics nature of the trade-off.

REFERENCES


