

How do Distribution and Time Zones affect Software Development? A Case Study on Communication

Martin Nordio, H.-Christian Estler, Bertrand Meyer, Julian Tschannen
ETH Zurich, Switzerland
E-mail: `firstname.lastname@inf.ethz.ch`

Carlo Ghezzi, Elisabetta Di Nitto
Politecnico di Milano, Italy
E-mail: `{carlo.ghezzi,dinitto}@polimi.it`

Abstract—Software projects have crossed seas and continents looking for talented developers, moving from local developments to geographically distributed projects. This paper presents a case study analyzing the effect of distribution and time zones on communication in distributed projects. The study was performed in a university course during two semesters, where students developed projects jointly with teams located in ten different countries in South America, Europe, and Asia. The study compares the results of the projects distributed in two locations with projects distributed in three locations. It also analyzes projects in different time zone ranges. The initial results show that the amount of communication in projects distributed in two locations is bigger than the communication in projects distributed in three locations. We also found that projects in closer time zones have more communication than projects in farther time zones. Furthermore, we analyze the reply time for e-mails of projects distributed in different time zones, and discuss the challenges faced by the students during these projects.

Index Terms—Communication, Time zones, Distributed software development, Empirical study

I. INTRODUCTION

Geographic distribution has not imposed a barrier for today’s software industry: software projects and companies cross rivers and seas looking for talented developers. Researchers [4], [16], [14], [12], [19] have pointed out the challenges of distributed software developments, emphasizing the importance of software engineering concepts such as requirements engineering, project management, and API design. Distributing projects over several countries, continents, and time zones makes these concepts even more relevant.

To evaluate the effect of distributed software development, studies [3], [11], [13], [10] have been performed. For example, Bird et al. [3] have studied the development of Windows Vista, and compared the failures of the components developed distributed with the failures of the components developed locally. Other studies [13], [10] focus on how time zones affect software development. An interesting question to answer is how distribution and time zones affect distributed software development, specially when the teams are distributed in several countries, and continents.

In this paper, we present a case study of distributed software development performed during two semesters in a university course. In this course, “Distributed and Outsourced Software

Engineering” (DOSE), students develop software in a distributed setting, collaborating with teams in Europe, Asia, and South America. The study was performed during the editions of DOSE in 2009 and 2010. In DOSE 2009, the projects were implemented jointly with seven universities (located in six countries); in 2010 the projects were developed in collaboration with eleven universities (located in ten different countries).

The study analyzes the whole development process: from requirements, interface specification (also known as API design), to implementation and testing. We collected data from 45 teams, where each team consisted of 3-5 students.

The study focuses on the communication of distributed projects, analyzing the time expended in communication over the total time of the project. The study includes projects developed in two and three different countries, and distributed in different time zones. We compare the results of two-location projects with the results of three-location projects. We also classify the projects according to their time zone difference, and we compare the results of the projects located in closer time zone with the ones in farther time zones.

The initial result of the study shows that there is not a significant difference in the overhead in communication of the two-location projects with the three-location projects. On average, projects distributed in two locations expended about 28% of the project only on communication, while projects distributed in three locations expended about 25% of the project for communication. We also found that there is no significant difference in communication for projects with a big time zone difference (in time zones more that 9 hours apart) and the projects with medium and short time zone difference (5-7 hours apart and 0-3 hours apart respectively). While we expected to have more overhead in three-location projects and two-location projects, the outcome of the study suggests the opposite. Even when the study was developed during two semesters, and involved 45 teams, we think the result is still preliminary.

The paper is organized as follows. In Section II, we present our research questions. Section III and Section IV describes the context of the study, and how the data was collected respectively. In Section V, we present the method and the analysis of the results. We discuss the challenges we faced during the

projects in Section VI. Sections VII and Section VIII describe threats to validity and related work respectively. Section IX summarizes the results and describes future work.

II. RESEARCH QUESTIONS

Communication in distributed projects is difficult. The lack of face to face meetings and synchronous communication makes the communication harder, and it can produce misunderstandings in the teams, and delays in the projects. Conventional wisdom is that if the projects are more distributed in countries with different cultures and with big time zone differences, these projects are more challenging.

With the outsourcing phenomena, companies started to outsource to different countries. This conventional wisdom also influenced outsourcing: it is believed that outsourcing to closer locations, with similar cultures and similar time zones is less risky. Thus a new phenomena was originated known as *nearshore*. The same conventional wisdom also applies for distributed software development.

We analyze this belief in our empirical study where projects are developed in countries with different cultures and time zones. We expect that projects distributed in two locations would have lower overhead in communication than projects in three locations. If the projects take place with a bigger time zone difference, we expect to have bigger delays in reply time than in projects with smaller time zone difference. Furthermore, we expect that projects in closer time zones have lower overhead in communication than projects distributed in farther time zones. Our research questions are:

- *RQ1: Is the amount of communication in three-location projects higher than in two-location projects?*
- *RQ2: Do projects distributed in farther time zones have more communication than projects distributed in closer time zones?*
- *RQ3: Is the average reply time for e-mails of projects in farther time zones higher than the average reply time for e-mails in closer projects?*

The case study not only reports whether these questions can be answered positively or negatively, but also what the difference is between the project settings. We also analyze individually these research questions in the different phases of the development; for example we report if *RQ1* can be answered positively or negatively for the requirements phase.

III. CONTEXT OF THIS STUDY

A. The DOSE Course

The study was developed during the Fall semesters in 2009 and 2010 in the “Distributed and Outsourced Software Engineering” (DOSE) course [20], [18]. The DOSE course targets master students with good experience in programming and some prior knowledge in software engineering. Since 2007, the course incorporated a distributed project. In this paper, we present the case study developed during the last two years; the collected data in the previous years is not included because it is incomplete and not representative.

In DOSE 2009 [8], the projects were done in collaboration with seven universities:

- 1) ETH Zurich, Switzerland
- 2) Hanoi University of Science and Technology, Vietnam
- 3) Odessa Polytechnic National University, Ukraine
- 4) Politecnico di Milano, Italy
- 5) State University of Nizhny Novgorod, Russia
- 6) University of Debrecen, Hungary, and
- 7) University of Zurich, Switzerland

In DOSE 2010 [9], the projects were done in collaboration with eleven universities. Besides the universities mentioned above, the following universities participated in the course:

- 1) Korea Advanced Institute of Science and Technology, Republic of Korea
- 2) University of Delhi, India
- 3) University of Rio Cuarto, Argentina, and
- 4) Wuhan University, China

B. The Projects

In DOSE 2009, the project consisted of a game platform for networked multi-player card games where players could log in and choose a game to play. The project consisted of eight subcomponents (each subcomponent developed by one group). A group consisted of two teams located in two different countries. For each game, one team implemented the logic of the game and the other team implemented the graphical user interface (GUI) and the network communication.

In DOSE 2010, the project was a learning platform to help users learn languages such as English, Spanish, German, etc. The project had eleven subcomponents (each subcomponent developed by one group). A group consisted of three teams located in three different countries. For each language component, one team implemented the logic of the component, one team the GUI, and one team the persistence layer.

The projects in 2009 and 2010 were organized in the following four phases:

- Phase 1: Scope document (2 weeks)
- Phase 2: Requirements (2 weeks)
- Phase 3: Interface specification (2 week)
- Phase 4: Implementation and Testing (6 weeks)

In the first week of the course, we provided a scope document describing the general architecture of the project. In Phase 1, the students developed their own scope document for their subcomponent describing the scope of each project, and the role of each student in the team. These scope documents helped the students avoid misunderstandings regarding the scope of each subcomponent. In the requirements phase, the students wrote a complete requirements specification document focusing on the description of the functional requirements. The interface specification were written in Eiffel using contracts (our previous work [21] shows a study of the use of contracts in distributed projects). The last phase was the implementation and testing of the system. The implementation was built on top of the interface specification and was done in Eiffel.

C. Projects' Outcomes

The outcomes of the projects were good.

In DOSE 2009, the eight games were fully implemented. All together, they consisted of 55'000 lines of code; each group implementation had an average of about 37 classes and 6'800 lines of code. In 2010 the eleven language modules were fully implemented, and consisted of 130'000 lines of code; on average, each subcomponent had 55 classes and 12'000 lines of code.

IV. DATA COLLECTION

The data used for the study has been collected during the two editions of the DOSE course in 2009 and 2010. The students reported for each phase of the project (scope document, requirements, interface specification, and implementation and testing) how much time they spent working on the project, how much time they used for communication, and the average reply time for e-mails. Each team collected this data independently and it was sent to us. The teams did not have access to the data of the other teams, so the data has not been influenced by the reply of other teams. All the data we have collected is available for download [7].

We also required the students to use a mailing list for the e-mail communication and to commit the chat logs into the repository. Using these archives, we corroborated the data given by the students.

We classify the projects by two criteria: number of countries working together in a group, and largest distance of time zones of these countries. The classification for number of countries is:

- **Two locations:** Projects geographically distributed in two countries, with one team per country. The project consisted of two subcomponents, each team implemented one subcomponent. We have collected information from 16 teams participating in the course (50 students). Typical group configurations were:
 - Switzerland - Italy
 - Switzerland - Ukraine
 - Italy - Hungary
 - Switzerland - Vietnam, etc

This data was collected in DOSE 2009.

- **Three locations:** Projects geographically distributed in three countries, with one team per country. The project consisted of three subcomponents, each team implemented one subcomponent. We have collected information from 30 teams participating in the course (103 students). Typical group configurations were:
 - Argentina - Switzerland - Vietnam
 - India - Switzerland - Hungary
 - Korea - Italy - Switzerland, etc

We collected this data in DOSE 2010.

For the time zone difference, we use the data collected during the edition of DOSE in 2010. The time zone differences are classified as follows:

- **Large:** Projects distributed in time zones more than 9 hours apart; they include teams in South America, Europe, and Asia. The group configurations were:
 - Argentina - Switzerland - Korea
 - Argentina - Italy - Vietnam
 - Argentina - Italy - China
 - Argentina - Switzerland - Vietnam

From the 30 teams, 12 teams had a large time zone range.

- **Medium:** Projects distributed in time zones more than 4 hours but less than 8 hours apart. They include teams in Europe and Asia, with group configurations:
 - Switzerland - Hungary - China
 - Switzerland - Russia - China
 - Switzerland - Hungary - India
 - Switzerland - Italy - Korea.

In this time zone range, there were 12 teams out of 30.

- **Small:** Projects distributed in time zones less than 4 hours apart; it includes teams in Europe. The group configurations were:
 - Switzerland - Italy - Ukraine
 - Switzerland - Hungary - Russia

We collected data from 6 teams. At the beginning of the study we also had another group in this category; however, we excluded the data because it is incomplete.

Figure 1 and Figure 2 present the geographic distribution of the students participating in DOSE 2009 and DOSE 2010 respectively.

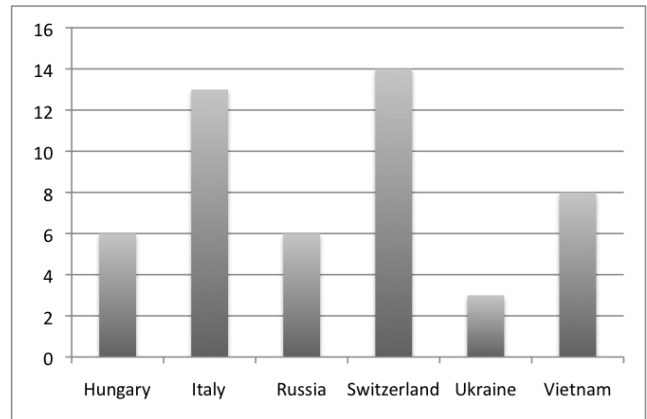


Fig. 1. Geographic distribution of students participating in DOSE 2009.

V. RESULTS

A. Analysis

To perform the analysis, we first classified the collected data by time zone ranges, as well as number of locations of the projects. For each phase in the development process (scope document, requirements, interface specification, and implementation and testing), and for each team, we collected the overall total time expended by the teams on the project as well as the time expended only for communication due to

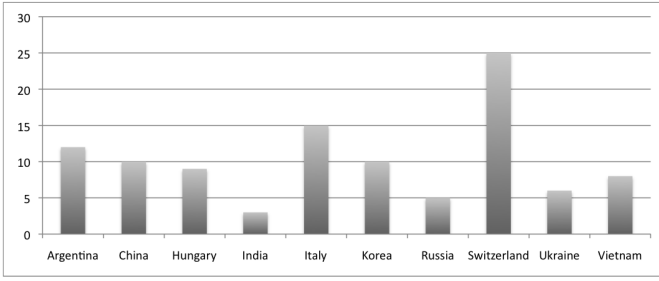


Fig. 2. Geographic distribution of students participating in DOSE 2010.

geographic distribution. Since teams had different sizes (3-4 students), we divided these numbers by the respective team sizes, obtaining the average of hours/person for all the teams and all phases.

Aside from calculating the average values for the different phases and time zone ranges, we also performed a *t-test* in order to check if the observed differences are statistically significant. We applied a significance level of $\alpha = 0.05$; i.e. an observed difference is significant with a probability of 95% if the corresponding *p*-value— shown in the tables below—is less than 0.05.

In the following subsections, we present the result of the study for each research question.

B. Communication in two-location and three-location Projects

We analyzed *RQI* by looking at the ratio of the time expended on communication due to distribution over the total time expended in the project. We use the ratio, and not the absolute number, to account for the different motivations and different experiences of the students. Students who spent more time on communication typically also spent more time on other tasks such as writing the requirements documents, implementing the system, etc. Using the ratio reduces the resulting differences and allows for a better comparison of the different data points.

We compare the data collected in DOSE 2009 and DOSE 2010. Since in DOSE 2009, teams were only distributed in *medium* and *small* time zone ranges, we omitted the data collected in the *large* time zone range for DOSE 2010. We compared four teams from DOSE 2009 with twelve teams from DOSE 2010 for the *medium* time zone range; for the *small* time zone range, we compared twelve teams in DOSE 2009 with six teams in DOSE 2010.

We formulated the null hypotheses $H_{1.1}, H_{1.2}$ and their alternative hypotheses $\overline{H}_{1.1}, \overline{H}_{1.2}$ for a two-tailed t-test to check if differences between the ratio values are significant. Let $C_{i,j}$ denote the expected average communication ratio in projects distributed in *i* locations within a *j* time zone range, where *i* is *two* or *three*, and *j* is *medium*, or *small*. For each development phase, and also for the average of the four phases, we define the following hypotheses:

$$H_{1.1} \triangleq C_{three,medium} = C_{two,medium}$$

$$\overline{H}_{1.1} \triangleq C_{three,medium} \neq C_{two,medium}$$

$$H_{1.2} \triangleq C_{three,small} = C_{two,small}$$

$$\overline{H}_{1.2} \triangleq C_{three,small} \neq C_{two,small}$$

Table I shows the ratio average of the *medium* and *small* time zone ranges for two-location and three-location projects. The results indicate projects with less distribution and smaller time zone ranges have higher communication. This tendency is also present in the results of most of the individual phases (phases 1-4). Although this outcome does not correspond with our initial expectations, it has been described in other studies [1], [15], [13] that distance and delay can negatively affect communication. We think that the reason for this reduction in the communication of three-location projects is that communication is more complex, and therefore it is reduced to the more important issues.

	Ph. 1	Ph. 2	Ph. 3	Ph. 4	Avg.
Medium (three-loc.)	31.3%	20.1%	22.3%	18.1%	22.9%
Medium (two-loc.)	33.2%	29.1%	15.8%	18.8%	24.2%
Small (three-loc.)	39.3%	18.1%	28.7%	22.8%	27.2%
Small (two-loc.)	48.2%	27.6%	33.0%	18.3%	31.8%

TABLE I

COMMUNICATION TIME COMPARISON OF MEDIUM TIME ZONE RANGE AND SHORT TIME ZONE RANGE FOR THREE-LOCATION AND TWO-LOCATION PROJECTS (RESEARCH QUESTION 1).

The results of the t-test, shown in Table II, reveal that the differences between the three-location and two-location projects are not significant. Thus, we cannot reject the null hypotheses $H_{1.1}$ and $H_{1.2}$, in favor of $\overline{H}_{1.1}$ and $\overline{H}_{1.2}$, respectively. It should be noted, however, that the number of available data points is rather low (only four values for DOSE 2009 with *medium* time zone range).

	Ph. 1	Ph. 2	Ph. 3	Ph. 4	Avg.
Medium (three- vs. two-loc.)	0.57	0.72	0.13	0.58	0.93
Small (three- vs. two-loc.)	0.10	0.50	0.69	0.17	0.33

TABLE II

p-VALUES FOR RESEARCH QUESTION 1: RESULTS OF A T-TEST COMPARING THE COMMUNICATION TIME BETWEEN THREE-LOCATION AND TWO-LOCATION PROJECTS.

To answer research question *RQI*, we use the results from Table I and Table II. The validity of a conclusion is threatened by the fact that our findings are not significant. Furthermore, the amount of data points available for the analysis is limited. Given that we cannot reject the null hypothesis, it is possible that any of the differences we observed in Table I are due to chance. The trend in Table I, however, indicates that communication gets reduced with more distribution; the opposite from what we expected to find. Thus, taking into account that we could not find a significant differences in communication but observed a result contrary to our assumption, we conclude

that *RQ1* should be answered negatively. As future work, we plan to extend the study and collect more data in the next editions of DOSE.

C. Communication in Projects in Different Time Zones

To answer *RQ2*, we use the communication data collected in DOSE 2010, i.e. twelve data points for the *large* time zones range, twelve data points for the *medium* time zone range and six data points for the *small* time zone range. The ratios of communication time to overall project time is shown in Table III. In this table, we observe a trend—against our initial expectation—towards more communication in smaller time zone ranges. The only phase that this trend does not occur is Phase 4 (the values for Phase 2 are all very close, with a difference of only 1%).

	Ph. 1	Ph. 2	Ph. 3	Ph. 4	Avg.
Large	22.8%	19.0%	17.1%	26.5%	21.3%
Medium	31.3%	20.1%	22.3%	18.1%	22.9%
Small	39.3%	18.1%	28.7%	22.8%	27.2%

TABLE III

AVERAGE RATIO OF COMMUNICATION TIME IN THE TIME ZONE RANGES (DOSE 2010).

The t-tests to check if the differences are statistically significant uses the null hypotheses $H_{2.1}, H_{2.2}$, and $H_{2.3}$ and alternative hypotheses $\overline{H}_{2.1}, \overline{H}_{2.2}$, and $\overline{H}_{2.3}$. Let C_j denote the expected communication ratio in a j time zone range, where j is *large, medium, or small*:

$$H_{2.1} \triangleq C_{large} = C_{medium}$$

$$\overline{H}_{2.1} \triangleq C_{large} \neq C_{medium}$$

$$H_{2.2} \triangleq C_{medium} = C_{small}$$

$$\overline{H}_{2.2} \triangleq C_{medium} \neq C_{small}$$

$$H_{2.3} \triangleq C_{large} = C_{small}$$

$$\overline{H}_{2.3} \triangleq C_{large} \neq C_{small}$$

The results of the test are shown in Table IV. As we chose a significance level of $\alpha = 0.05$, it is not possible to reject the null hypotheses. Therefore, despite that the data of Table III indicates a trend towards less communication in farther time zone ranges, the differences might be random. Based on the trend in the data and the non-rejection of the null hypothesis, we conclude that *RQ2* is answered negatively.

	Ph. 1	Ph. 2	Ph. 3	Ph. 4	Avg.
Large - Medium	0.39	0.28	0.23	0.13	0.60
Medium - Small	0.23	0.31	0.66	0.36	0.38
Large - Small	0.13	0.98	0.23	0.66	0.81

TABLE IV

p-VALUES FOR $H_{2.1}, H_{2.2}$ AND $H_{2.3}$: RESULTS OF A T-TEST COMPARING THE COMMUNICATION TIME FROM DIFFERENT TIME ZONE RANGES.

D. Reply Time of Projects in Different Time Zones

The third research question, *RQ3*, concerns the average reply time of e-mails. We also distinguished between the four phases and calculated the overall average. The values shown in Table V are the average numbers of hours it took for a student to get an answer from their group members. The data presented was collected during DOSE 2010, i.e. twelve data points for the *large* time zone range, twelve data points for the *medium* time zone range and six data points for the *small* time zone range.

	Ph. 1	Ph. 2	Ph. 3	Ph. 4	Avg.
Large	12.1	9.1	7.8	14.2	10.8
Medium	10.0	5.6	4.7	8.4	7.2
Small	5.2	5.3	6.3	7.2	6.0

TABLE V

AVERAGE RESPONSE TIME FOR E-MAILS IN HOURS (RESEARCH QUESTION 3).

The absolute numbers indicate, as we expected, a longer response time in the larger time zones. This result is present in most of the phases, and the difference in the different time zone ranges are similar for most of phases.

Let R_j denote the expected average reply time of e-mails in a j time zone range, where j is *large, medium, or small*. The hypotheses are defined as follows:

$$H_{3.1} \triangleq R_{large} = R_{medium}$$

$$\overline{H}_{3.1} \triangleq R_{large} \neq R_{medium}$$

$$H_{3.2} \triangleq R_{medium} = R_{small}$$

$$\overline{H}_{3.2} \triangleq R_{medium} \neq R_{small}$$

$$H_{3.3} \triangleq R_{large} = R_{small}$$

$$\overline{H}_{3.3} \triangleq R_{large} \neq R_{small}$$

Testing the significance of these differences under these hypotheses shows—in Table VI—that, in most cases, the differences in response time are not significant. We thus cannot reject the null hypotheses and, despite the trend observed in the data, answer *RQ3* negatively.

	Ph. 1	Ph. 2	Ph. 3	Ph. 4	Avg.
Large - Medium	0.60	0.16	0.31	0.15	0.17
Medium - Small	0.31	0.88	0.48	0.72	0.25
Large - Small	0.03	0.25	0.75	0.25	0.12

TABLE VI

p-VALUES FOR $H_{3.1}, H_{3.2}$, AND $H_{3.3}$: RESULTS OF THE T-TEST ON THE AVERAGE RESPONSE TIME FOR E-MAILS.

VI. DISCUSSION

In this section, we discuss the experience of DOSE, analyzing other aspects of the projects such as the project

organization, the communication in the groups, the number of commits, and the students' feedback.

A. Project Organization

The projects in the editions of DOSE 2009 and 2010 were organized in groups of two and three teams. Each team has developed its own scope document and requirements document, and each team has implemented its own requirements. The code ownership is on the level of teams. Thus, the teams were not allowed to change the code of another team. This organization simplified the projects, reducing the communication with other teams, and making the development more efficient.

We also strongly suggested a consistent use of tools such as configuration management tools, issue tracker, wiki pages and google docs. The students also used the same development environment, and we shared the projects settings in the repository to avoid errors due to different setups.

This project organization was defined at the beginning of the projects in both DOSE 2009 and DOSE 2010. We think that this project organization contributed to reduce the overhead in communication, and to get similar outcomes in the different projects.

B. Communication

In DOSE 2010, communication was performed through three main means: Skype chats, Skype voice calls, and e-mails. All the e-mails were sent to a mailing list and were archived. The Skype logs were committed to the repository. Students reported that they preferred Skype chats or e-mail communication over voice communication. They preferred this way of communication because it was easier to communicate as some students have strong accents or their English knowledge was not so good.

Figure 3 shows the average of the volume of the Skype chats and e-mails classified by time zone ranges (the volume is in KB of the messages sent). While we expected to have a higher volume for the *large* time zone range, the figure shows a similar volume for the *large* and the *short* time zone ranges; the *medium* time zone range has the highest volume, however, the difference is not significant. The *large* and *medium* time zones ranges have more communication using e-mails, while the *short* time zone range has more communication using Skype. We think the reason for this is that in the *small* time zone range the team members can set up meetings easier than in the *large* time zone range, due to the availability of the team members.

C. Commits

We have analyzed the number of commits done by the teams in DOSE 2010, classifying them by the time zones ranges. Figure 4 presents the number of commits done in each phase, and it shows that, on average, there is no significant difference in the number of commits in the different time zones ranges. The distribution for the first three phases is similar. However, for Phase 4, there is an increase of the number of commits in the *large* time zone range.

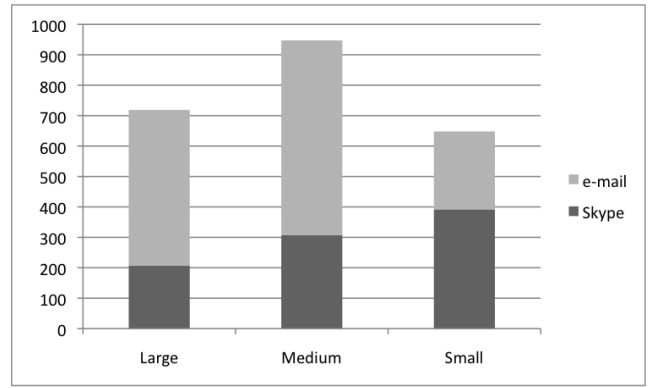


Fig. 3. Volume of the Skype chats and e-mails in KB classified by time zones (DOSE 2010)

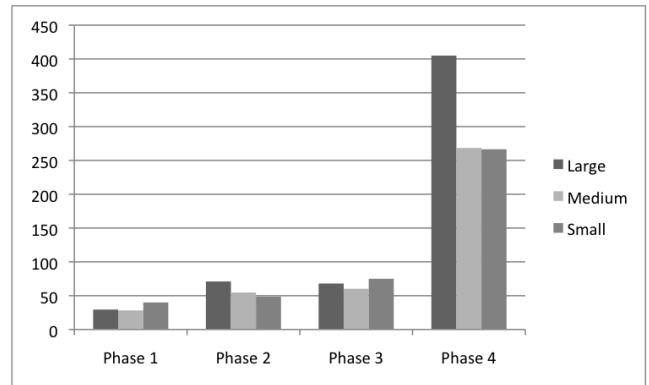


Fig. 4. Number of commits per phase classified by time zone ranges (DOSE 2010)

D. Students' Feedback

At the end of DOSE 2010, we asked the students to fill in a questionnaire. The goal of the questionnaire was to get the feedback of the students and their opinion of the effect of time zones in distributed software development. Students also reported whether cultural differences affected the development. The questionnaire was filled in by 90 students out of 103.

Table VII summarizes the result of the questionnaire. On average, the feedback reports that neither time zones nor cultural differences have affected quality, productivity, and have not caused communication overhead. The values were from 1 to 5, where 1 represents *not at all*, and 5 *very much*. The feedback is classified by time zone ranges. The result follows an order where the *small* time zone range has a lower value, and the *large* time zone range has a higher value. The values for the *small* time zone range are lower because the time zone difference is smaller, and there is not a big cultural difference between teams in Europe.

We also asked the students how they think the project development would be affected, if the project were developed by the same teams but all teams being in the same location. The last four entries in Table VII shows the result of these questions. There is not significant difference in the reply of

	Large	Medium	Small	Average
Time zones affected quality	2.6	2.0	1.4	2.1
Time zones affected productivity	3.1	2.5	1.4	2.5
Time zones caused communication overhead	3.2	2.6	1.7	2.6
Cultural differences affected quality	2.2	2.0	1.9	2.1
Cultural differences affected productivity	2.3	2.2	1.8	2.1
Cultural differences caused communication overhead	2.5	2.6	1.8	2.4
Local projects: the development would be easier	4.1	3.7	4.1	4.0
Local projects: the quality would be better	3.7	3.1	4.0	3.6
Local projects: the productivity would be higher	4.1	3.3	4.0	3.8
Local projects: the communication overhead would be lower	4.0	3.7	4.2	3.9

TABLE VII

STUDENTS' FEEDBACK FOR THE EFFECT OF TIME ZONES AND CULTURAL DIFFERENCES IN DOSE 2010. VALUES RANGE FROM 1 TO 5: 1 NOT AT ALL; 5 VERY MUCH

the different time zones. The result indicates that the students assume the development would be easier, the quality better, the productivity higher, and the overhead of communication would be lower. As future work, we plan to study these aspects, and compare local development with distributed development.

The feedback also included a question about the students preference to set up a new software projects: 35% reported that they would prefer a local project, 5% would prefer a distributed project, and 60% reported that any kind of project would be fine.

VII. THREATS TO VALIDITY

We distinguish between internal and external threats to validity. Internal validity refers to whether the study supports the findings. External validity refers to whether the findings can be generalized.

A. Internal Validity

The results of the analysis have shown that our data is not sufficient to be able to answer to the research question. Although the trend in the data confirms, for example, that the amount of communication in three-location projects is not higher than in two-location projects, we would need to collect additional data in order to draw strong evidence-based conclusions. Furthermore, the raw data collected are numbers that students reported (at the end of the different phases) to their teaching assistants. This presents a threat to validity, as student might (unwillingly) provide us with wrong numbers. We have minimized this threat by checking the reported numbers on communication against the available logs from chats and mailing lists. However, for certain numbers like *overall time spend on the project*, we have no means of verifying the data.

Another threat results from the fact that the projects implemented in two-location (DOSE 2009) and three-location (DOSE 2010) are different. While the structure and the complexity of the projects are similar, the different projects might cause undesired differences in the data.

B. External Validity

As our findings are in line with the results of other researchers, it can be assumed that the DOSE course is an appropriate setup to study the effects of distributed software development. It should be noted, however, that the motivation of students, and the processes they use to develop software, are likely to be different from those of people working in industry. Thus, it should be further investigated whether or not the findings of a study in an academic setting can be directly transferred to industry as well.

A more specific threat concerns the data from DOSE 2010: all the groups in the *large* time zone range included a team from Argentina. To get more general results, the study should include teams in different countries for all time zone ranges.

VIII. RELATED WORK

The challenges and effects of distributed development have been subject to research for more than two decades, for example analyzing how distributed development influences the quality of software products and the productivity in the development process. Bird et al. [3] present a case study on the development of Windows Vista, comparing the failures of the components developed distributed with the failures of the components developed locally. They have found no difference in the failures. In their study, a considerable number of developers were located in the same campus (in the same city). Our study was done using a totally distributed project, and focuses only on the communication aspects of the distribution. As future work, we plan to extend the study to also analyze the failures produced in these projects.

Ramasubbu et al. [22] also study the influence of distribution on quality. They develop a model of software development, and then use the model to understand the consequences of distribution on performance. In contrast to Bird et al.'s work [3], they found that distribution significantly reduces productivity and affects quality.

A study by Spinellis [23] examined how distributed development affects defect density, code style and productivity in the open source project FreeBSD. The analysis revealed that

there is almost no correlation between geographic distribution and defect density or productivity.

Similar to our study, other researchers have investigated the effect of distribution on communication. Allen [1] reported that the frequency of communication amongst engineers whose offices are more than 30 meters apart dropped to almost the same level of those by engineers with offices separated by several miles. Comparable results were reported by Kraut et al. [15].

Carmel [4] identifies *loss of communication* as one of four major risk factors which can lead to the failure of global software projects. Carmel and Agarwal [5] proposed to reduce intensive collaboration, temporal distance and national and organizational distance to mitigate the problems of distributed development. An experience report by Battin et al. [2] about a large-scale distributed project at Motorola also emphasizes the importance of establishing good relationships amongst the distributed teams.

A study by Herbsleb and Mockus [11] analyzed the effects of globally distributed development with regard to resolving modification requests and differences in communication. They found that on average it takes 2.5 times as long to complete distributed work items. When accounting for factors like number of people working on a modification request or size of change, however, the differences were no longer significant. The analysis of distributed communication revealed that developers communicate much more frequently with three co-located colleagues than with their remote colleagues. Also, the size of the social network (number of colleagues a developer interacts with) were significantly smaller in the distributed case. Similar to Herbsleb and Mockus' study, we have observed that the communication in two-location projects was higher than the communication in three-location projects. The same result applies to the time zone ranges. As future work, we plan to compare the distributed projects with projects developed locally.

Nguyen et al. [17] report on an empirical study of IBM's Jazz project which was developed globally distributed at five different sites. The study examined the effects of communication delay as well as task completion time. They found that distance does not have a strong effect on either one. It is noteworthy, however, that the analyzed communication was restricted to comments on work items and did not include other media such as e-mail, chat or voice. Our study was performed in an academic environment, however, it analyzes the amount of communication, especially e-mail, chat and voice, in the whole development process (from requirements to testing) for projects with different distributions.

To analyze the effects of time zone ranges, Espinosa et al. [10] conducted a study using several identical projects. They performed a laboratory experiment in which groups of two subjects had to collaboratively solve a task; they collected data by surveying students. The task to solve was, however, comparably small (at most one hour work time was given) and not a software task. Time zone ranges were mimicked by introducing work time overlaps (zero, one-third, two-third,

full) between the two subjects. Espinosa et al. studied how the time zone ranges affected production speed and production quality only. While an experiment in such a controlled environment reduces the threats to validity, it is questionable if it could be used to study the effects of time zone ranges on communication as they are experienced in software projects ranging over several months.

Another controlled experiment was performed by van Solingen et al. [24] to study the impact of the number of sites on the overall working speed. The experiment consisted of two to four sites. They have reported that when the number of sites increases, the overall working speed of sites increases. Deshpande et al. [6] present a study, based on interviews, on the effect of cultural differences in distributed projects.

IX. CONCLUSIONS AND FUTURE WORK

In this paper, we have presented a case study on the communication of distributed projects. The study shows that there is no significant difference in the amount of communication of two-location projects and three-location projects. The results show a trend that the amount of communication in two-location projects is higher than the amount of communication in three-location projects.

The study also analyzes the effect of time zone differences classifying the projects in three time ranges: *large*, *medium*, and *small*. The result also shows no significant difference in the communication of projects in these time zone ranges; however, the data shows a trend towards more communication in the *small* time zone range.

We also analyzed the reply time for e-mails in projects located in different time zones. We found that in projects located in the *small* time zone range, the reply time for e-mails was faster than in projects located in the *large* time zone range.

Our results indicate a trend on the effect of communication, however, the analysis reveals that the differences are not significant. As future work, we plan to collect more data in the next editions of DOSE in order to have more reliable data.

We also plan to extend the study and compare the distributed projects with projects developed in a single location. Additionally, we want to study the quality of the produced software and compare the number of failures in local projects to two-location projects and three-location projects. For future studies, we will keep the classification by time zone ranges.

Acknowledgments: We would like to thank all the people involved in DOSE: Do Le Minh, Franco Brusatti, Giordano Tamburrelli, Huynh Quyet Thang, Lajos Kollar, Mei Tang, Natalia Komlevaja, Nazareno Aguirre, Peter Kolb, Raffaella Mirandola, Sergey Karpenko, Sungwon Kang, Victor Krisilov, Viktor Gergel; and the students who took the course.

REFERENCES

- [1] T. J. Allen. *Managing the Flow of Technology*. MIT Press, 1977.
- [2] R. D. Battin, R. Crocker, J. Kreidler, and K. Subramanian. Leveraging resources in global software development. *IEEE Software*, 18:70–77, 2001.

- [3] C. Bird, N. Nagappan, P. Devanbu, H. Gall, and B. Murphy. Does distributed development affect software quality? An empirical case study of Windows Vista. In *Proceedings of the 31st International Conference on Software Engineering*, ICSE '09, pages 518–528, Washington, DC, USA, 2009. IEEE Computer Society.
- [4] E. Carmel. *Global software teams: collaborating across borders and time zones*. Prentice Hall PTR, Upper Saddle River, NJ, USA, 1999.
- [5] E. Carmel and R. Agarwal. Tactical approaches for alleviating distance in global software development. *IEEE Softw.*, 18:22–29, March 2001.
- [6] S. Deshpande, I. Richardson, V. Casey, and S. Beecham. Culture in global software development - a weakness or strength? In *Global Software Engineering (ICGSE), 2010 5th IEEE International Conference on*, pages 67–76, 2010.
- [7] Data collected in DOSE 2009 and DOSE 2010. <http://se.inf.ethz.ch/people/nordio/dose2010/>.
- [8] DOSE 2009. <http://se.inf.ethz.ch/teaching/2009-H/dose-0273/index.html>.
- [9] DOSE 2010. <http://se.inf.ethz.ch/teaching/2010-H/dose-0273/index.html>.
- [10] J. A. Espinosa, N. Nan, and E. Carmel. Do Gradations of Time Zone Separation Make a Difference in Performance? A First Laboratory Study. In *International Conference on Global Software Engineering (ICGSE 2007)*, pages 12–22. IEEE, Aug. 2007.
- [11] J. 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. Herbsleb and D. Moitra. Global software development. *Software, IEEE*, 18(2), 2001.
- [13] J. D. Herbsleb, A. Mockus, T. A. Finholt, and R. E. Grinter. Distance, dependencies, and delay in a global collaboration. In *Proceedings of the 2000 ACM conference on Computer supported cooperative work, CSCW '00*, pages 319–328, New York, NY, USA, 2000. ACM.
- [14] H. Holmstrom, E. O. Conchuir, P. J. Agerfalk, and B. Fitzgerald. Global software development challenges: A case study on temporal, geographical and socio-cultural distance. In *Global Software Engineering, 2006. ICGSE '06. International Conference on*, pages 3–11, 2006.
- [15] R. E. Kraut, C. Egidio, and J. Galegher. Patterns of contact and communication in scientific research collaborations. In J. Galegher, R. E. Kraut, and C. Egidio, editors, *Intellectual teamwork: Social and technological bases for cooperative work*, pages 149–171. L. Erlbaum Associates Inc., Hillsdale, NJ, USA, 1990.
- [16] B. Meyer. The unspoken revolution in software engineering. *IEEE Computer*, 39(1):121–124, 2006.
- [17] T. Nguyen, T. Wolf, and D. Damian. Global Software Development and Delay: Does Distance Still Matter? *2008 IEEE International Conference on Global Software Engineering*, pages 45–54, Aug. 2008.
- [18] M. Nordio, C. Ghezzi, B. Meyer, E. D. Nitto, G. Tamburrelli, J. Tschanen, N. Aguirre, and V. Kulkarni. Teaching software engineering using globally distributed projects: the dose course. In *Collaborative Teaching of Globally Distributed Software Development - Community Building Workshop (CTGDSD)*, New York, NY, USA, 2011. ACM.
- [19] M. Nordio, M. Joseph, B. Meyer, and A. Terekhov. Software Engineering Approaches for Outsourced and Offshore Development, 4th International Conference, St. Petersburg, Russia. Lecture Notes in Business Information Processing 54, Springer-Verlag, 2010.
- [20] M. Nordio, R. Mitin, and B. Meyer. Advanced hands-on training for distributed and outsourced software engineering. In *ICSE '10: Proceedings of the 32nd ACM/IEEE International Conference on Software Engineering*, 2010.
- [21] M. Nordio, R. Mitin, B. Meyer, C. Ghezzi, E. D. Nitto, and G. Tamburrelli. The Role of Contracts in Distributed Development. In *Software Engineering Approaches for Offshore and Outsourced Development*, 2009.
- [22] N. Ramasubbu and R. Balan. Globally Distributed Software Development Project Performance: An Empirical Analysis. In *Proceedings of the 6th joint meeting of the European software engineering conference and the ACM SIGSOFT symposium on The foundations of software engineering*, pages 125–134. ACM, 2007.
- [23] D. Spinellis. Global Software Development in the FreeBSD Project. *Proceedings of the 2006 international workshop on Global software development for the practitioner - GSD '06*, page 73, 2006.
- [24] R. van Solingen and M. Valkema. The impact of number of sites in a follow the sun setting on the actual and perceived working speed and accuracy: A controlled experiment. *International Conference on Global Software Engineering*, pages 165–174, 2010.