Facilitating contagion trust through tools in Global Systems Engineering teams

Ban Al-Ani [a,*], Sabrina Marczak [b,1], David Redmiles [a,2], Rafael Prikladnicki [b,3]

[a] Department of Informatics, Donald Bren School of Information and Computer Sciences, University of California, Irvine, CA 92697-3425, USA
[b] Pontifícia Universidade Católica do Rio Grande do Sul (PUCRS), Av. Ipiranga 6681, Faculdade de Informática, Prédio 32, Sala 505, Partenon, Porto Alegre, RS 90619-900, Brazil

A B S T R A C T

Context: In Global Systems Engineering teams, researchers have found that trust can be *transitive* to some degree or imported (*swift trust*) under certain conditions. We argue that trust can be *contagion* and seeded by tools (spread from one individual to another through tools).
Objective: We sought to investigate the potential for using tools to support the development of trust in such teams and facilitate *contagion trust*. Specifically, we sought to investigate whether any existing tools support the development of trust in such teams and which information helps such development, whether the visualization of past collaborations would help developing trust, and what tools or features practitioners would wish for, if they had a magic wand.
Method: We interviewed 71 employees from five multinational organizations. We focused on gaining an understanding of the tools that are currently used to engender trust and the information needed to facilitate contagion, in which conditions visualizations of past collaborations are helpful, and what software tool features could help develop trust. Our analysis was guided by grounded theory.
Results: We found evidence that supports the theory of contagion trust and tools can be used to initiate the development of trust. These tools include software tools, office technologies, or organizational structures. Practitioners' needs were functional (e.g. audio channel with remote colleagues) and/or non-functional (e.g. can be adopted in sites with poor infrastructure).
Conclusion: Our study illustrates that tools can be used to facilitate *contagion trust* and provides three main contributions. First, our exploration of how existing tools are used provides a guide to effective practices in such teams. Second, the descriptions of features that can facilitate contagion trust provide useful design implications for future tools. Third, the identification of the kind of information that facilitates contagion trust provides an understanding of practitioners' underlying needs that can be used to develop collaboration tools.

© 2013 Elsevier B.V. All rights reserved.

1. Introduction

"I think if I had a magic wand, what I would do in order to build more trust …" (P61 – female, quality assurance developer located in Ireland)

* Corresponding author. Present address: 400 Covina Blvd., San Dimas, CA 91773, USA. Tel.: +1 909 291 5184.
E-mail addresses: balani@ics.uci.edu, ban.alani@adp.com (B. Al-Ani), sabrina.marczak@pucrs.br (S. Marczak), redmiles@ics.uci.edu (D. Redmiles), rafael.prikladnicki@pucrs.br (R. Prikladnicki).
[1] Tel.: +55 (51) 3320 3558x8646.
[2] Tel.: +1 949 824 3823.
[3] Tel.: +55 (51) 3320 3694.

Globally distributed teams' interactions typically lack socially contextual cues, such as nonverbal information and status, which can prohibit the development of trust in such teams. Consequently, it is expected that team members will feel more anonymous and usually more focused on themselves and less on others. The lack of social cues may lead to increased depersonalization, lower cohesiveness, and less social conformity. This can often mean that team members do not share necessary information and prefer to attend to personal information [1]. Previous exploratory studies (e.g., [2]) revealed that there exists a potential for tools to influence the development of trust through visualizations of collaborative traces. Collaborative traces are representations of past and current activity of a group of developers manipulating software development artifacts and can be made accessible through tool support [3].

Previous research also established that trust can be imported under certain conditions and that *swift trust* can be engendered

as a result [4]. Researchers have also confirmed that trust is not fully *transitive* in the mathematical sense [5]. *Swift trust* is formed by importing trust rather than developing trust. It is at its highest at project inception and typically develops in certain conditions [4]. Trust is also often considered transitive in the sense that if John trusts Maria, and Maria trusts Abbass, then John will trust Abbass. This transitivity is often questioned by researchers, some of whom assert that trust is only transitive under certain conditions [6], and may not be transferred at the same levels [5].

Our extensive review of seminal work in addition to our own work that we report in this article led us to conclude that while trust can only be imported under certain conditions and was not fully transitive, it could be *contagion* and seeded by tools. In other words, it is possible for trust to spread from one individual to another through direct or indirect interactions that are facilitated by tools. We will discuss these concepts and others further in our review of related work. We also report aspects of trust in Global Systems Engineering teams derived from this study elsewhere [7,8]. This article complements our previous work, which focuses on the nature of trust, expectations and processes, and collaborators' adaptations necessary to enable collaborations across international boundaries.

Specifically, in this present work, we seek to investigate the potential for using tools to support the development of trust in distributed development teams and facilitate *contagion trust*. We sought to investigate the following research questions:

(RQ1) Can existing tools or tool features engender or support the development of trust in distributed teams?

(RQ2) Can visualizations of previous performances influence the sense of trust towards new team members?

(RQ3) What tools or features do individuals working in globally distributed development teams feel they need to engender trust towards others that are still not available or do not exist yet?

We conducted a series of 71 interviews over a 12-month period to investigate these research questions within the larger context of trust in Global Systems Engineering teams. In sum, we found that trust is contagion and that developers use some of the features of existing tools to promote trust. Participants mentioned a myriad of tools that included software tools, office technologies, or organizational structures. Practitioners also discussed what they thought would help them develop a sense of trust towards remote collaborators. Our study investigated their functional (e.g. audio channel with remote colleagues) and non-functional (e.g. can be adopted in sites with poor regional infrastructure) needs, collecting detailed accounts of the kind of information they need to facilitate trust.

Our review of related literature, an outline of our research method, the analysis of study data and relevant findings are presented in the following sections. The review of literature in Section 2 is made up of a section that describes different facets of trust in general and also within Global Systems Engineering teams specifically. We also explore the contagion behavior in teams in Section 2 of our review of literature. Our review of literature concludes with a discussion of supporting the development of trust contagion through tools.

A description of our study approach in Section 3 is followed by an account of our study findings in Section 4. These findings are presented in three consecutive sections. First, we present study findings concerned with the usage of existing tools to engender trust that include our exploration of Web 2.0 usage, available office technologies and applications in addition to other organizational resources available to them.

Second, we present findings that were derived from discussing a hypothetical tool that we presented to our participants along with a scenario in which they had access to a tool in which socio-technical dependencies of team members are depicted. The participants discussed the likelihood that such depiction could help the development of contagion trust. Participants' responses were categorized as being positively disposed towards such a hypothetical tool, negatively disposed or conditionally disposed towards such a tool.

Finally, we present participants' responses when faced with the freedom to choose to support the development of trust through any means. We stress that such means have no boundaries, i.e. without the current constraints imposed by current tools available in the market. In this scenario, the participants are given a "magic wand" and can wish for whatever they want. Thus participants typically wished for direct support (functional requirements) or freedom (non-functional requirements) in this hypothetical scenario, and the responses are presented accordingly. The paper concludes with a discussion of study findings in Section 5, the study limitations in Section 6, and the implications of our findings for the theory and practice of Global System Engineering teams in Section 7.

2. A Review of related literature: definition of terms and concepts

Globally distributed teams have become prevalent as the result of the interplay of factors which include an increase in travel costs, a competitive market, the distribution of resources, and the availability of software platforms for collaborative activities, among other factors [33]. While there is some progress in supporting collaborations among members of virtual teams, we find trust remains an aspect that bears further investigation.

A team is generally defined as being made up of two or more individuals who share a common goal or purpose to achieve a certain outcome [9]. We use the term Systems Engineering teams to refer to the individuals who are involved in the diverse aspects associated with software development such as the lawyers who negotiate contracts across sites, the software engineers, the hardware engineers that develop hardware in which the software will be embedded, etc. The term Global Systems Engineering teams refers to distribution of these individuals across geographical and temporal distances. Such distribution typically necessitates the need to adopt one or more tools to help overcome some of the challenges of collaborating across different boundaries that can emerge during a project's lifetime. These challenges are exacerbated when additional boundaries are introduced as a result of geospatial and temporal distances when team members are located in culturally diverse regions, and may not meet during a project's lifetime. Such challenges can inhibit the development of trust in such teams [8].

The primary objective of our research was to investigate the possibility that tools can be used to support the development of trust in Global Systems Engineering teams to facilitate *contagion trust*. To this end, we initially focused on developing an understanding of study terms and concepts used namely, *trust, tool usage,* and *contagion* all within the context of *Global Systems Engineering* and based on existing work. We present the results of this first step (that are relevant to the findings we report in this paper) in the following sections.

2.1. An understanding of trust and global software engineering teams

Trust is considered important for collaborators to work effectively and share information openly. It can reduce transaction costs, increase confidence and security in the relationship; promote open, substantive and influential information exchange [4]. Researchers have found that without trust transactions must be carefully contracted and monitored to prevent exploitation, workers change the nature of their collaborations to avoid the need for