Distance, Dependencies, and Delay in a Global Collaboration

James D. Herbsleb[1], Audris Mockus[1], Thomas A. Finholt[2], Rebecca E. Grinter[3]*
[1]Bell Labs, Lucent Technologies, Naperville, IL 60563
[2]School of Information, University of Michigan, Ann Arbor, MI 48109
[3]Xerox PARC, Palo Alto, CA 94304
{jherbsleb, audris, beki}@lucent.com, finholt@umich.edu, grinter@parc.xerox.com

ABSTRACT

Collaborations over distance must contend with the loss of the rich, subtle interactions that co-located teams use to coordinate their work. Previous research has suggested that one consequence of this loss is that cross-site work will take longer than comparable single-site work. We use both survey data and data from the change management system to measure the extent of delay in a multi-site software development organization. We also measure site interdependence, differences in same-site and cross-site communication patterns, and analyze the relationship of these variables to delay. Our results show a significant relationship between delay in cross-site work and the degree to which remote colleagues are perceived to help out when workloads are heavy. This result is particularly troubling in light of the finding that workers generally believed they were as helpful to their remote colleagues as to their local colleagues. We discuss implications of our findings for collaboration technology for distributed organizations.

Keywords

Global collaboration, software development, delay, speed, interdependence, awareness, informal communication

INTRODUCTION

CSCW research, with its focus on understanding practice, has revealed the many subtleties of collaborative work. We know much more about how context, peripheral awareness, incidental properties of artifacts, and informal conversations allow us to coordinate our activities with others, and how technologies can help, or hinder those patterns. These studies share one important feature – they focus on single site collaboration, occasions where everyone is co-located.

Studies of distributed work paint a different picture [4, 25, 27]. We know from these studies that multiple site work presents considerably more challenges precisely because of the loss of the opportunities for rich interaction [24]. Evidence of how much collaboration happens through rich interactions and the fact that geographically distributed settings present more challenges for coordinating work comes at a time when more companies are globalizing.

Corporations like Lucent Technologies are becoming increasingly distributed for a variety of reasons. First, mergers and acquisitions to adjust and complement product lines often lead to new sites becoming part of the company. Second, to participate in some markets government regulations request the location of some local development operations. Third, it can make sense for market reasons to locate parts of the corporation where the market for a particular technology exists. Fourth, the competition for highly skilled technical staff is driving companies to hire them wherever in the world the talent can be found. Finally, most corporations, especially those in the software business, hope that geographic distribution could lead to round-the-clock development, which offers the promise of reducing development cycles by increasing the amount of time in the day that software is being developed. For example, the working day in India does not overlap with the working day in the United States, making it theoretically possible to get 16 or more hours of development in one day.

As development becomes increasingly distributed, little is known about how the loss of rich opportunities to interact impacts the development cycle. This paper reports findings that distributing work across multiple sites significantly impacts development life cycles, and that interdependence and communication patterns may explain the loss of speed.

For many of the reasons described above, a major software development effort at Lucent Technologies has been distributed for several years. It began with two sites in two countries in one continent. It has now grown to six primary sites located in four countries, in two continents, with an additional seventh supporting site in a third continent. This paper reports how problems with delays impacted this organization, and how the understanding achieved from this

empirical research informed the development of tools to overcome these problems.

In the remainder of this introduction, we review literature on distributed work and how it differs from co-located work. In the following section we briefly describe our empirical methods. Then, we present new results on the degree to which the work shares dependencies across sites, communication patterns across and within sites, and results showing the relationship of cross-site work and delay. Finally, we draw out the design implications of these observations, and conclude the paper.

Subtleties of face-to-face work

As mentioned above, CSCW research over the last decade has highlighted the rich and subtle interactions that occur in single-site collaborative work. For example, Heath and Luff [16] showed how operators of the London Underground developed ways of working together without specifically coordinating their efforts, relying on speaking out loud and monitoring. They illustrated this with their discussion of how the person responsible for scheduling the trains running communicated delays to the person responsible for passenger announcements. The scheduler, speaking to a train driver asked him to slow down, and the announcer overhearing this activity informed the passengers that the next train would be delayed. Monitoring activities like this have also been reported in studies of air traffic control where controllers look at the screens of their colleagues to see where their planes are coming from, and make information visible to their colleagues by annotating flight strips to signal critical information [19].

Other studies, away from the control room have highlighted similar mechanisms for interacting to resolve collaborative problems. Ackerman and Halverson's [2] study of help-line operators working together to solve an individual's request illustrates the wide variety of resources synthesized into a working organizational memory of how the problem was dealt with last time it arose. The work of the help line agents, much like the work of the purchasing agents who Suchman studied [29], shows that co-located work colleagues have opportunities for rich interactions simply because they can talk, listen, and watch each other.

Communication and distance

In contrast to the rich interaction of face-to-face work, there is very convincing evidence that the frequency of communication drops off sharply with physical separation among co-workers' offices, and that the sphere of frequent communication is surprisingly small. Tom Allen [3], in a study of engineering organizations, reported that the frequency of communication among engineers decreased with distance. Further, he noted that when engineers' offices were about 30 meters or more apart, the frequency of communication dropped to nearly the same low level as people with offices separated by many miles.

Kraut et al [20] found similar results for scientists. Further, they found that the rate at which scientists collaborated spontaneously with one another was also a function of distance between offices, and that this effect was more powerful than the effect of same-discipline scientists tending to collaborate more frequently with one another. Presumably, the more frequent communications led to conversations in which common interests were discovered and acted upon.

These findings are particularly troubling in rapidly evolving, high technology environments, where the competitors, products, standards, and customers routinely create a demand for significant, unforeseen changes in requirements throughout the development cycle. In organizations with rapidly changing environments and "unstable" projects, informal communication is particularly important [13, 21]. For example, as requirements change, it is hard for the formal mechanisms of communication, such as specification documents to react quickly enough. Often news of change is propagated informally among the development staff. Under such conditions, the pattern of lateral communication across sites should be particularly important in the environment under study.

Research showing the importance of informal com-munication has lead to a variety of technologies designed to stimulate casual conversation among workers at different sites. These technologies have included video [1, 9, 11, 23], audio [18], and text [7]. To this list we must now also add instant messaging, a technology that has spread very rapidly, and is beginning to infiltrate the work place (e.g., [28]).

These observations about communication and distance also highlight the importance of understanding the dependencies among the various kinds of work involved in software development [14]. In a study of six software engineering organizations, Grinter, Herbsleb, and Perry [15] observed four different ways of organizing work across sites that evolved within a single global corporation. Each represented an attempt to minimize requirements for cross-site communication in the context of particular types of product architectures and mechanisms for coordinating work. There are also indications that, where possible, work groups will try to reduce the coupling of cross-site work [25].

In their study of distributed student teams taking part in a realistic management game Fussell et al [12] found that e-mail was the primary medium used to coordinate activities. When there was increased e-mail exchange and more face-to-face meetings, teams were better coordinated. Additionally, better coordination and more discussion of strategy both had a positive impact on team performance. Their data also suggest that in the context of this game, monitoring the environment is more important than monitoring people or documents (see also [6]).

In a case study of a software engineering organization spread across several sites, Herbsleb and Grinter [17] found that the organization used a number of mechanisms, including plans, processes, and interface specifications, to coordinate the cross-site work. Each mechanism, however, was vulnerable to imperfect foresight and unexpected events, which required substantial communication to coordinate activities and renegotiate commitments. Despite the need for communication, there was a nearly total absence of informal, unplanned communication across sites.

The difficulties of knowing who to contact about what, of initiating contact, and of communicating effectively across sites, led to a number of serious coordination problems. Among these problems were unrecognized conflicts among the assumptions made at different sites and incorrect interpretation of communications. The most frequent consequence of cross-site problems was *delay* in the resolution of work issues. By delay, we mean the additional time it takes to resolve an issue when more than one site is involved. So, for example, if a part of the design or code needs to be changed, or if someone needs a better understanding of how some part of the product works, people at more than one site may need to be involved in information exchange, negotiation, and so on, in order to find a solution. Such issues arise very frequently in software development.

Delays in resolving work issues can slow development considerably. Issues that would typically be resolved in hours or minutes often stretched out to days or weeks in the effort to find, establish contact, and have the necessary collaborative sessions with the right people to achieve resolution.

Qualitative studies (e.g., [17]), have shown how individuals are disrupted by cross-site coordination challenges. But questions remain about the cumulative effects, for example, how distance affects the speed with which collaborative tasks are accomplished. In addition to being important research questions, these are critical pragmatic issues as businesses become more globally distributed. Speed to market has become the most critical factor for succeeding with new products [8]. In this paper, we examine the effect of distributed work on speed, and then examine a number of properties of cross-site versus same-site communication that may account for these differences. Finally, we discuss the implications of these findings for tools to address these communication issues.

Research questions
This paper reports a study of one geographically distributed organization, with particular attention to the effects of geographic distribution on delay in the development life cycle. We also examine data on the degree of interdependence of work at different sites as well as patterns and quality of communication in order to shed light on possible causes of delay.

Does cross-site work introduce delay, as compared to same-site work? Previous research suggests that working across sites introduces substantial delay because of reduced communication, difficulty in finding the right person and establishing contact, as well as having an effective collaborative session. We examine quantitative data comparing the time required for similar same-site and cross-site work.

To what degree is work at the different sites interdependent? Does interdependence diminish over time? The organization studied is typical of software engineering organizations in many respects, i.e., it is free to create and modify work arrangements only within the relatively narrow confines given by the location of skilled technical people, experts in particular disciplines, and location of physical resources. We have developed a measure of the coupling of work in order to assess the extent and trends of coupling in the cross-site work.

What differences are there between same-site and cross-site communication networks and their effectiveness? Are they stable over time? What is the relative size of local and cross-site networks? Is there a perception of greater misunderstanding of tasks, priorities, plans, and changes across sites? Given our previous work, we expect that change, and the need for new information to deal with the unexpected will be a particular problem. The problems will be exaggerated if the people in an individual's network change rapidly over time.

SITE AND METHODS
In this section we describe the sites of study, including some background on the products built. We also discuss how the work is divided among sites. We conclude with a description of the methods used to analyze and collect the data.

Sites
Geographically distributed software development is pervasive among most large technology companies, including Lucent Technologies. We chose one department of the company to study for three reasons:

First, the department was willing to host researchers and provide us with access to developers, documents, and source code. Second, they work in a complex area of telephony, where the market requirements and standards are changing rapidly. This makes coordinating the development work extremely difficult and subject to continuous change. In addition, this product competes in an aggressive market which brings extreme time pressures to development work. Third, the department has cross-site development, described below.

In this study we focus on four locations, one in the UK, one in Germany, and two in India, where the department does a large share of its development work. These sites exchange information frequently and make decisions that require cross-site synchronization. The German site had existed for

a number of years, and the people there had considerable experience working together on similar systems. However, it had not previously participated in cross-site development where parts of the product are split across sites. The UK site has existed for about three years, and thus had no existing relationships to any other Lucent site. One Indian site was also about three years old. The other was a software contractor, not actually a part of Lucent, but it had worked with the German and UK sites for several years. With the exception of having only restricted access to the Lucent intranet, the contractor site participated fully in projects, in ways indistinguishable from Lucent sites.

The department also has interactions with other divisions of the company because the product must interact with other technologies. Many of these technologies are built in the United States so the developers coordinate work with these other sites. These US sites had not previously worked together, nor had they worked before with the UK or German sites. In all cases, the collaborations span different languages, cultures, and many time zones, making them more difficult.

Methods

Our results draw on modification request and survey data.

Modification Requests.

Like many software development organizations, the department we studied used a Change Management (CM) system to organize and track its development work. CM systems organize development by providing mechanisms to ensure that developers coordinate changes they make to the software. Typically they provide mechanisms for versioning the code, and some ability to manage two or more developers making changes to the same software at the same time, in a structured way. Without a CM system these kinds of parallel changes would probably occur but without people realizing, leading to the software breaking. CM systems track development work through correlating the actual changes in the code with requests to make those alterations. By following requests it is possible to see what changes were made to which parts of the software, whether all the changes were actually made, and who made them. It is because of the organizational and tracking features of CM systems, that they present such unique opportunities to study collaborative work (see [14, 30]).

In the CM system that this development organization uses, the basic tracking unit is called the Modification Request (MR) which is a request to incorporate a specific functionality into the software. Some MRs ask for new functionality, others ask for specific problems or bugs to be fixed. All development work in the organization was done within the framework of an MR, using Sablime and ClearCase. Moreover, processes surrounding the CM system were structured to support MRs.

The software used for tracking MRs automatically collects several valuable types of data. It establishes a record for each MR of who made the request, the date the request was made (or "opened"), and each change ("delta") that is made to the code base in order to fulfill the request. For each change, it records the login of the person submitting the code, and the time, size, and date of the submission. Large, complex changes typically have many deltas, whereas small, simpler changes have only a few, or even just one.

MRs are the basic unit of work in this software development. Moreover, MRs and their equivalents in other CM systems are pervasive in most software development work. By performing straightforward calculations on the MR data, it is possible to derive several important measures [22], such as the following:

Work interval. The difference between the date of the first delta and the last delta for an MR is a good approximation of the period of time, or interval, that was required to do the work of implementing the change.

Full interval. The difference between the date the MR was opened and the date of the last delta is a somewhat longer interval. It includes the work interval and also the time to determine whether to actually implement the change, to assign a priority, to assign the work to particular individuals, and for these individuals to actually start the work.

Measuring interdependence: distributed versus co-located MRs. By looking up the locations of all the individuals (the person who opened the MR and all the developers who contributed deltas) associated with the logins, it is possible to determine if they are co-located or if they reside at more than one site. If we found more than one location among the individuals associated with the MR, we classified it as "cross-site." If we did not, we classified it as "local."

The ratio of cross-site MRs to total MRs provides an approximation of the degree of interdependence of the work among sites. Work required by a single MR must generally be closely coordinated, since all of the individual changes to the code must work together to implement a single change in functionality. Work on separate MRs is usually less tightly coupled, although there may also be code changes in separate MRs that must be coordinated. In general, a higher proportion of MRs with changes that span sites is a good indicator of how closely coupled the development work is. Since MRs also record the identities of everyone who works on them, MR analysis can also be used to give an indication of how the work of one individual is related to work of another in a given time period.

Survey

In November 1998, 117 employees located in Germany and the UK were invited to complete a Web-based questionnaire. Most of the workers were software engineers, with some managers and some administrative support personnel. In June 1999, a second administration of a similar survey was undertaken. In all, 160 employees in

Germany, UK, and two sites in India were invited to take the survey.

The first questionnaire consisted of 68 items, the second was 65. Both included questions covering demographics, patterns of communication, working relationships, communication and coordination, information exchange, and language. The respondents provided two answers for most questions: one with regard to local co-workers and the other with regard to distant co-workers. Many identical questions were included in both administrations of the survey. There were some deletions and additions, however, in order to drop questions that did not seem useful, to measure new variables, and to refine our measurements of others.

The surveys were administered in English in the UK and India. A German language version was produced using back translation techniques, and was available for German speakers. Both versions were pilot tested with members of the organization being studied.

Site	Survey 1, 1998	Survey 2, 1999
UK site	33	23
German site	41	39
India Internal	N/A	9
India Contractor	N/A	21

Table 1. Number of survey respondents by location.

Overall, 98 of 117 surveyed employees completed the first questionnaire, for a response rate of 83%[1]. Across the four sites, 160 employees were invited to participate in the second wave survey. We obtained usable responses from 96 individuals, for a response rate of 60%.[2]

RESULTS

Delay

We have two different measures of delay that allow us to compare single-site work with cross-site work and to validate different measures against each other. One measure is derived from our second survey, which included the following two questions:

> How many times in the past month was your own work delayed because you needed information, discussion, a decision, from someone at your site or another site?

> What was the average length of the delays you experienced before acquiring the needed

information, having the discussion, or being informed of the decision by the person from your site or the other site?

For each question, the respondent answered by supplying one number for "local site" and another number for "distant site." Of the 92 respondents, 39 reported at least 1 delay in the past month for the local site, and 48 reported at least one delay for the remote site. Averaged over all 92 respondents, the mean number of local delays was 2.1 delays per month, and the mean duration was .9 days. For cross-site delays, the mean number was 1.9 delays per month, and the mean duration was 2.4 days.

In order to test the significance of the differences in number and duration of local and remote delay, a paired observation t-test was performed on a square root transformation of the data[3]. The difference between the number of delays (local versus remote) was not significant (t=0.1758, df=91, not significant). The difference, however, in duration (local delays versus remote delays) was statistically significant (t=2.5079, df=91, p< 0.02). In summary, while there is no significant difference in the number of delays reported, their duration does vary significantly with delays crossing sites taking almost a day and a half longer than single site cases.

We see similar findings in the MR data. We extracted all of the single-site MRs, i.e., where everyone involved in the MR (the person who made the request and all the people who carried out the work of making the change) resided at one site, and compared them with the MRs which involved at least two sites. The results are shown in Figure 1.
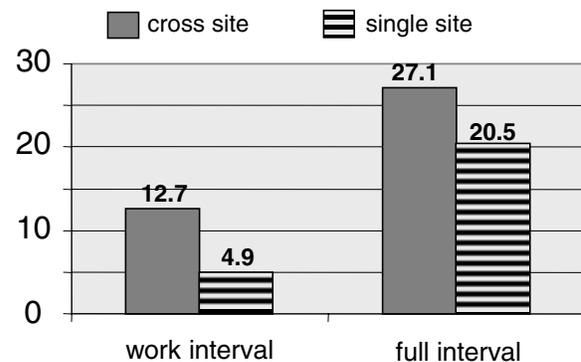

Figure 1. Average number of days to complete work on one MR. Work interval is first delta to last delta, full interval is MR open to last delta.

This figure shows that the average single-site MR took about 5 days to complete, from the time the work actually began until the last change was made (work interval). In contrast, MRs which involved more than one site took 12.7

---

[1] In this first survey, 22 of the responses were from sites we have not yet been able to visit. Because we were not certain we understood the relation of these sites to the two primary sites, these responses were eliminated from the 1998 survey.

[2] Four of the respondents reported no contact with any other site, so their data were eliminated.

[3] The scale for the delay data is truncated at zero, so the distribution is skewed, and consequently not suitable for a t-test. A square root transformation on interval produced a good approximation to the normal distribution and was used in the tests.

days, more than 2.5 times as long, to complete. If we look instead at the "full interval," i.e., the days it took to complete the request measured from the day the request was made, the difference between single-site interval (20.5 days) and distributed interval (27.1 days) is similar. (The full interval includes not only the time it takes to do the work, but also to review the request, assign it a priority, and assign the work.) The differences in interval are statistically significant ($p < 0.001$) using a t-test.

We wanted to test for the possibility that these results were distorted by MRs that were very small changes, which would be quick, simple, and require just one person, at one site. In order to test this possibility, we removed the smallest modifications (where the change was accomplished in a single delta) and compared work intervals for the remaining, relatively complex modifications. The co-located versus multi-site difference actually *increased*, from 7.8 days to 8.9 days.

The data reported in this section suggest that cross-site work carries a heavy penalty by slowing work down. In the next sections we turn to questions of how interdependent the sites are, by which we mean, how much the one site relies on another in the course of the development. We also examine the types of communications, and their patterns, within and across sites.

Interdependence of Sites
Overall interdependence. Figure 2 shows the interdependence of the work across sites, as measured by the proportion of multi-site MRs to all MRs, plotted as a function of time. Overall, about 16% of the coding occurs in MRs that are split across sites, and therefore represents tightly-coupled work. The remaining 84% is in general much less tightly-coupled.
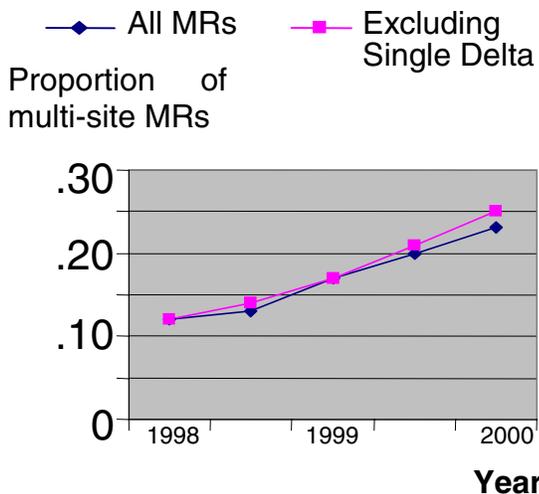

Figure 2. Proportion of multiple-site MRs over time. The lower line includes all MRs, while the upper line excludes single delta (hence potentially trivial) MRs.

As Figure 2 shows, the sites appeared to increase substantially in interdependence over time, as the proportion of (nontrivial) multi-site changes increased from about .12 to about .25. The fractions of distributed changes significantly increased over time ($p < 0.001$ using logistic regression with predictor time and response indicator of whether the MR is distributed across locations).

Size of work interdependence networks. In order to understand how developers' self-reported communication networks related to their measured work dependencies, we extracted a "per person" interdependence measure (as distinct from the overall site measure in the previous subsection) from the MR data. Each MR is associated with some set of people who either wrote the MR or contributed code. For each developer, we constructed a set that was the union (excluding that developer) of all people who worked on all the MRs that the developer worked on. The number of people in this union is a measure of the number of individuals with whom this developer shared tightly-coupled work. We counted the number of such individuals for each developer, separating those who resided at the same site as the developer, and those who resided at another site.

The results showed that during the two years we studied, for each person who appeared at least once in the MR database, the average local work interdependence was 5.4 people and the average remote interdependence was 3 people with local to remote ratio of just under 2. If we weight each interaction by frequency (number of MRs) then the weighted local interdependence is 12.6 and remote 4.7 with ratio of 2.7. This indicates that there are many more chances to "get to know" the local people through more frequent work contacts.

Communication
Size of personal networks. In order to get a rough estimate of the size of local and remote communication networks, we asked people to

> **Consider an average week. How many different people do you typically interact with at work during the course of the week from your <local, remote> site?[4]** ($t=12.4036$, $df=77$, $p<.0001$)

As we expected, the results were quite different for local and remote sites. The mean for local was 16.0, and for remote 4.9. This difference is highly significant.

How personal networks change over time. The two surveys included a "name generator" item that requested respondents to name up to 10 people he/she communicates with locally and up to 10 he/she communicates with remotely. We calculated the ratio of people each person named in both surveys to the total number of distinct people named by the person. The average ratio was 18% of the

---

[4] Again, because these data were bounded at zero, we performed the t-test on a square root transformation.

remote network and 22% of the local network, indicating a large turnover within a period of only seven months.

This figure is somewhat inflated for the local networks. The reason is that we only provided spaces on the survey for 10 remote and 10 distant colleagues. This is probably sufficient for remote networks (self-reported mean size of 4.9), but insufficient for local networks (self-reported mean size of 16). For local colleagues, the changes partly represent a difference in the samples of the total network that happened to be reported on the two occasions.

Media preferences. There was a large difference in the preferred means of (non-face-to-face) communication, depending on whether the communication occurred among people speaking the same native language versus a different native language. We asked

(S1) When face-to-face communication is not possible, I prefer to use electronic mail rather than the telephone. (t=6.05, df=73, p<.0001)

The respondents showed a marked difference, with a much greater tendency to agree when the communication was directed toward a native speaker of a *different* language.

As one might expect, we saw a comparable difference in the following question about telephone preferences:

(S1) When face-to-face communication is not possible, the telephone is less ambiguous than electronic mail. (t=4.54, df=72, p<.0001)

Here, people showed highly significant tendency to agree when the communication was with a native speaker of the *same* language. This confirms earlier qualitative observations of a similar relationship between native language and media preference [17].

Effectiveness of communication. We asked questions in order to test previous observations about differences in informal communication occurring within a site and between sites. We asked several questions about the difficulty of finding and contacting people. For all of these questions, the differences were highly statistically significant, in the expected direction, favoring local communication. (Questions labeled S1 appeared on the first survey, S2 appeared on the second; items in boldface are used in a later analysis.)

(S1) I lose time trying to figure out who to contact regarding my work. (t=4.44, df=66, p<.0001).

(S1, S2) **People I need to communicate with are difficult to find.** (S1: t=2.82, df=67, p=.006; S2: t=2.59, df=68, p=.01).

There was a significant difference in responses for remote and local sites, with a much greater tendency to believe that people at remote sites were more difficult to find and to contact.

We also asked several questions designed to assess the extent to which important information flowed through the communication network, and about inadequacies and barriers.

(S1) I often get useful work-related information through casual conversations (t=5.44, df=64, p<.0001).

(S1) There have been times when I was accidentally excluded from information which was shared by my co-workers (t=3.56, df=61, p=.0007).

(S2) **My co-workers provide timely information about changes in current plans** (t=6.3, df=91, p<.0001).

Again, the differences in responses across sites were highly significant, and in the direction suggesting better communication within a single site.

We also asked several questions in order to determine what kinds of misunderstandings may have arisen locally and across sites. Interestingly, there was little evidence for misunderstanding at the level of specific tasks. Questions probing these issues revealed that workers were no more likely to disagree about task priorities or doubt the clarity of task assignments for remote workers that for local workers:

(S2) There is disagreement about task priorities. (t=1.09, df=91, not significant).

(S2) When work is assigned, everyone is clear about his or her task. (t=1.62, df=91, not significant).

A question at the level of overall plans, however, showed a perception of significantly greater clarity in plans for local collaborations:

(S2) **Plans for future action are clearly formed** (t=2.9, df=91, p=.005).

A related issue is the response to load balancing problems, and the willingness to help a co-worker when they become overloaded. We asked about the respondents' willingness to help out when a co-worker is experiencing a particularly heavy workload:

(S2) I assist my co-workers with heavy workloads, beyond what I am required to do.

The results show only a small, nonsignificant difference in answers between local and remote (t=1.05, df=91, not significant). However, we also asked:

(S2) **My co-workers assist me with heavy workloads, beyond what they are required to do.** (t=6.26, df=91, p<.0001)

For this question, the respondents reported a sizeable, highly significant difference between the help offered by their (more helpful) local and (less helpful) remote colleagues. The contrast in these two answers is quite revealing. While people generally seem to believe they assist local and remote colleagues equally, they report that

they are much more likely to receive assistance from local than remote colleagues.

Communication and delay. In order to determine which of these differences (i.e., responses to the questions in boldface) are the most likely candidates for causing the much longer cross-site delays, we performed two linear regressions. In the first, we regressed all six variables (responses to the questions in boldface) where significant local versus cross-site differences were found. In the first, we regressed local communication variables on the square root of the length of delay for local communication. We performed an analogous regression for cross-site communication and delay variables. (We only used questions from the second survey since delay data were not collected on the first.)

In the case of local communications none of the predictors was significant. Table 2 shows the results for cross-site communication and delay. The only predictor among these cross-site variables is the response to whether co-workers assist me with heavy workloads. The higher the reported assistance, the lower the delay.

Cross-site Data	Value	Std. Error	t-value	Pr(>|t|)
(Intercept)	2.2	0.7	2.9	0.005
Coworkers assist me	**-0.3**	**0.1**	**-2.6**	**0.01**
Plans clearly formed	-0.04	0.1	-.5	*
Hard to find	-0.01	0.1	-0.09	*
Size personal network	-0.02	0.02	-1	*
Get timely information	0.09	0.1	0.8	*

Table 2. Regression of cross-site survey variables on duration of remote-site delay (indicates not significant). Only the "Coworkers assist me" item is significantly related to delay duration.*

CONCLUSIONS

Speed is perhaps the single most important success factor in modern high technology businesses. In the face of the growing globalization of all facets of work, from virtual teams to virtual corporations, the demand for speed must be accomplished, if it is to be accomplished at all, in a geographically distributed environment.

Does cross-site work introduce delay, as compared to same-site work? The answer would appear to be a resounding "yes." In this study, we have presented evidence that speed presents a challenge indeed in multi-site work. Diminished communication across distance and the loss of the subtle modes of face-to-face communication and coordination that co-located work affords, appear to have rather dramatic and unfortunate consequences. Both the survey data and the MR data point toward a significant slowdown of work that spans sites, as compared to work involving the same people that does not cross sites. The fact

that both sources of data indicate substantial cross-site delays increases our confidence in this result.

To what degree is work at the different sites interdependent? Does interdependence diminish over time? We think it is very important to begin to understand how to measure the degree of interdependence so that the effects of various levels of coupling can be understood and addressed. We need a variety of measures for different kinds of work. In this paper, we offer one measure that can be used by most software engineering organizations, i.e., those that are using CM systems such as Sablime, ClearCase, or any of a number of other commercial products.

This measure, of course, has its limitations. It is useful only for those parts of development that are subjected to change control, which generally means coding and maintenance. This may not be as great a limitation as it may first appear to be, since telecommunications and most other long-lived software systems spend most of their life in development and maintenance. Interdependencies in requirements, design, and testing may be missed, however, since many organizations (including the one in this study) do not collect relevant change data (e.g., to design documents, test cases, etc.).

The change data we presented showed an increasing trend toward site interdependence. This runs counter to other observations in the literature (e.g., [25]) where cross-site work was rearranged to make it less closely coupled. This may reflect constraints about where the relevant expertise is, the addition of new people and sites that must be given some of the work even if the coupling to other work is tighter than one would like, or it may reflect a tendency of a software architecture to deteriorate over time so as to limit the ability to isolate changes to particular parts of the product [10]

What differences are there between same-site and cross-site communication networks and their effectiveness? Are they stable over time? We identified several differences, including the size of the communication network, the difficulty of finding people, the reduced likelihood of getting timely information, the clarity of plans, and the reduced likelihood of receiving help with heavy workloads. Both remote and local social networks were extremely volatile, with about 80% change in seven months.

Interestingly, and perhaps ironically, the only factor statistically related to the length of delays was the reported lower levels of receiving help with especially heavy workloads. The irony is that this is one factor where we have data indicating that no one seems to perceive him/herself as part of the problem. In other words, people believe they are no less helpful to remote colleagues, but the same population of respondents reports they are less likely to receive help from remote, as compared to local, colleagues.

There are several possible explanations of this effect. One simple explanation is that local networks are larger, hence one receives more help locally. The fact that answers to the parallel question about help rendered by the respondent are not different for local and remote colleagues casts doubt on whether their responses are influenced by network size.

Alternatively, it may be that people attempt to be equally helpful, but help offered across sites is relatively ineffective, so the perception of the recipient is that little help has been rendered. Or it may be that it is difficult to accurately convey the urgency of a situation across sites, so the potential helper believes he/she is offering an appropriate level of help, but underestimates the problem. These possibilities have somewhat different implications for collaboration tools, since the first would require tools that help with effectively carrying out the work, while the latter places more importance on communication tools that help convey more context, and perhaps more emotional content. In any case, this is a particularly urgent problem since it is directly related to delay, and the responses suggest that people in general will see no need to take any action.

TOOLS FOR GLOBAL COLLABORATION

Our results provide strong evidence that multi-site work is associated with delay, and that the interdependence between sites is increasing. There are a number of hints in the data about the kinds of technologies that might be effective. The research reported in this paper is part of a larger, multi-year project that is developing tools in all of these areas.

Finding experts. Workers have difficulty finding the right people across sites (e.g., [2]). The extreme volatility of communication networks suggests that this will be a continuing problem, especially at remote sites, where there is relatively little sparse interaction, and correspondingly few opportunities to learn who does what, and who has what expertise, and to be aware of where they are now.

We are currently deploying a tool called Experience Browser which provides a visualization of the CM system, designed to make it easy to discover who has experience working on which parts of the code, and to get contact information for that person.

Awareness. Instant messaging applications can give some indication of whether a person is available, i.e., logged on to a server, such as AOL Instant Messenger. It also provides lightweight chat capabilities, which makes it very easy to start up informal, text-based conversation. Generally, there is strong and increasing interest in such systems (tens of millions of users according to news reports [28]). We are currently deploying a version of an IM tool called Rear View Mirror that includes functionality for work groups [5]. It also incorporates MUD-like group chat facilities, the purpose of which is to provide an ongoing, continuous cross-site discussion forum, and to avoid the intrusiveness of instant messaging which demands an immediate response from a single recipient. It also has the advantage, when used across language boundaries, of allowing people to converse in text, for which we saw a strong preference in our survey.

Shared calendars can also play a central role by letting people at other sites know where someone is, when they might be free, and even who they have been meeting with [26]. We have deployed a web-based calendar tool (which we call CalendarBot) in the development organization, and it is receiving heavy use, i.e., thousands of hits per month.

We are currently trialing a prototype web-based tool that shows international clocks, online calendars, and contact information for groups of people the user wants to stay in touch with. It will also be linked with the development organization web site that is being organized around teams. Navigating to a team's home page will automatically allow the tool to display the appropriate clocks, calendars, and person information for that team.

Richer interaction. Finally, the finding about non-help during heavy workload and delay may argue for tools supporting richer interaction, i.e., high quality audio and video, that may be more effective in conveying the nuances of expression and emotion that allow more accurate determination of urgency.

ACKNOWLEDGMENTS

REFERENCES

1. Abel, M. J., Experiences in an Exploratory Distributed Organization, in Intellectual Teamwork: Social Foundations of Cooperative Work, J. Galegher, R.E. Kraut, and C. Egido, Editors. 1990, Lawrence Erlbaum Associates: Hillsdale, New Jersey. 489-510.

2. Ackerman, M. S. and C. Halverson. Considering an Organization's Memory. In Computer Supported Collaborative Work. 1998. Seattle, WA: ACM Press 39-48.

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

4. Bellotti, V. and S. Bly. Walking Away from the Desktop Computer: Distributed Collaboration and Mobility in a Product Design Team. In ACM Conference on Computer Supported Cooperative Work CSCW '96. 1996. Cambridge, MA: New York, N.Y.: ACM Press 209-218.

5. Boyer, D. G., et al. Virtual Community Presence Awareness. In CSCW '98 Workshop on Designing Virtual Communities for Work. 1999. Seattle, WA.

6. Cadiz, J. J., et al., The Awareness Monitor: A Coordination Tool for Asynchronous, Distributed Work

Teams. Unpublished manuscript. Carnegie Mellon University, 1999.

7. Churchill, E. F. and S. Bly. It's all in the words: Supporting work activities with lightweight tools. in GROUP '99. 1999. Phoenix, AZ 40-49.

8. Cusumano, M. A. and D. B. Yoffie, Competing on Internet Time. 1998, New York, NY: The Free Press.

9. Dourish, P. and S. Bly. Portholes: Supporting Awareness in a Distributed Work Group. In ACM CHI'92 Conference on Human Factors in Computing Systems. 1992. Monterey, CA.: ACM Press 541-547.

10. Eick, S. G., et al. Does Code Decay? Assessing the Evidence from Change Management Data. IEEE Transactions on Software Engineering, in press.

11. Fish, R. S., et al. Evaluating Video as a Technology for Informal Communication. in CHI'92. 1992. Monterey, CA: ACM 37-48.

12. Fussell, S. R., et al. Coordination, overload and team performance: Effects of team communication strategies. in Computer Supported Cooperative Work. 1998. Seattle, WA: ACM Press 275-284.

13. Galbraith, J., Organizational Design. 1977, Reading, MA: Addison-Wesley.

14. Grinter, R. E. Recomposition: Putting It All Back Together Again. in ACM Conference on Computer Supported Cooperative Work (CSCW '98). 1998. Seattle, Washington: ACM Press.

15. Grinter, R. E., J. D. Herbsleb, and D. E. Perry. The Geography of Coordination: Dealing with Distance in R&D Work. In GROUP '99. 1999. Phoenix, AZ.

16. Heath, C. and P. Luff. Collaborative Activity and Technological Design: Task Coordination in London Underground Control Rooms. In ECSCW '91. 1991. Amsterdam 65-80.

17. Herbsleb, J. D. and R. E. Grinter. Splitting the Organization and Integrating the Code: Conway's Law Revisited. In 21st International Conference on Software Engineering (ICSE 99). 1999. Los Angeles, CA: ACM Press 85-95.

18. Hindus, D., et al. Thunderwire: A field study of an audio-only media space. In Computer Supported Cooperative Work. 1996. Boston, MA: ACM Press 238-247.

19. Hughes, J. A., D. Randall, and D. Shapiro. From Ethnographic Record to System Design: Some Experiences from the Field. Computer Supported Cooperative Work: An International Journal 1, 3, 1993, 123-141.

20. Kraut, R. E., C. Egido, and J. Galegher, Patterns of contact and communication in scientific research collaborations, in Intellectual Teamwork: Social Foundations of Cooperative Work, J. Galegher, R.E. Kraut, and C. Egido, Editors. 1990, Lawrence Erlbaum Associates: Hillsdale, New Jersey. 149-172.

21. Kraut, R. E. and L. A. Streeter. Coordination in Software Development. Communications of the ACM 38, 3, 1995, 69-81.

22. Mockus, A., et al., On Measurement and Analysis of Software Changes. Bell Labs Technical Report No. BL0113590-990401-06TM, Murray Hill, NJ,

23. Obata, A. and K. Sasaki. OfficeWalker: A virtual visiting system based on proxemics. in Computer Supported Cooperative Work. 1998. Seattle, WA 1-10.

24. Olson, G. M. and J. S. Olson. Distance Matters. Human-Computer Interaction, in press.

25. Olson, J. S. and S. Teasley. Groupware in the Wild: Lessons Learned from a Year of Virtual Collocation. In Computer Supported Cooperative Work. 1996. Cambridge, MA: New York, N.Y.: ACM Press 419-427.

26. Palen, L. Social, individual, and technological issues for groupware calendar systems. In CHI '99. 1999. Pittsburgh, PA.

27. Robertson, T. Cooperative Work and Lived Cognition: A Taxonomy of Embodied Interaction. In Fifth European Conference on Computer-Supported Cooperative Work ECSCW '97. 1997. Lancaster, UK: Dordrecht, Netherlands: Kluwer Academic Publishers 205-220.

28. Slatalla, M., The office meeting that never ends. New York Times, Sept. 23, 1999.

29. Suchman, L. A. Office Procedure as Practical Action: Models of Work and System Design. ACM Transactions on Office Information Systems 1, 4, 1983, 320-328.

30. Tellioglu, H. and I. Wagner. Negotiating Boundaries: Configuration Management in Software Development Teams. Computer Supported Cooperative Work: The Journal of Collaborative Computing 6, 4, 1997, 251-274.