On the Development of a Theoretical Model of the Impact of Trust in the Performance of Distributed Software Projects

Sabrina Marczak, Vanessa Gomes
Faculdade de Informática
Pontifícia Universidade Católica do Rio Grande do Sul (PUCRS)
Porto Alegre, RS, Brazil
sabrina.marczak@pucrs.br, vanessa.gomes@acad.pucrs.br

***Abstract*—Trust is often defined as the belief that the trustee will meet the positive expectations of the trustor. Although several studies have discussed the topic, little is still known about the impact of trust (or lack of it) in the performance of distributed software projects. In this paper we present initial findings of an empirically informed study that aimed to identify which factors influence positively or negatively one's perceived trustworthiness of others in the project and the impact of such factors on specific project performance measures. Availability, competence, expertise, face-to-face communication, and leadership are among the factors considered to positively influence the development of trust and the consequent achievement of performance metrics. This is a first step on a larger investigation aiming to develop a theoretical model of the impact of trust in the performance of distributed software projects. Such a model can be used by researchers as a reference framework to further investigate the topic and by practitioners to better manage and organize distributed software teams.**

***Index Terms*—Distributed software development, trust, trust influential factors, project performance, empirical study, theoretical model**

I. Introduction

Trust is a topic of interest in several disciplines, such as psychology, sociology, and computer science, and as such can be examined at different levels—individual, team, and institutional, for example, and defined accordingly. One of the most common definitions for trust is *the belief that the trustee will meet the positive expectations of the trustor* [1]. These expectations form as a results of actions and behaviors, and can influence the level of trust as a member poses in another.

The importance of trust in distributed software projects has been recognized by several researchers (e.g. [2][3][4]). In an empirically-based study of six software development companies Jalali, Gencel, and Šmite [5] found that trust building is initially a static process that evolves to a dynamic situation as the project develops and team members change opinion about their colleagues behavior towards the project activities. Al-Ani and colleagues confirm that trust development is a dynamic process in their study of 5 multinational companies and further knowledge describing the phases that compose such dynamic process, namely formation, dissolution, adjustment and restoration [6]. In an empirical study of a large organization Al-Ani and Redmiles identified team size, project type, diversity, and leadership as factors that might affect the development of trust in distributed teams [1]. Trainer and Redmiles [7] have expanded on these factors based on a review of literature. Expertise, years of experience, availability, reputation are examples of suggested factors.

Other studies of distributed software teams suggest the critical role trust plays in project performance (e.g., [8][9]). However, to the best of our knowledge the only study on the field exploring such relationship is from Moe and Šmite [10]. They have investigated the impact of lack of trust in the success or failure of distributed software projects in an Easter European company and found that the lack of trust caused a decrease in productivity, quality, information exchange and feedback, and morale among the employees as well as an increase in relationship conflicts.

We sought to expand this initial knowledge and conducted an empirical study of 4 distributed software projects from 3 distinct companies to investigate which factors influence one's perceived trustworthiness of others in the project and what is the impact of such factors on project performance measures. We present our initial findings in this paper.

II. Method

Our empirical study consisted of the application of a survey instrument that listed 30 trust factors obtained from a systematic literature review (refer to Table 1 for the complete list) and 7 project performance metrics obtained through interviews with experienced project managers of distributed software projects. The metrics are: cost deviation, effort deviation, productivity, requirements completion, requirements volatility, product quality, and time adherence. Appendix A and B present the factors and metrics definitions, respectively. The instrument was applied either in person or over the telephone and its application lasted 30 min on average.

To answer the instrument the participant was instructed to indicate his opinion based on his own overall experience working with distributed teams about which of the 30 trust

factors previously defined are associated with each one of the 7 defined metrics. The following types of association were indicated: (i) *negative* (-), whether the trust factor negatively influences the performance metric; (iii) *neutral* (0), whether the influence can be positive or negative depending on the situation; (iii) *positive* (+), whether there is a positive influence; and (iv) *N/A* (NA), whether the respondent believes the factor does not apply. We understand that perceptions about the factors can be interchangeable. For example, availability might positively affect one's trust in a colleague but lack of availability might have a negative influence. Therefore, we asked the participants to indicate the category they believe that better describe their perception.

A total of 33 participants from 4 distinct projects accepted invitation to participate. Two projects were from a large US-based IT manufacturing company (6 members and 67% response rate for project 1, 6 members and 50% for project 2), one from a large Brazilian company with customers located in several states and the US (8 members and 75% response rate), and one from a local IT company with offices in 4 states (30 members and 67% response rate). Respondents were on average between 31-40 years old, have 11.5 years experience in software engineering, and 6.5 years experience working with distributed teams. The participants' roles distribution is as follows: 3 business analysts, 6 system analysts, 9 developers, 7 test engineers, 4 testers, 2 build engineers, and 2 managers.

III. Initial Findings

Simple statistics analysis was conducted to describe the findings. The percentages in Table 1 indicate the relative number of the predominant answers for each trust factor over the total of the 33 respondents. For instance, line 1 indicates that the *adoption of patterns* factor was considered to influence the establishment of trust and the participants believe that it positively affects each one of the listed performance metrics, some more than others. Hence, this finding suggests that the more the distributed teams adopt patterns to guide work the more they rely on their colleagues benefiting the cost of the project (CD), helping achieving the estimated effort (ED), increasing productivity (P), facilitating attending the requirements agreed (RC), reducing requirements volatility (RV), increasing quality (PD), and staying on time (TA).

Figure 1 summarizes the factors that *positively* influence the development of trust and as a consequence *positively* affect the defined performance measures. We can see that for the collaboration, competence, expertise, leadership, and work experience factors about three-thirds of the respondents agree on their opinion for each of the metrics. For the other 5 factors, namely adoption of patterns, availability, F2F communication, monitoring, and prior work experience on average half of the respondents reached an agreement. This percentage is still significant considering this is a step of a larger investigation.

TABLE I. Results

Trust factors	CD	ED	P	RC	RV	PQ	TA
Adoption of patterns	+ 61%	+ 55%	+ 67%	+ 52%	+ 45%	+ 85%	+ 55%
Availability	+ 45%	+ 52%	+ 55%	+ 58%	+ 39%	+ 64%	+ 61%
Betrayal	+ 55%	- 55%	- 45%	0 42%	- 42%	- 52%	- 52%
Collaboration	+ 70%	+ 55%	+ 79%	+ 76%	+ 52%	+ 76%	+ 76%
Comm. media	+ 42%	0 45%	0 55%	0 55%	0 45%	0 48%	+ 48%
Competence	+ 85%	+ 79%	+ 76%	+ 79%	+ 64%	+ 82%	+ 82%
Culture	+ 42%	0 39%	0 52%	0 48%	0 48%	0 55%	0 52%
Expertise	+ 76%	+ 67%	+ 82%	+ 88%	+ 67%	+ 76%	+ 70%
F2F comm.	+ 55%	+ 55%	+ 48%	+ 52%	+ 58%	+ 58%	+ 55%
Fear of job loss	+ 52%	- 39%	0 48%	0 42%	0 42%	- 42%	0 52%
Freq. of mtgs.	+ 42%	0 55%	0 58%	0 52%	+ 42%	0 42%	+ 36%
Geographical distance	+ 58%	- 48%	0 61%	0 39%	- 39%	0 42%	0 48%
Homophily	+ 48%	0 36%	0 39%	0 42%	0 36%	0 39%	0 42%
Informal com.	+ 36%	0 52%	0 39%	0 55%	0 36%	0 33%	0 30%
Intuition	+ 48%	0 45%	0 48%	0 52%	0 45%	0 42%	0 48%
Language	+ 39%	+ 27%	0 48%	0 45%	0 39%	0 33%	0 52%
Leadership	+ 52%	+ 70%	+ 67%	+ 73%	+ 48%	+ 55%	+ 70%
Monitoring	+ 48%	+ 58%	+ 52%	+ 64%	+ 48%	+ 67%	+ 73%
Prior work experience	+ 61%	+ 61%	+ 58%	+ 52%	+ 45%	0 48%	+ 52%
Project size	+ 52%	0 45%	0 55%	0 42%	0 45%	0 67%	0 70%
Project type	+ 45%	0 58%	0 55%	0 55%	0 52%	0 55%	0 61%
Project changes	+ 58%	- 61%	- 58%	- 64%	- 61%	- 61%	- 67%
Reputation	+ 42%	0 36%	0 42%	0 52%	0 48%	0 36%	0 52%
Response time	+ 52%	+ 52%	0 45%	+ 39%	0 39%	0 39%	0 45%
Role	+ 64%	0 58%	0 42%	0 58%	0 39%	0 48%	0 76%
Shared personal info	+ 36%	0 45%	0 45%	0 45%	0 48%	0 58%	0 48%
Team diversity	+ 39%	0 42%	0 48%	0 36%	0 48%	0 55%	0 52%
Team size	+ 55%	0 45%	0 55%	0 36%	0 70%	0 76%	0 67%
Virtual comm.	+ 36%	+ 39%	0 55%	0 58%	0 45%	0 48%	0 42%
Work experience	+ 70%	+ 73%	+ 67%	+ 55%	0 58%	+ 70%	+ 70%

Fig. 1 Factors that positively influence trust development and project performance

Figure 2 indicates the factors that *negatively* influence the development of trust and as a consequence *negatively* affect the defined performance measures. Two factors fall into this category only: betrayal and project changes, having on average half and a bit over 60% of the participants agreeing on the impact respectively.

Fig. 2 Factors that negatively influence trust development and project performance

Figure 4 outlines the factors that have *neutral* influence on the development of trust and as a consequence *neutral* effect on the listed performance measures, i.e. these factors might either have a positive or a negative effect depending of the context it is related to. The majority of the factors, 18 of them, fall into this category suggesting that trust is not a static perception as previously reported in literature ([5][6]) and its presence or absence can change the project's results. For all factors there is on average an agreement among half of the respondents about the impact these factors cause in the development of trust and on the presented performance measures.

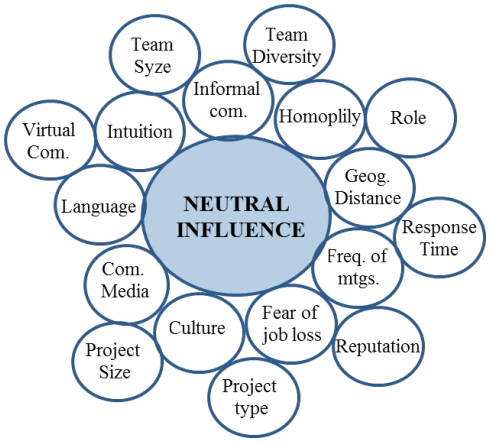

Fig. 3 Factors that have neutral influence on trust development and on project performance

IV. Discussion

We have found that about two-thirds of the factors identified from literature influence the development of trust and interfere with its growth according to the situation the team members are in. For instance, cultural differences might facilitate or challenge the development of trust towards remote colleagues according to the participants' perceptions of culture. Three of these factors—project type, team diversity, and team size were suggested as factors that might have a negative influence on the development of trust by Al-Ani and Redmiles [1]. Our initial findings suggest that such factors can also play a positive role in certain situations. Further qualitative exploration is necessary to gain a better understanding of which situations create the environment to lead to such findings.

We also found that about one-third of the factors positively affect one's perceived trustworthiness of others in the project and that this is positive for the achievement of the projects defined metrics. Leadership, the fourth suggested factor from Al-Ani and Redmiles' work [1], is among these factors and corroborates their findings. Only two of the factors, betrayal and project changes, are considered having a negative impact. The act of betraying someone is somehow expected to negatively impact a relationship as well as frequent changes to baseline versions of work products (e.g. requirements or architecture) are known to affect project outcomes (e.g. [11]).

More specifically, results indicate that the productivity and quality metrics are positively or negatively impacted by most of the factors that lead to trust (or lack of it), corroborating to a certain extent with Moe and Šmite's previous findings [10]. Our preliminary findings help confirm that trust building is a dynamic process and that it impacts project outcomes in distributed software projects. In addition, the large amount of factors indicated as of neutral influence suggest that the establishment of trust is context-specific.

V. Final Remarks

In this paper, we present the initial findings of our survey of 33 participants from 4 distinct projects of 3 large IT companies. The purpose of the survey is to identify which factors positively or negatively influence one's perceived trustworthiness of others in the project and the impact of such factors on specific project performance measures. We found that our participants considered the majority of the factors neutral, meaning that they might influence positively or negatively the development of trust and as a consequence might impact positively or negatively the achievement of the project performance metrics. One-third of the factors are considered to have positive influence, and only two of them had a negative influence.

The number of participants and their concentration on the same location (a state in the South of Brazil) are limitations of this study. However, this is a relevant first step towards developing a theoretical model of the impact of trust in the performance of distributed software projects. We expect the availability of such model to be helpful to researchers as a reference framework to further investigate the topic and to

practitioners to guide improvement actions aiming to better define and manage distributed software teams.

Next we will replicate the study is large scale and consider global distribution to minimize the impact of cultural and time zone differences influence on the model. We will then be able to apply more refined statistical methods to confirm the significance of our findings.

ACKNOWLEDGMENT

This study is part of a major research investigation under development at the Munddos Research Groups, coordinated by Professor Dr. Rafael Prikladnicki. We thank for his support and valuable insights. This research is partially financed by Dell Computers of Brazil Ltda with funds from the Brazilian Federal Law No. 8.248/91.

REFERENCES

[1] B. Al-Ani and D. Redmiles, "In strangers we trust? Findings of an empirical study of distributed development," in Proc. of the Int'l Conference on Global Software Engineering, Limerick, Ireland, August 2009, p. 121-130.

[2] J. Treinen and S. Miller-Frost, "Following the sun: Case studies in global software development," in IBM Systems Journal, vol. 45, no. 4, 2006, p. 773-783.

[3] J. Herbsleb, D. Paulish, and M. Bass, "Global software development at Siemens: Experience from nine projects," in Proc of the Int'l Conference on Software Engineering, St. Louis, USA, May 2005, p. 524-533.

[4] J. Zheng, E. Veinott, N. Bos, J. Olson, and G. Olson, "Trust without touch: jumpstarting long-distance trust with initial social activities," in Proc. Of the SIGCHI Conf. on Human Factors in Computing Systems, Minneapolis, USA, April 2002, p. 141-146.

[5] S. Jalali, C. Gencel, and D. Šmite, "Trust dynamics in global software engineering," in Proc. of the Int'l Symp. on Empirical Software Eng. and Measurement, Sept 2010, article 23, 9 pages.

[6] B. Al-Ani, M. Bietz, Y. Wang, E. Trainer, B. Koehne, S. Marczak, D. Redmiles, and R. Prikladnicki, "Globally distributed developers: Their trust expectations and processes," in Proc. of the Conf. on Computer Supported Cooperative Work and Social Computing, Feb 2013, San Antonio, USA, pp. 563-573.

[7] E. Trainer and D. Redmiles, "Foundations for the design of visualizations that support trust in distributed teams," in Proc. of the Advanced Visual Interfaces Int'l Working Conf., Capri Island, Italy, May 2012, p- 34-41.

[8] R. Sabherwal, "The role of trust in outsourced IS development projects," Com. Of the ACM, vol. 42, no. 2, Feb 1999, p. 80-87.

[9] J. Walther and U. Bunz, "The rules of virtual groups: Trust, liking and performance in computer-mediated communication," Journal of Communication, vol. 55, 2005, p. 828-846.

[10] N. Moe and D. Šmite, "Understanding a lack of trust in global software teams: A multiple-case study," Software Process Improvement and Practice, vol. 13, Jun 2008, p. 217-231.

[11] B. Curtis, H. Kransner, and N. Iscoe, "A field study of the software design process for large systems," Com. of ACM, vol. 31, no. 11, p. 1268-1287.

APPENDIX

A. DEFINITION OF THE TRUST FACTORS

Adoption of patterns: To adopt certain standards to support the work to be done such as CMMI, ISO, BPMI, ITIL, among others.
Availability: Being a handy person, always present and available to answer questions and to assist coworkers, in person or virtually.
Betrayal: To be false or disloyal, to reveal against one's desire or will.
Collaboration: Cooperation between project members.
Communication media: Technological means employed to establish communication. Eg.: E-mail, chat, phone, etc.
Competence: The quality of being competent; adequacy; possession of required skill, knowledge, qualification, or capacity.
Culture: The culture of a country, which may be different from another country, with different customs that can be conflicting. Eg.: The culture of India is different from the culture of Brazil.
Expertise: To have in-depth knowledge about a certain topic, technology or business domain.
F2F communication: To communicate with other in a presential manner.
Fear of job loss: One's fear of losing the job to a remote colleague; to believe that others might want to take away one's role in the project.
Frequency of meetings: The frequency at which meetings are set up.
Geographical distance: Physical distance between project teams.
Homophily: Similarity among team members. Eg.: similar age or gender.
Informal communication: Unplanned meetings. Eg.: Discussing a project issue in the coffee area or while smoking a cigarette.
Intuition: Direct perception of truth or fact, independent of any reasoning process; immediate apprehension. Eg: To sympathize with a colleague even without any personal contact.
Language: The language spoken by the project members.
Leadership: The function of a leader, a person who guides a group.
Monitoring: Constant monitoring of progress of team members.
Prior work experience: Time one has previously worked with a colleague or knows another colleague from working together in past projects.
Project size: Number of people allocated to work on a project.
Project type: The classification of a project according to its main goal. E.g. Improvement, maintenance, new development, innovation.
Project changes: Modifications that occur in the project after a baseline is approved. E.g.: Changes to the defined scope.
Reputation: A favorable and publicly recognized name or standing for merit, achievement, reliability.
Response time: Delay between the time something is requested and it is resolved.
Role: It is the role that person plays in project. Eg Project manager.
Shared personal information: Share personal information with coworkers in order to foster interpersonal relationships.
Team diversity: Different profiles within the same team. Eg.: Having in a single team shy, outgoing, among other characteristics' team members.
Team size: Number of members of a team.
Virtual communication: type of communication characterized by not being face-to-face and supported through technological means.
Years of professional experience: Time of professional experience that a team member has.

B. DEFINITION OF THE METRICS

Cost deviation: The percentage between the actual cost / estimated cost.
Effort deviation: Deviation of the number of hours/man in comparison with the planned effort.
Productivity: Number of lines of code per developer.
Requirements completion: Number of requirements completed compared to the list of requirements agreed in the project scope.
Requirements volatility: Number of additional requirements added to the initial project scope.
Product quality: Measurements of the quality of the system being delivered. E.g., the percentage of defects found in each testing phase.
Time adherence: Adherence to the project schedule.