

Global software development and delay: Does distance still matter?

Thanh Nguyen, Timo Wolf, Daniela Damian

Department of Computer Science

University of Victoria, Canada

Software Engineering interAction Lab (SEGAL)

{duythanh, timowolf}@uvic.ca, danielad@cs.uvic.ca

Abstract

Nowadays, distributed development is common in software development. Besides many advantages, research in the last decade has consistently found that distribution has a negative impact on collaboration in general, and communication and task completion time in particular. Adapted processes, practices and tools are demanded to overcome these challenges.

We report on an empirical study of communication structures and delay, as well as task completion times in IBM'sTM distributed development project JazzTM. The Jazz project explicitly focuses on distributed collaboration and has adapted processes and tools to overcome known challenges. We explored the effect of distance on communication and task completion time and use social network analysis to obtain insights about the collaboration in the Jazz project. We discuss our findings in the light of existing literature on distributed collaboration and delays.

1. Introduction

Global software development promises many benefits such as decreased development cost, access to a larger skill pool, proximity to customer, or twenty-four hour development by following the sun. Unfortunately, many challenges have to be addressed for global teams to truly benefit from global development. These challenges cover many facets of software engineering, including processes, infrastructure, or business strategies [22]. A recurring challenge, constantly recognized by research and industry, is cross-site communication and collaboration [1, 4, 12, 8, 21]. Studies found that cross-site development took longer than single site [11] and that follow the sun development is not realized successfully [15].

Specialized processes, practices, tools as well as an awareness of effects caused by geographical distribution are demanded and required to overcome the cross-site commu-

nication and collaboration challenges. While these challenges were new some years ago, nowadays, specialized tools are being developed (e.g. Jazz [6], EGRET [27], Sysiphus [3]) and companies are more aware of these challenges and adapting their processes and practices for distributed development.

A significant amount of research has reported about delays in communication of distributed teams, since one of the early reports by Herbsleb and colleagues [7, 10, 11] and an acknowledgment that *distance matters* [20]. In this paper we report from an empirical study of communication and task delay in global software development involving 16 distributed sites around the world. We study a distributed project at IBM, the Jazz development project, over a period of 23 months. The project uses the Jazz platform, which is a state-of-the-practice development environment aiming at providing collaborative support to distributed teams [6]. Our goal was to investigate whether the effects of distance reported in the literature in the last decade are still significant in affecting the project communication and delay.

Our paper is structured as follows. In Section 2 we motivate our work by discussing related work in collaborative software development, as well as current reports about challenges in distributed communication. Our research questions are provided in Section 3. Specifically, we focus on effects of distribution on communication delay, communication structures and task completion times. The study settings, the data constructs and the data collection are described in Section 4. Section 5 reports our analysis, and results are discussed in Section 6. Possible threats to validity are discussed in Section 7. We conclude our paper in Section 8.

2. Related work

We review here the results of empirical studies of distance, communication and delay and motivate the design of our investigation into Jazz.

The importance of collaboration and the special role

played by *informal communication* in software development is now well known (e.g. [9, 17]). Whereas formal and planned communication is sufficient to deal with planned activities, informal communication is critical to react on unanticipated and unplanned issues. In distributed settings, however, informal cross-site communication becomes problematic, impacting activities of expertise finding and interpersonal relationship building [7]. This in turn translates to problems of delay in distributed projects.

One of the more notable series of empirical investigations of distance and delay in software development is the work of Herbsleb and colleagues. Early reports of communication problems and lengthened cycle time to resolve systems issues (e.g. [10, 9]), are followed by systematic studies into distance and speed at large distributed organizations (e.g. [11, 13]). The 2003 study [11] at Lucent provides systematic evidence about significant *communication delay* and *task completion delay* for modification requests that involved cross-site work. A comparison of data from same-site and cross-site projects indicates that tasks involving distributed participants take about 2.5-times longer to complete than similar collocated tasks, result that is explained by the perceived communication delay as reported in interview data. Other factors influencing task completion time included the number of people involved in the task, as well as the size of the task. Not surprising to this study, there were significant differences in the size of distributed vs. same-site communication networks. This negative impact of distance on the properties of distributed social networks has recently been found in by Ehrlich and colleagues as well [5].

Most of these findings are confirmed in later studies. An experience report [13] of nine distributed projects at Siemens brings insights about benefits and challenges in distributed work, identified through interviews conducted at three different sites. Besides reported benefits, communication and collaboration across sites continued to be identified as a big problem, leading to *reduced development pace*. The interviewees clearly stated that face-to-face communication is the most important and fastest communication. Frustrations arose when the pace of interaction went down.

In contrast to work in industry commercial environments, open source projects do not seem to have problems with distributed communication, collaboration and development following the sun. In 2006, Spinellis [29] analyzed the source code from the CVS repository, the bug reporting system and the geographical developer locations of the freeBSD operating system. He found that round-the-clock development takes place and the amount of distance does not negatively affects performance, code quality or the density of defects. Although these findings are promising, it is difficult to derive lessons learned for industrial projects, as the characteristics of open source projects are very dif-

ferent. First, the analysis has no collocated control group so that everyone is distributed and face-to-face communication is not existent. Thus, it is impossible to conclude that distributed development works as well as collocated development. Second, project characteristics such as developing on time and on budget are not existent. Third, developers are volunteers and are highly experienced and motivated.

In summary, the literature in the last decade contains a body of empirical evidence about increased communication delay and work completion time in distributed projects, generally due to lack of informal communication and difficulties in expertise finding. In response to these challenges, the research community proposed new approaches and tools to support more effective collaboration in distributed software engineering. Many solutions focus on integrating informal communication, collaboration and maintaining awareness across sites into development tools [18, 24, 27, 28, 3, 30, 19]. While they are promising, systematic evaluation of these approaches and tools in large-scale distributed projects is needed.

In this paper we describe an empirical study that aims at evaluating the impact of distance on communication and task completion time after almost a decade since the problems of distance have been reported. We specifically investigate distance and delay in a project that uses the Jazz development tool and which was developed specifically to support the collaboration of large distributed teams. The outline of our research questions in the next section is followed by the description of our research methodology.

3. Research Questions

Our research questions center around communication delay, task completion time and communication structures in distributed collaboration, as well as other factors that could introduce delays in communication besides geographical distance. Our investigation is in the context of Jazz development, of which characteristics we describe in detail in Section 4.1.

RQ1: Does distributed communication introduce delay as compared to same-site communication? Previous research [7, 11, 23] identified communication delay by increased response times in distributed projects. We examine quantitative data from the Jazz project and compare response times between same-site and multiple-site communication.

RQ2: Do distributed tasks take longer to complete than same-site tasks? Previous research [11, 23] found that tasks involving people from distributed geographical sites take longer as tasks involving people from only one site. We investigate if these findings still hold by

comparing completion times between tasks involving different numbers of geographical distributed sites.

RQ3: What other factors influence communication delay and time to complete a task? Many factors besides the geographical distribution may influence the communication and task delay [11]. We investigate the relationship between communication and task delay and other factors such as the number of participants in the communication or the severity and priority of the tasks.

RQ4: What communication structure is used in distributed settings? In the study of teams, social networks are recognized as important in fostering relationships, trust and knowledge management. In distributed software development teams in particular, the study of social networks proved useful in understanding patterns of collaboration and coordination in global teams [5, 14]. Previous research identified differences in the size and communication structures of same-site vs. distributed communication networks. We examine the collaboration structure of the entire distributed project by using social network analysis techniques.

4. Methodology

This section describes the design of our study. Before we describe our constructs and data collection methods, we provide details about the study setting, i.e. the development setup of the Jazz project.

4.1. Study settings

We studied communication and collaboration in the Jazz development environment, over a period of 23 months.

Jazz is a development environment that tightly integrates programming, communication, and project management. With collaboration support as one of its main goals, Jazz provides integrated support for work planning and tracking, continuous builds, static code analysis and reporting tools to measure the quality of the work products and the development process.

Jazz development involves 16 different sites that are located in the United States, Canada, and Europe. The development and testing takes place at 7 different sites and it involves 151 active participants working in 47 functional teams. From now on we will only refer to the 7 sites in our study. Jazz members belong to multiple functional teams (referred to as teams henceforth), each managed by a team lead. The team leads report to a project management committee, which is responsible for the project-wide coordination. The team size ranges from 1 to 20 and has an average

of 5.7 members. The number of developers per geographical site ranges from 7 to 24 and is 14.8 in average.

The Jazz development uses the *Eclipse Way* process. It defines six-week iteration cycles, which are separated into planning, development and stabilize activities. Every iteration releases a new milestone build of the product. The project management committee formulates the goals and features for each release. Every team has an iteration plan that consists of unstructured text and task descriptions, which are captured in *work items*.

Work items represent single assignable and traceable tasks. Different types of work items are created to represent defects, enhancements, and general tasks. Commenting on the work items is the main task-related communication channel. A second broadcast communication channel is email, which is mostly used for announcements. Chat is used for synchronous communication among the developers.

In this paper, we focus on work items as they represent the most identifiable unit of work, i.e. tasks, and which provide the context for communication and collaboration. Task related communication is established by commenting on the work items. Depending on the location and team memberships of the authors, the communication can be across teams and different geographical sites.

4.2. Data constructs

The main variables of interest in our study are distance, communication and task completion time. Our unit of analysis is the work item and the associated collaboration that involves members from same and different-sites. We define the following constructs:

Number of sites involved in the communication and completion of a work item is our conceptualization of distance. For each work item, we use the location of each author of a comment to obtain the number of sites that were involved in communication. For example, if a work item had two commenting authors from two different sites then the work item falls into the two-site category.

Response Time is our conceptualization of communication and its possible delay because of distance. We assume that the comment thread on a work item represents a conversation about the work item, and that a comment is a reply to the previously created comment. We measure the time intervals between the creation dates of sequential comments. For each work item, the response time is the average of all comment time intervals of a work item.

Resolution Time is our conceptualization of task completion time. For each work item, we compute the dif-

ference between the completion date and the creation date.

Previous literature [11] suggests that also other factors may affect communication in distributed development. We study the possible relationships between response and task completion time with the following variables: *Number of comments* on a work time, as a measure of how controversy the work item was discussed; *Number of authors* as the number of different authors that created the comments of each work item; *Severity* of a work item as an indication of how critical the work item is (could be unclassified, minor, normal, major, critical, and blocker), and *Priority* of a work item, as a measure of work item importance during planning (could be unassigned, low, medium, or high).

4.3. Data collection

The main artifacts, including work items, contributors, representing the Jazz developers, and the team structures are located in the server component of Jazz. A query plugin was implemented to extract the required artifacts from the Jazz server. We imported the resulting data into a relational database management system.

We extracted a total number of 18,618 work items. However, as some geographically locations of Jazz participants were unknown, we decided to remove 4,876 work items in which any missing contributor location or any unknown contributor was involved. In the resulting data, we could identify all authors, locations and the creation dates for all work items and comments.

Our analysis thus involved data on a total of 13,742 work items that were created between October 2005 and November 2007. 0.73% of these work items were created in 2005, 29.46% in 2006, and 69.82% in 2007. The work items have 43,967 different comments in total. The number of comments per work items ranges from 0 to 58 and 2.75 in average. 2,669 work items had no comments and after excluding them, the average number of comments became 3.41. We

	Number Of Work Items	Mean (in days)	Median (in days)	Mean of first two comments
1 Site	4543	18.88	2.43	17.77
2 Sites	2448	16.72	3.04	14.06
3 Sites	406	13.21	3.70	11.76
4 Sites	66	8.42	4.99	5.51
5 Sites	10	5.60	4.07	1.03

Table 1. The mean and median response times in days for the different numbers of involved sites.

identified 148 contributors including their locations, and 52 functional teams.

5. Data Analysis and Results

In this section we describe the data analysis methods and findings in relation to each of our research question in part. As our response variables do not follow a normal distribution (e.g. see Figure 4 and Figure 5), we use nonparametric statistics for the tests reported below.

RQ1: Does distributed communication introduce delay as compared to same-site communication?

To examine response time differences for work items involving communication from different numbers of sites, we categorized the work items by the number of involved sites and computed the average response times for each of the work items. Table 1 shows the different categories (up to 5 sites involved in a work item communication), the number of work items in each category, and the mean and median response time for each category. We also computed the mean of the first two comments in each site category as a comparison to the mean response time of all comments.

One can see that the means and medians behave differently when the number of sites increases. While the mean response time decreases, the median response time increases when more sites are involved. With data that is not normally distributed the median values become a better indicator of variations in the variable. Note that the number of work items when communication involved up to three sites is much larger than when communication involved 4 or 5 sites.

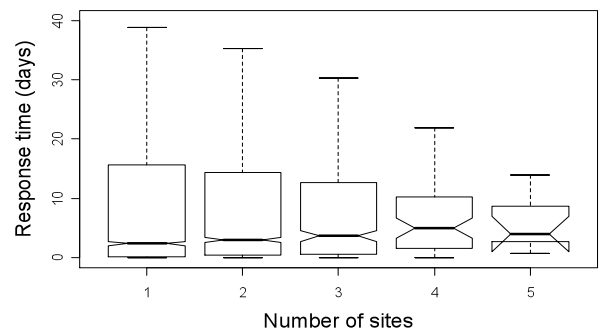


Figure 1. Box plot of the work item response times categorized by the number of sites involved in the work item communication.

The box plot in Figure 1 shows the response times of all work items categorized by their number of sites. The horizontal bar in the middle shows the median value. The bottom / top line of the box show the 25th / 75th percentile. The complete box show where 50% of the data lie. The whiskers show the minimum and maximum response time values, excluding outliers.

The Kruskal-Wallis test of difference in response times across the 5 different number of site categories yields a statistically significant difference ($K=26.31$, $p<0.001$).

To examine the size of this effect, we applied correlation tests. In other words, to investigate the strength of the relationship between distribution and delay in communication, we tested for a correlation between the number of sites and the average work item response time. We ran the non-parametric correlation test Kendall Tau [26]. The correlation result yielded a very low correlation value ($\tau=0.05$, $p<0.001$), which means that distributed communication does not appear to introduce a significant amount of delay compared to same-site communication.

RQ2: Do distributed tasks take longer to complete than same-site tasks?

Similar to the previous analysis, in order to compare resolution times of work items completed at the same site or distributed sites, we categorized the work items by their number of sites and calculated the mean and median resolution times for all work items (see Table 2).

	Number Of Work Items	Mean (in days)	Median (in days)
1 Site	3975	43.17	8.38
2 Sites	2019	54.10	13.00
3 Sites	329	72.86	25.96
4 Sites	51	85.90	50.75
5 Sites	5	53.13	55.00

Table 2. The mean and median resolution times in days for the different numbers of involved sites.

In contrast to the effects on response time, the trend in the mean and median resolution times as the number of sites shows an increase in both means and medians when the communication involved contributors from one to many sites.

The box plot in Figure 2 also shows that the maximum and the variance of resolution time increases when the number of involved sites increases. While the Kruskal-Wallis test of difference in resolution times across the 5 different number of site categories yields a statistically significant

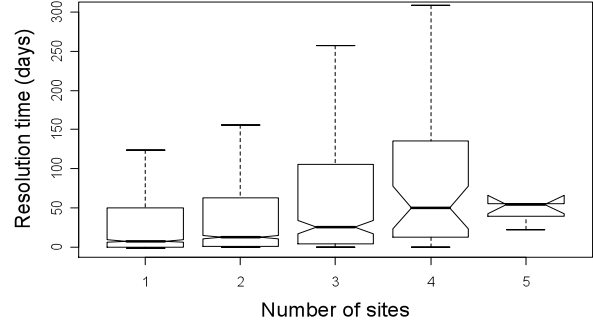


Figure 2. Box plot of the work item resolution times categorized by the number of sites involved in the work item communication.

difference ($K=101.37$, $p<0.001$), the Kendall Tau correlation test of association yielded an extremely low correlation value ($\tau=0.09$, $p<0.01$). This indicates that distributed communication does not appear to introduce a significant amount of delay compared to same-site task completion time.

RQ3: What other factors influence communication delay and time to complete a task?

To investigate whether other factors influence the communication delay and task completion time, we investigated the relationship between response and resolution time and the other factors defined in Section 4.2: response resolution time, severity, number of commenting authors, number of comments, number of sites and priority. Table 3 shows the results of the Kendall Tau correlation tests for each of these relationships.

Despite these correlations being statistically significant at $p<0.001$, the actual correlation strengths are very low, except for the association between response time and resolution time, the association between number of authors and

Factor	1	2	3	4	5	6	7
1 Response Time	1.0	0.54	-0.10	0.08	0.05	0.05	-0.04
2 Resolution Time		1.0	-0.13	0.11	0.15	0.09	-0.08
3 Severity			1.0	0.10	0.12	0.10	0.14
4 No. of authors				1.0	0.56	0.53	0.02
5 No. of comments					1.0	0.35	0.03
6 Number of sites						1.0	0.02
7 Priority							1.0

Table 3. Kendall Tau correlation results. The column header relate to the row numbers.

number of comments, and the association between number of authors and number of sites. We believe that these correlations, the very low included, were found statistically significant due to our very large sample size.

RQ4: What communication structure is used in distributed settings?

To obtain more insights about the collaboration patterns in Jazz, we constructed a project-wide communication-based social network, as shown in Figure 3. In the network, each node represents a project participant, and a tie is drawn between two project members if they commented on at least one common work item. The network has 112 nodes and 3296 ties. The shape of a node represents the geographical location of the participant, locations which are also highlighted by grey line ovals.

To investigate communication patterns and information flow in the network, we ran a core-periphery test [2], to identify the presence of a subset of the network that acts as a core in the network-wide communication. Studies (e.g. [14]) shown that networks with a strong core-periphery structure are better than fractionated networks because they have less coordination problems. In a core-periphery structure, members of the core will act as specialists who broker the information more efficiently than in fractionated networks, where members are connected in loosely connected cliques. The core-periphery test yielded a correlation of 0.758, which is a high value indicating the presence of a core. We also computed a measure of the core, using the k-core measure [25], where the k indicates that each member connects to k other members in the core.

In Figure 3, the center nodes have more connections than the outer nodes. The k-core with $k > 25$ is highlighted with a circle in the center. It has a membership of 60 out of 112 members in the entire Jazz project, and a number of 2118 ties out of a total of 3296 ties in the network. This indicates a large core including almost half of the members and about 75% of the communication in the project.

6. Discussion

Motivated by reports of challenges of distributed development and advances in collaborative software development environments in the last decade, our study sought to bring yet another piece of evidence about coordination, communication and delay in software engineering.

Although delays in communication have been a recurring theme in published reports of industrial practice, our investigation of collaboration patterns in the Jazz team does not show the same effect. In what follows we interpret the results of the several analyses we conducted on response

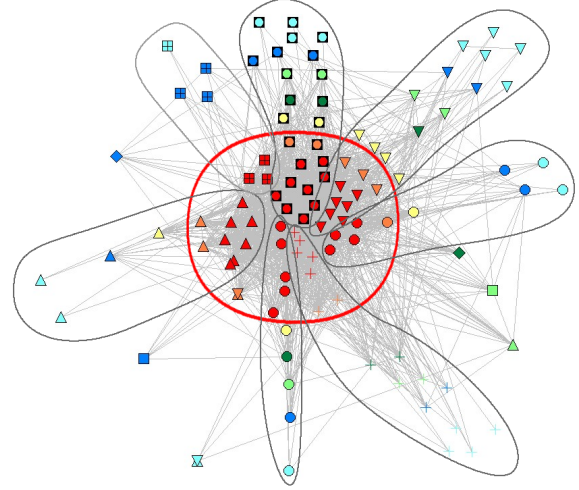


Figure 3. Communication-based social network of the Jazz project. The circle in the center shows the k-core ($k > 25$). The grey ovals indicate different geographical locations.

time and task completion times in Jazz, along with the findings of communication structure in the Jazz project-wide social network. It is our intention to reveal a rich picture of collaboration supported by one of the more recent development tools, focused on supporting collaboration in distributed teams: Jazz.

Communication and task completion in distributed teams

In contrast to previous studies, we did not find that geographical distance introduces significant delays in communication and task completion. Based on previous literature, we expected that responses on and resolution of work assignments would take longer as the number of sites involved in the communication increased. Thus we analyzed the response and resolution times in relation to the number of sites involved in each work item. Our findings show no clear impact of distance. Although the test of difference Kruskal-Wallis tests showed statistical significance (indicating that the variation in *one* of the distributed communication settings is different from the others), the correlation results that tested the size of this effect were extremely low. This indicates that variations in communication and task completion times cannot be at all associated with the variation in geographical distance of collaborators in the project. Thus, the expectation that larger distribution, as found in collaboration involved a higher number of sites, is associated with an increase in both response and work item resolution times

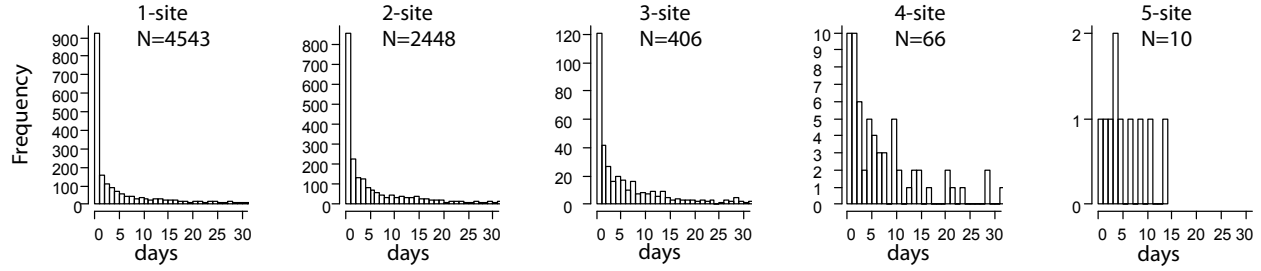


Figure 4. Histograms of response times categorized by number of sites.

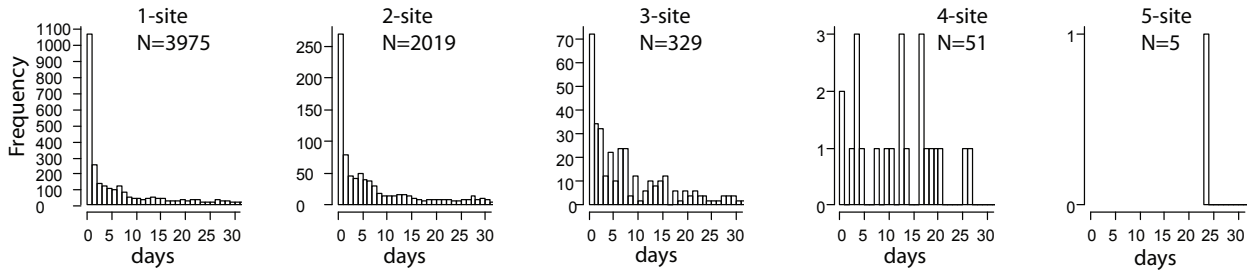


Figure 5. Histograms of resolution times categorized by number of sites.

was not confirmed in our study.

One possible explanation for this difference in our findings is that previous studies used qualitative, self-reported interview or survey data, which might not have provided an objective measure of communication delay. The interviewed participants might have remembered and reported only about high delayed communication instances, possibly a small and unrepresentative sample to generalize from. Our data is less prone to participant recollection problems.

Another possible explanation might come from the characteristics of the Jazz project participants. We can expect that developers developing a tool that focus on collaboration support reflect on how they communicate and collaborate within the project and optimize these activities. Thus, they may perform better than developers in other projects, which focus on a different domain and deal with distribution issues as lower priority items. In addition, demographic information such as age and familiarity with text based communication specific to commenting on work items or chat might influence the ability to collaborate across sites. Unfortunately, we do not have demographic information of the Jazz team.

The descriptive statistics in our data set also allows us to do a more in-depth analysis of the trends in communication and resolution times in same-site and distributed communi-

cation:

First we discuss the distribution of response times for the five different categories of work items as shown in Figure 4. With the exception of the 5-site category, the 1,2,3 and 4-site categories show a consistent pattern: about one third of work items have an average response time of one day; about 10% of work items have an average response time of 2 days (exception is in the 4-site category), and the remaining work items exhibit a long-tail distribution pattern indicating the presence of some comments with very long response times. This distribution, together with information shown in the box plot may explain the inverse trend in the pattern of mean vs. median shown in Table 1: the number of long responses decreases as the number of involved sites increases (mean value decreases as number of sites increases), whereas there are fewer quick response as the number of sites increases (median value increases as number of sites increases).

Further, as our data is not normally distributed, the median is an important measure for the trend in response time (as opposed to means). As such, one can observe that the median response time in the 2-site distributed communication is only 0.61 days larger than the median response time in the same-site communication category. Without claiming that this is the difference in average times between same-site and 2-site response times (as such computation is not per-

missible on median values), we believe it still indicates that collaborators in 2-site communication wait for a response in average about half a day longer than those in co-located communication. When this is considered in the light of 6-week long iteration cycles, one may understandably not find it disturbing. A similar trend can be observed as the number of sites increases. The 5-site case is somewhat difficult to analyze. With a small sample size of 10 work items, the response time distribution shows an almost even distribution of the response times from 1 to 13 days. However, this is too small of a sample to allow us to regard it as representative for this case.

In summary, we believe that the geographical distribution did not have a practical effect on the response times in the Jazz environment.

Second, the evidence on the resolution times in the several distributed communication settings is not as clear and rather mixed. The descriptive statistics provide interesting insights about the distribution of resolution times when the number of sites increases. As shown in Table 2, both mean and median values increase from 1 to 5-site communications. The histograms in Figure 5 show a similar trend for the first three 1,2 and 3-site categories: in each of the categories there are many work items with short resolution times, followed by the remaining in a long-tail like distribution; about 20% of resolution times are within one day for both same-site and 2-site communication, while the percentage of same resolution times is 10% for 3-site communication. The fact that both median and mean values increase as the number of sites increases is indicating that likely the number of very long resolution time work items (at the end of the long tail) is increasing as the number of sites increases. The data in the 4 and 5-site categories however is rather small in size to allow us to draw conclusions about any pattern. As seen in the histograms, the resolution times for the 51 work items in the 4-site category are almost evenly distributed between 0 to 27-day wait time, with some outliers with over 30-day wait time. Similarly, the data for the 5-site category shows rather long resolution times. Again, the very few data points in our set, 5, is not sufficient in drawing any conclusions for the trend in the 5-site communication category. In summary, we believe that the difference in resolution times when the number of sites increases is not practically significant, given the rather similar distribution of response time values in the same site, 2 and 3-site distributed communication.

Interestingly enough, none of the investigations into correlations between response and resolution times and other factors as suggested by literature revealed possible alternative explanations for variations in communication and task completion in the different distributed settings. We thus turn our attention to the findings about the communication structure in the Jazz social network, to explain our findings on

communication and task completion.

Communication structure in distributed social networks

The lack of significant communication and resolution time delay does not come as a surprise when the characteristics of the Jazz collaboration environment are being considered. Drawing on knowledge of the Jazz process, the Eclipse Way, as well as on our private conversations with Jazz project managers, we provide further explanation to our findings: The applied process of the Jazz project, with its collection of best practices and the supporting Jazz tool, contribute positively to reducing communication delay. As indicated to us, Jazz project members are encouraged to immediately respond to requests and comments from members of remote teams. This practice is conducive of cross-site communication and, apart from reducing the communication delay across sites, contributes to enhanced familiarity and closeness of participants located at different geographic sites. As a result, increased familiarity reduces communication problems such as misunderstandings and mistrust across sites [7, 16] and contributes to a denser social network in the project.

One important finding about the communication structure in Jazz is that the project-wide communication-based social network has a large core of active contributors, suggesting that many project members were actively communicating across functional teams as well as distances. This finding contradicts expectations formed by previous studies. Since maintaining interpersonal relationships and awareness of work and expertise have been found challenging in distributed teams (e.g. [5, 7, 10]), a less dense network would be expected in such a large distributed team. Distributed social networks are typically significantly smaller than same-site networks, with a restricted flow of information across sites [5, 11]. The presence of particular team members acting as *point people* through whom much of the team communication flows to the distanced teams, was found to ensure the efficiency of coordination across teams [14].

The Jazz network, in contrast, appears to be very connected across distances and functional teams. While we did not compare the properties of same-site vs. distributed networks in Jazz, we find that the distributed network has a large core. Instead of having multiple clusters of connected cores, each associated with each geographical location in the project, the entire project communicates through one core. With a core comprising about half of the entire project team, the network in Figure 3 also shows that about half of the members from each geographical location are in the core. Having multiple members of each team in the core, connecting their team to others, reduces possible communi-

cation bottleneck problems and introduces redundant communication channels, enabling fast communication. While these core members may act as information brokers to the rest of the team, the network as a whole exhibits a rather informal hierarchical organization, since the other peripheral members in each geographical location appear to be fairly connected to the other members that are in and out of the core area.

We believe that this particular project-wide communication structure was enabled and continually supported by the above-mentioned communication practices, substantially contributing to reducing possible communication delay caused by distance.

7. Threats to validity

We recognize a number of threats in the construct and external validity of our study results.

7.1. Construct validity

Our **communication** is based on comments on work items. Besides commenting on work items, the Jazz project members also use other sources of information and that we did not consider, such as email lists, chat, web-based information and face-to-face meetings. Based on our conversations with the Jazz team and observation of the work item online commenting behavior, we believe that comments are most commonly used method to communicate about work items, largely because the comments were work item-specific information immediately available with the work item and easily accessible to project members irrespective of their geographical location.

Our conceptualization of **response time** assumes that every comment on a work item is a response to the previous comment on that work item. The assumption might not be correct for all comments, since a comment might be also independent from the previous comments. We investigated some work items manually and assume that our assumption is correct for the majority of comments. Further, because the second comment is more likely a response to a first comment, we also computed the means of only the first two comments (see Table 1). Since the means in both cases are similar, we are more confident that this threat is not as important.

The **resolution time** construct is based on the creation and resolution date of work items to measure the task completion times. This interval also includes possible idle time, i.e. the time between the creation and the actual time when someone starts working on the task described by the work item. For example, an unimportant work items may be created and then ignored for a long time, until someone resolves it in a short time. As we have a very large sample

size, we believe that these rare cases lose their impact on our results.

7.2. External validity

We studied IBM's distributed project Jazz, of which characteristics are very specific. The Jazz team uses an integrated development environment, while self-hosting the development of the tool they are developing. Similar results may be expected if another distributed project uses Jazz, and follow the provided Eclipse Way process, including their best practices.

8. Conclusion

Does distance still matter? We show that distance does not have as strong of an effect on distributed communication delay and task completion as we have seen in past research. The effect of distance is being mitigated in collaborative environments such as Jazz, in which communication in large distributed work teams is facilitated by an ability to asynchronously comment on and tracking activity of work items. The mechanism of contributing to a shared repository containing work items and associated comments from members that contributed to their implementation facilitates knowledge exchange and aids expertise seeking. These mechanisms together with processes and practices that particularly account for distributed interactions enable effective cross-site collaboration.

However, we recognize that we have made only a first step in the study of development environments that enable complex and richer than before distributed collaboration. Perhaps more important than answers are questions that our research is raising about today's distributed collaboration:

- What are the specific practices enabled by rich collaboration environments that allow distributed groups to engage in cross-site collaboration that is as timely and effective as same-site collaboration?
- What is the interplay between asynchronous commenting on work items of interest and other communication channels that contribute to effective cross-site collaboration?
- What are the main technical features in today's collaborative environments such as Jazz that are successfully supporting distributed collaboration?
- What brings people together in successful distributed collaboration? Is it the characteristics of the product being developed, history of working relationships or the affordances of advanced collaborative development environments?

The research community as well as the software practitioners will benefit from immediate answers to these questions.

9. Acknowledgements

We want to thank all members of the IBM Jazz project who helped us, particularly Harold Ossher of IBM TJ Watson. Many thanks to the members of the SEGAL lab. Funding was provided by NSERC and IBM Jazz Technology Fellowship Award. IBM and Jazz are trademarks of IBM Corporation in the US, other countries, or both.

References

- [1] R. D. Battin, R. Crocker, J. Kreidler, and K. Subramanian. Leveraging Resources in Global Software Development. *IEEE Software*, pages 70–77, March/April 2001.
- [2] S. P. Borgatti and M. G. Everett. Models of core/periphery structures. *Social Networks*, 21(4):375–395, 2000.
- [3] B. Bruegge, A. H. Dutoit, and T. Wolf. Sysphus: Enabling Informal Collaboration in Global Software Development. In *Proceedings of the 1st Intl. Conference on Global Software Engineering (ICGSE 06)*, 2006.
- [4] D. Damian. Stakeholders in Global Requirements Engineering: Lessons Learned from Practice. *IEEE Software*, 24(2):21–27, 2007.
- [5] K. Ehrlich, G. Valetto, and M. Helander. Seeing inside: Using social network analysis to understand patterns of collaboration and coordination in global software teams. In *Proceedings of the 2nd Intl. Conference on Global Software Engineering (ICGSE 07)*, 2007.
- [6] R. Frost. Jazz and the Eclipse Way of Collaboration. *IEEE Software*, 24(06):114–117, 2007.
- [7] R. E. Grinter, J. D. Herbsleb, and D. E. Perry. The Geography of Coordination: Dealing with Distance in R&D Work. In *GROUP '99: Proceedings of the Intl. ACM SIGGROUP conference on Supporting group work*, 1999.
- [8] J. D. Herbsleb. Global Software Engineering: The Future of Socio-technical Coordination. In *FOSE '07: 2007 Future of Software Engineering*, 2007.
- [9] J. D. Herbsleb and R. E. Grinter. Architectures, Coordination, and Distance: Conway's Law and Beyond. *IEEE Software*, 16(5):63–70, 1999.
- [10] J. D. Herbsleb and R. E. Grinter. Splitting the organization and integrating the code: Conway's law revisited. In *ICSE '99: Proceedings of the 21st Intl. Conference on Software engineering*, pages 85–95, 1999.
- [11] J. D. Herbsleb and A. Mockus. An Empirical Study of Speed and Communication in Globally Distributed Software Development. *IEEE Transactions on Software Engineering*, 29(6):481–494, June 2003.
- [12] J. D. Herbsleb and D. Moitra. Guest Editors' Introduction: Global Software Development. *IEEE Software*, 18(2), 2001.
- [13] J. D. Herbsleb, D. J. Paulish, and M. Bass. Global Software Development at Siemens: Experience from Nine Projects. In *ICSE '05: Proceedings of the 27th Intl. Conference on Software engineering*, pages 524–533, 2005.
- [14] P. Hinds and C. McGrath. Structures that Work: Social Structure, Work Structure and Coordination Ease in Geographically Distributed Teams. In *Proceedings of the 20th Conference on Computer Supported Cooperative Work*, 2006.
- [15] H. Holmstrom, E. O. Conchuir, P. J. Ågerfalk, and B. Fitzgerald. Global Software Development Challenges: A Case Study on Temporal, Geographical and Socio-Cultural Distance. In *Proceedings of the 1st Intl. Conference on Global Software Engineering (ICGSE 06)*, 2006.
- [16] S. L. Jarvenpaa and D. E. Leidner. Communication and Trust in Global Virtual Teams. *Organization Science*, 10(6):791–815, 1999.
- [17] R. E. Kraut and L. A. Streeter. Coordination in Software Development. *Communications of the ACM*, 38(3):69–81, March 1995.
- [18] F. Lanubile, T. Mallardo, and F. Calefato. Tool Support for Geographically Dispersed Inspection Teams. *Software Process: Improvement and Practice*, 8(4):217–231, October/December 2003.
- [19] B. Lings, B. Lundell, P. J. Ågerfalk, and B. Fitzgerald. *Ten Strategies for Successful Distributed Development*, chapter 9. The Transfer and Diffusion of Information Technology for Organizational Resilience. Springer, 2006.
- [20] G. M. Olson and J. S. Olson. Distance Matters. *Human-Computer Interaction*, 15(2&3):139–178, 2000.
- [21] D. E. Perry, N. A. Staudenmayer, and L. G. Votta. People, Organizations, and Process Improvement. *IEEE Software*, 11(4):36–45, 1994.
- [22] R. Prikładnicki, J. L. N. Audy, D. Damian, and T. C. de Oliveira. Distributed Software Development: Practices and challenges in different business strategies of offshoring and onshoring. In *Proceedings of the 2nd Intl. Conference on Global Software Engineering (ICGSE 07)*, 2007.
- [23] S. Sarker and S. Sahay. Implications of space and time for distributed work: an interpretive study of US-Norwegian systems development teams. *Eur. J. Inf. Syst.*, 13(1):3–20, 2004.
- [24] A. Sarma, Z. Noroozi, and A. van der Hoek. Palantír: Raising Awareness among Configuration Management Workspaces. In *ICSE '03: Proceedings of the 25th Intl. Conference on Software Engineering*, 2003.
- [25] S. B. Seidman. Network structure and minimum degree. *Social Networks*, 5:269–287, 1983.
- [26] S. Siegel. *Nonparametric Statistics for the Behavioral Sciences*. McGraw-Hill Companies, 1st edition, 1956.
- [27] V. Sinha, B. Sengupta, and S. Chandra. Enabling Collaboration in Distributed Requirements Management. *IEEE Software*, 23(5):52–61, 2006.
- [28] H. Spanjers, M. ter Huurne, B. Graaf, M. Lormans, D. Bendas, and R. van Solingen. Tool Support for Distributed Software Engineering. In *Proceedings of the 1st Intl. Conference on Global Software Engineering (ICGSE 06)*, 2006.
- [29] D. Spinellis. Global software development in the freeBSD project. In *GSD '06: Proceedings of the 2006 Intl. workshop on Global software development for the practitioner*, 2006.
- [30] T. Wolf. *Rationale-based Unified Software Engineering Model*. Dissertation, Technische Universität München, July 2007.