Collaboration Patterns and the Impact of Distance on Awareness in Requirements-Centred Social Networks

Daniela Damian, Sabrina Marczak, and Irwin Kwan
Software Engineering Global interAction Lab - SEGAL
Department of Computer Science - University of Victoria
3800 Finnerty Road, Victoria, British Columbia, Canada
{danielad, smarczak, irwink}@cs.uvic.ca

Abstract

Because of intense collaborative needs, requirements engineering is a challenge in global software development. How do distributed teams manage the development of requirements in environments that require significant cross-site collaboration and coordination? In this paper, we report research that used social network analysis to explore collaboration and awareness among team members during requirements management in an industrial distributed software team. Using the lens of a requirements-centred social network to group team members who work on a particular requirement, we collected data to characterize requirements-centric collaborations in a project, and to examine aspects of awareness of requirements changes within these networks.

Our findings indicate organic patterns of collaboration involving considerable cross-site interaction, in which communication of changes was the most predominant reason for interaction. Although we did not find evidence that distance affects developers’ awareness of remote team members who work on the same requirements, distance affected how accessible the remote colleagues were. We discuss implications for knowledge sharing and coordination of work on a requirement in distributed teams, and propose directions for the design of collaboration tools that support awareness in distributed requirements management.

1. Introduction

Global software development (GSD), driven by growing business opportunities and advanced communication technologies, has created challenges in coordination and collaboration. The increase in distance between project team members brings about problems in awareness of progress that affects one’s work [8][1][6]. Requirements engineering (RE) in particular is a key issue in GSD. Contribution from many stakeholder roles is needed throughout the software development life cycle to define, develop and test requirements. Up-to-date information about requirements and their evolution in changing environments is critical. Changes to requirements and design specifications, frequent in large software projects [1], need to be communicated promptly to team members to avoid negative impacts on quality and team productivity. However, not only does GSD introduce delays in project communication [9], but distributed software teams have to coordinate work across diverse organizational settings, cultural backgrounds, and time zone differences [4][9]. Consequently, developers have difficulty coordinating requirements development; there is little support for monitoring progress of requirements or identifying team members that are knowledgeable of certain features [8]. While some collaborative tools aiming at supporting RE in distributed teams [11] rely on teams self-subscribing to communication about a particular requirement, we found that teams that have relevant knowledge and work related to particular requirements have dynamic membership with unpredictable patterns [10]. These teams evolve over time and are affected by factors of geographical distance, organizational structure, and RE process. What we still do not know is how distributed teams manage requirements evolution and change information, and what gaps in communication and awareness need to be identified in order to avoid loss of critical knowledge within these networks.

In this paper, we report from a case study of an industrial distributed project where we focus on understanding collaboration within groups of people related to a certain requirement. We use social network
analysis and the concept of requirements-centred social networks (RCSNs) to analyze collaboration and awareness of other members working on same or interrelated requirements.

We seek to (1) gain deeper insights into characteristics of requirements-centred social networks and their dynamic nature, such as who the members of these networks are, what information these members exchange, which roles and interactions are emergent, and how team membership differs from those anticipated in project plans as well as collaboration patterns across distanced sites; and (2) examine aspects of awareness of colleagues working on the same requirements and how easily they can be reached, how someone can provide assistance with particular requirements, as well as the impact of distance and communication on this awareness.

The paper begins with an introduction of RCSN and awareness issues within RCSNs in sections 2 and 3. Then we describe our research methodology at a large software organization with distributed teams in Section 4. Section 5 describes our findings, while Section 6 discusses them in relation to our goals and draws implications for research and practice. Section 7 presents our conclusions, the research limitations and future work.

2. Dynamic Requirements-Centred Social Networks

The traditional definition of a team in software engineering presents an image of a strict hierarchical structure in which team members work on related components. However, studies of current practice [6][9] reveal that team members are often working on multiple requirements within cross-functional teams that may contain developers, testers, and technical writers. A study of requirements-engineering process improvement in a medium-size organization [3] indicates the positive impact of collaboration within cross-functional teams during requirements-management processes. These teams, composed of designers, developers and testers, used collaborative activities during the analysis of requirements to maintain awareness of requirements and their changes throughout the project life cycle, reducing rework and risk while increasing developer productivity.

We define a requirements-centred team (RCT) as a cross-functional team in which each member is involved in a particular stage of a requirement’s development (e.g. design, code or test). A team member may belong to more than one RCT at one time, and a project has as many RCTs as the number of requirements. By relating the team members who work on the same requirements, we can gain a better understanding of how people collaborate and coordinate based on the requirements-related tasks that they complete.

RCTs, in addition to encompassing members from different teams, has a changing membership. As the development of a requirement evolves, more people are involved in contributing to the corresponding RCT. Our own research indicates that the group of people who work on common feature is continually expanding [9], as a result of expertise seeking and management of inter-dependencies between requirements. Ehrlich and Chang also found that team members often go outside of their established team boundaries when seeking information about their work [6]. When a member of an RCT collaborates with a person who was not initially allocated to work on a specific requirement in order to help develop this requirement, this person is considered as an add, or an emergent team member, to the RCT.

We can represent relationships among team members in a RCT using a social network. We thus define a requirement-centred social network (RCSN) as a social network that represents an RCT. Each connection between the members in an RCSN represents a communication line between two team members in which the participants communicate about the requirement. We can use the RCSN to study the collaboration among team members relevant to the design, development, and testing of a requirement, as well as any awareness problems they experience in distributed interaction.

To better understand the dynamic nature of an RCT, and how we can support effective collaboration within cross-functional teams, especially distributed, we can use an RCSN to study the evolution of the RCT over time. By deriving an RCSN from project plan data (such as a work breakdown structure), we can generate a planned RCSN that indicates who should be communicating with whom in the project. We can compare this to an actual RCSN generated using actual communication data during the project, and identify the differences between the planned RCSN and the actual RCSN. Work comparing a coordination requirements matrix and an actual coordination matrix has been introduced by Cataldo, et al. [1] as a method to examine gaps in communication. While Cataldo, et al identify coordination requirements based on code dependencies, we are interested in examining the dynamics of requirements-centric networks.

The RCSN, with its emphasis on communication lines between team members, can be a useful tool when studying interaction in distributed teams. We can observe communication among the collocated team members, as well as between members of the
distributed team, and study effects due to distance and availability. In this study we are particularly interested in using the RCSNs to examine how a distributed development team in industry propagates information to make every team member aware of the state of the requirements, their changes, and manages collaboration around them.

Our research seeks to understand the RCT by characterizing their composition and dynamic evolution throughout software development. In our study, we collect information from project plans to generate a planned RCSN, and use questionnaire data gathered during the project to generate an actual RCSN. We intend to provide information on what roles become members of RCSNs, what information they exchange and what portion of this communication is cross-site, and what the emergent roles are that are involved in emergent interactions. This information can be used to improve how distributed team members create and maintain awareness of each other during the development requirements work they are allocated to. In distributed settings, which do not benefit from informal communication, collaborative tools may be improved to support the timely propagation of requirements change information among members whose work is related to particular requirements.

3. Awareness within Requirements Engineering

RE involves information acquisition and expertise seeking, as well as coordination across multiple teams. Geographical distance is known to negatively affect communication among stakeholders [9]. Awareness, which is defined as an understanding of the activities of others, which in turn provides a context for one’s own activities, is linked to issues related to coordinating team effort. Communication is a main means to distribute awareness information, so we expect that a developer’s awareness of whom else is working on his requirements and whether they are available for contact is affected in GSD. We are interested in how aware team members in a GSD project are, and what influences awareness.

General awareness [6] refers to an individual’s knowledge of who has expertise in the project. With respect to requirements work, an individual with good general awareness can readily identify who in the project team can provide help with a particular requirement. We are interested in how a distributed development team, which is limited to remote communication, learns about each other’s abilities. Current awareness, or task awareness, refers to how aware an individual is of another individual’s workload [6]. In requirements engineering, this is important because a software developer would benefit from knowing about developers working on related or similar requirements. Availability [6] refers to how aware one is of an individual’s accessibility for information seeking.

Research so far indicates that awareness is significantly affected in distributed teams. Ehrlich, et al [6] found not only that people tend to communicate more frequently with someone they have current or general awareness on, but also that both general and current awareness, as well as availability decrease significantly across sites. Herbsleb reported similar findings, observing that outside of scheduled communication, distributed team members rarely communicated with each other [8].

In this paper, we focus on studying the impact of distance on awareness within RCTs. Unlike previous research [6] that studied awareness in the project at large, our investigation is highly focused on awareness within a group of people associated to the same requirement and which we believe should maintain high awareness of requirements updates. In our analysis, we (1) collected information on whether project members maintain awareness of work and changes within RCSNs, and (2) investigated whether current awareness, general awareness and availability are correlated with communication patterns in the project and whether they are affected by geographical distance. Further, we wanted to obtain insights into whether project members have awareness of members that work on inter-related requirements, so that to further our understanding of coordination needs within requirements-centric social networks.

4. Research methodology

We conducted a case study at the Brazilian software development center of one large international IT manufacturing company. The project we selected for study had significant interaction between the Brazilian location and the United States (US) headquarters location. The project was an infrastructure maintenance project, to an application developed in US seven years ago. The project team consisted of 10 members (7 in Brazil and 3 in US); more specifically, 6 developers, 1 system architect, 1 test leader and 2 technical leaders (one of them also acted as a business analyst).

4.1. Data collection methods

We used three data collection methods: document inspection, questionnaire and interview.
Document inspection. We inspected documentation to identify the planned RCSNs. For each requirement described in the Software Requirement Specification document, we identified the list of people allocated to work in every task related to the particular requirement. This list was identified through inspection of the Project Management and Task Allocation Plans.

Questionnaire. We designed a questionnaire after identifying the planned RCSNs inquiring about awareness of other team members in the network. We applied the questionnaire to each team member independently during the Testing phase of the project. A paper version questionnaire was given to 7 Brazilian members and an e-mail version was given to 3 US members. We received responses from 7 (100%) Brazilians and 1 (33.3%) American member, thus overall response rate was 80%.

The questionnaire consisted of 29 questions intended to collect data on the three areas of interest: demographics (geographical location, years of professional experience in IT area, time working on the company, and role on the project), communication patterns in RCSNs, and awareness in RCSNs.

To study communication patterns in RCSNs, we collected data on both the planned and actual RCSNs. We first identified details of each respondent’s interaction with the team members we identified as belonging to the respective planned RCSNs. This included reason for communication (Communication of Changes, Coordination of Activities, Implementation of Issues, Planning, Requirement Clarification, Risk Assessment, Support, Synchronization of Code, and Other), the communication media used (Chat, E-mail, Face to Face, Telephone, Voice mail, and Other), and the frequency of communication (Less than once a week, Once or twice a week, 3-4 times a week, 5 or more times a week). Note that this questionnaire item was customized for each project member: Upon identification of the requirements he was allocated to work on, the respective RCSN (and their members) was identified and thus the list of names included in the question was changed accordingly. Then we asked the respondents to identify other people they interact with (“Please identify the names of additional people that you communicate with at least once a week in connection with this project and provide details of interaction”), as a way of identifying membership of actual RCSNs. For these emergent interactions, we asked the reason for interaction (regarding the requirement discussed), the media used to support the communication, and the information exchanged (same categories of interaction listed above).

In order to investigate the awareness in the project, we first collected data from the project at large: whether team members have awareness of who is working on the same requirements (locally or remotely), how team members identify who is working on the same requirements, if they believe the communication of changes is in time to avoid rework, and which information they currently receive and would like to receive about requirements changes.

A second set of questions was customized to collect data specifically within the identified requirement networks (RCSNs), as follows.

Current awareness: How aware are you of what the following team members are doing that is related to your work? Provide answers in relation to the requirements you are working on. (Very aware, Aware, Not aware, Very unaware, N/A) General awareness: How aware are you whether this team member could help you in your work? (Very aware, Aware, Not aware, Very unaware, N/A) Availability: How easy is it for you to reach these team members when you need help or information related to the project? (Very easy, Easy, Difficult, Very difficult)

To study these types of awareness within RCSNs, we asked each respondent to answer these questions only in relation to those people we identified as belonging to the particular respondent’s RCSNs. Thus, this list differed from one person to the other according to the requirements the team member was allocated to work on and his/her planned RCSN. For example, if one team member worked on requirement 6 (R6) and on requirement 8 (R8), then we included in this team member’s list all people working on R6 and those working on R8. The list of requirements also differed from one team member to another.

Interviews. We conducted 5 follow-up interviews to collect missing data from questionnaires. Later we interviewed the Brazilian Technical Leader to obtain information about work and communication structures, and to investigate their impact on the way the project team coordinates and collaborates to perform its work.

5. Findings

Out of the 8 respondents, 7 were located in Brazil (BR) and 1 in US. Their roles distribution was as follows: 5 developers, 1 architect, 1 test leader, and 1 technical leader. The respondents had in average 8 years of experience in IT area. Three of them were new in the company (hired less than 1 year ago), 2 have been working between 1 and 3 years, and 3 between 4 and 7 years. During interviews we found out that 100% of the developers have been working in the same application since they were hired, suggesting that they are comfortable with each other. Each respondent was involved with the project since its conception.
5.1. Characterization of RCSNs

Who is involved in actual RCSNs? Thirteen requirements were identified in the Software Requirement Specification document and we thus generated 13 planned RCSNs. Each RCSN included a subset of the 10 team members who were assigned to the project. To build each corresponding actual RCSNs, we identified, from questionnaire data, all project members that the respondents identified as communicating about the requirement in addition to those we listed from the planned RCSN. Analyzing these 13 actual RCSNs, we identified a total of 38 emergent team members (adds) across all RCSNs; these are non-unique, i.e. there are duplicates across RCSNs. Table 1 shows the information from the planned and actual RCSN of requirement 6 (R6), as an example, as well as the distribution of team members across all RCSNs. On average, there are a total of 2.9 emergent people per network, 2.3 of them (79.3%) are from Brazil and 0.6 (20.7%) from US. This indicates that, on average, about one third of the team members in the project are emergent in an RCSN.

Table 1. Distribution of team members

<table>
<thead>
<tr>
<th>Requirement</th>
<th>Team members in planned RCSN</th>
<th>Team members in actual RCSN</th>
<th>Emergent people in the actual RCSN (adds)</th>
</tr>
</thead>
<tbody>
<tr>
<td>R6</td>
<td>3 1 4 2 1 1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total across all requirements.</td>
<td>41 22 71 30 30 8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Average</td>
<td>3.2 1.7 5.5 2.3 2.3 0.6</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

To further characterize the membership of the RCSNs, and in particular their emergent members, we analyzed the emergent team members’ roles within the 13 dynamic RCSNs. Table 2 shows the total of people in each role; data is shown across all RCSNs. We see that 19 out of 38 (50%) of roles involved in emergent interactions are Developers, 13 (34.2%) are Testers and 6 (15.8%) are Technical Leaders. Note that none of the planned RCSNs included a Tester.

Table 2. Distribution of team members by RCSN per role

<table>
<thead>
<tr>
<th>Role</th>
<th>Team members in planned RCSN</th>
<th>Team members in actual RCSN</th>
<th>Emergent people in the actual RCSN (adds)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Developer</td>
<td>15 1 15 3 31 1</td>
<td>17 2</td>
<td></td>
</tr>
<tr>
<td>Tester</td>
<td>0 0 13 13 13 0</td>
<td>0 0</td>
<td></td>
</tr>
<tr>
<td>System Arch</td>
<td>13 13 13 13 13 13</td>
<td>0 0</td>
<td></td>
</tr>
<tr>
<td>Test Lead</td>
<td>13 8 13 13 13 14</td>
<td>0 6</td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td>41 22 71 30 30 8</td>
<td>30 8</td>
<td></td>
</tr>
</tbody>
</table>

What interactions and information exchange can be found in RCSNs? To identify patterns of collaboration and information exchange within RCSNs, we identified, from questionnaire data, who talked to whom and what information they exchanged within each RCSN. Every time a respondent indicated that he contacted a team member regarding the requirement, we counted one communication line between the two members and recorded it as a non-directed edge between two nodes in the respective RCSN. If a pair of team members reported that they communicated with each other, we counted only one line between them to avoid counting duplicated lines of communication.

As an example, Figure 1 shows the sociogram for the actual RCSN for R6. Each node represents a person, and an edge is an instance of communication between these people regarding the requirement. Each project member is identified by his/her name (fictitious, to protect anonymity), location (BR or US), and role. Project members included in the planned RCSN are identified between brackets and emergent people are identified between square brackets. The dotted lines indicate communication lines that were expected from the planned RCSN, and the solid lines indicate the communications reported in the actual RCSN. Details for the other RCSNs can be found online at segal.uvic.ca/collaborationpatterns.

A total of 140 unique lines of communication were reported. Of these, 43.6% (61/140) were cross-site communications and 37.1% (52/140) were emergent interactions. Of the emergent interactions, 32.7% (17/52) took place among cross-site members. The distribution of emergent interactions in total is 25% within sites (35/140) and 12.1% across sites (17/140). This indicates that a large amount of development work involves communication across sites, but that local interactions are more likely to emerge than remote interactions, suggesting that a project member seeks information more readily from a local colleague.

To characterize the information exchanged within RCSNs, we also included information provided by our respondents about the nature of interaction they had with the other project members working on same requirements. In Figure 1, the labels on the edges indicate the reasons for communication as provided in the questionnaire: Communication of Changes (“Change” in Charts 1 and 2, “C” in Figure 1), Coordination of Activities (Coord, CA), Implementation of Issues (Implem, I), Planning (Planning, P), Requirement Clarification (Clarif, RC), Risk Assessment (Risk, RA), Support (Support, S), Synchronization of Code (Sync, SC), or Other. For example, Kyle reported that reasons for interaction
with David as Communication of Changes (CC) and Coordination of activities (CA).

Figure 1. Actual RCSN for Requirement 6

Chart 1 and 2 report the number of times information on the nature of interaction within RCSNs was reported by the respondents. Respondents may have indicated several reasons for communication in the interaction with a particular person in the RCSN. Duplicate lines of communication between two team members were counted once.

Chart 1 reports the number of times the nature of interaction was reported for the planned networks and the emergent networks. A total of 499 reasons of interactions were reported, 330 between members of planned RCSNs and 169 between the adds in the actual RCSNs. From Chart 1, we see that the most often reported planned interaction was about the communication of changes (n=63/330), and that the most frequent emergent interaction was about the coordination of activities (n=40/169). Overall, communication of changes (n=86/499) and coordination of activities (n=85/499) were the most frequently discussed topics.

Chart 2 presents the reason for interactions characterized by location. Within sites indicates communication between two people at the same location, and across sites indicates communication between two people at different locations. Looking at this data, we see similar trends within and across locations. Within sites, interactions are mostly about coordination of activities (n=52/284) and communication of changes (n=50/284). Across sites, no single communication topic stood out.

5.2. Awareness within and across RCSN

To examine awareness within the requirements-centric networks, we asked the respondents if they know who in the project team is working on the same requirements they are allocated on. The answers were positive: Of the respondents, 85% were always or most of time aware of whom in the local project team is working on the same requirements, and 75% were always or most of time aware about the remote team members. Only 12.5% of the respondents were never aware of who is working on the same requirements in the remote team.

When asked how they identify these people, 44.4% of respondents indicated unplanned interactions, and while 22.2% mentioned group meetings. The remaining 33.4% stayed aware through task assignments or project documentation. This indicates that communication is the main source of information in the project team and reinforces the importance of face-to-face interactions.

An important aspect of staying aware of changes that take place within the project is to avoid rework. When asked whether the changes in the project were communicated in time to avoid rework 75% of respondents believed that always or most of time the changes are communicated in time. Although 25% of respondents believe the alerts are timely “because changes are immediately communicated”, 37.5% of them still believed that there were some issues about communication of changes (“changes take long to be communicated”, or “some project team members are not communicated”, or “changes are not communicated at the same time in both locations”). This indicates that, although the respondents believed the alerts were timely, there was still a feeling of gaps in communication.

Chart 1. Reason for interaction in Planned and Emergent communication lines

Chart 2. Reason for interaction within sites and across sites
Knowledge sharing across sites is also a known problem in global teams [5] as inadequate channeling of information, or incomplete information is communicated across sites [4]. We thus asked which information they currently receive, when requirements changes are communicated, as well as which information they would like to additionally receive to help them understand better which requirements have been changed at the remote site. They could select one or multiple reasons among: Requirement identification, Justification of the change, Schedule changes, and Other. Responses included requirement identification (75% of responses), schedule changes (62.5%) and justification of change (37.5%). Among the additional information they would find helpful, respondents indicated requirement identification (50%; an interesting finding since 75% already reported as receiving requirements), schedule changes (75%) and justification of change (75%). We can observe that the respondents already receive mostly the information desired, but there is room to improve the communication about “justification of change” and attend respondents’ expectations about this type of information.

**Relationship between distance, communication and awareness.** We performed multiple correlation tests between distance and the factors of communication frequency, general awareness and availability respectively. For the factor distance, we coded pairs of people as collocated if they were both working in the same country, and as remote if they were working in different countries. For the other factors, the dichotomized values from the questionnaire are as follows: Communication frequency as Less than 3 times a week/3 or more times a week, General awareness as Aware/Not aware (including N/A option), and Availability as Easy/Difficult.

**Table 3. Relationship between distance and communication, general awareness and availability**

| Variable             | beta  
|----------------------|-------
| Communication frequency | 0.426*  
| General awareness   | 0.07  
| Availability        | 0.254**  

n=10  *p<0.02, **p<0.05

The correlation is calculated using QAP (quadratic assignment procedure). Because network data is not independently measured, traditional parametric correlation methods cannot be used. Table 3 provides the results of the QAP correlation tests. We found a significant decline of communication frequency over distance (r=0.254 and p<0.05), meaning that the communication with the remote colleagues was less frequent than with the local ones, and that remote team members are more difficult to reach than the local ones respectively.

**Relationship between communication and awareness.** We further analyzed the relationship between communication frequency and the same two types of awareness. Data for the communication network was dichotomized the same way as described previously. Table 4 shows the QAP correlation between the communication frequency factor and the general awareness and availability networks. We found a significant decline of general awareness when frequency of communication was lower. This indicates that people were more likely to communicate with someone who they know can help in the work.

**Table 4 - Relationship between communication and general awareness and availability**

| Variable             | beta  
|----------------------|-------
| General awareness   | 0.302*  
| Availability        | 0.117  

n=10  * p<0.05

6. Discussion

The goal of our study was to explore collaboration in requirements-centric social networks, and in particular to discover whether distance affects it and one if its key aspects: task awareness.

By analyzing the RCSNs for each requirement, we identify a number of observable patterns, and discuss the implications for the design of collaborative tools with improved awareness support for distributed teams.

First, we found that each requirement’s RCSN was extremely dynamic and included important cross-site interactions. More people collaborated during the development of each requirement than was originally planned, and about one third of interactions in the team were with emergent team members. Whereas in our previous study of one distributed team [10] we identified that teams working on same requirement are dynamic, here we bring further evidence in this direction; we observed a general trend: all 13 RCSNs consistently had emergent interactions (see Appendix at segal.uvic.ca/collaborationpatterns ). In particular, a developer was most likely to be added to an RCSN, though technical leaders and testers were added as well. Interestingly, the tester was not involved in any planned RCSN, but appeared in every actual RCSN suggesting that the tester was always involved in the
requirements work for this project. Despite this omission, the team was aware of the tester’s planned participation in requirements work. Why was the tester omitted from planning documentation? Perhaps the project assumes that testers are involved in the work of each requirement, or maybe project managers forget to include testers in their plans, but the development team is aware of their participation. How did the team members know they had to interact with developers and testers not included in planned RCSN, especially if they were remote? We were not able to identify the reason they emerged in a particular RCSN. This prompts us to ask: Who are these developers, and why have they become involved in these interactions for particular requirements? Future research should try to understand more about current awareness mechanisms that allowed these members to know who to involve in their knowledge seeking interactions.

In each requirements network, communication of changes was found to be the major reason for interaction. Although changes consisted of such a large proportion of both intra-site and cross-site communications, 62.5% of the respondents believed that they did not receive requirements in a timely fashion. This highlights an area in which improved change awareness, by increasing its timeliness, may have a strong effect on collaboration. Future research should pursue the development of improved methods for communication of changes, both to improve the timeliness of change notifications and to reduce the communication overhead required to make team members aware of every change.

Our findings with respect to awareness in RCSNs also have important implications for requirements-centric collaboration and design of collaborative systems. A team member generally knew who else was working on the same requirements, regardless of geographical location. This was somewhat surprising, given the RCSN’s dynamic nature, and the significant number of emergent relationships in these networks. Furthermore, the results of the correlation between frequent communication and awareness suggest that those who communicated more were also more aware. When this is analyzed in light of findings that project members keep aware of each other through regular meetings (22.2%), or through the high proportion (44.4%) of unplanned interactions, the considerable reliance on verbal communication or local experts leads to research questions such as What type of local or verbal interaction facilitates the maintenance of this awareness? How can an awareness system replicate it in the distributed interaction? Which information from the development environment can be collected by such an awareness system automatically in order to supply it to the project members? At the same time, more investigation is also needed into the impact of other factors such as work (process) or ethnic culture on awareness. While we only sought to correlate awareness with communication in this study, it is also possible that awareness was also maintained as a result of certain procedures for knowledge dissemination in project meetings (process); or may have been hindered due to different communication styles across sites.

The impact of distance on the collaboration within these dynamic RCSNs is not clear-cut and rather interesting. Although we did not find evidence for a relationship between distance and whether or not a developer is aware of a remote team member who can help him on his requirements, distance did affect how frequent the communication was and how easy it was to contact this person. This provides a number of implications for knowledge sharing and expertise finding in RCSNs, as well as coordination of work on a particular requirement. Despite the fact that a developer may know which remote member to contact regarding the requirements he is working on, that person is perceived as being difficult to communicate with. Important research questions that emerge then are: How can awareness mechanisms provide information on availability of members related to one’s work, regardless of geographical distance? and What awareness information would help project members reach the remote colleagues more effectively?

6.1. Requirements-Dependency Awareness Network

The emergent interactions with the adds in the RCSNs and the findings of general awareness about members outside the planned RCSN prompted us to further investigate current awareness of work that is related, but tangential, to one’s work. Hence we conducted a post-hoc analysis of awareness within what we term a requirements-dependency awareness network (RDAN), and which include those members who work on inter-dependent requirements. For example, Figure 2 shows who works on the inter-related requirements R6 and R8 (role and affiliation to requirements is indicated in brackets and square brackets). In building this network we used responses on current awareness within RCSNs.

Awareness of others who work on inter-related requirements is important for expertise seeking, as well as for propagation of change information to those whose work depends on one’s work on particular requirements. Instead of displaying communication edges as in an RCSN, the RDAN indicates that a member reported presence of awareness (Aware and Very Aware) of members that work on inter-dependent
requirements. Although in Figure 1 the lines are non-directed, we chose to indicate who is aware of whom in Figure 2 adding arrows.

Because communication and awareness were correlated in our findings, and each member’s communication frequency is important for understanding potential for knowledge sharing within these related networks, we chose to further study these two factors together in RDANs. In this example, Table 5 shows the awareness and communication levels for members in Figure 2; we can also overlap an RCSN and an RDAN graphically to investigate the relationship between communication and awareness.

### Table 5. Awareness and communication levels for each member in Figure 2

<table>
<thead>
<tr>
<th>Team member</th>
<th>Awareness</th>
<th>Communication (Frequency)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Of others</td>
<td>That others have</td>
</tr>
<tr>
<td>Kyle</td>
<td>9</td>
<td>2</td>
</tr>
<tr>
<td>Mike</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>George</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>Jack</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>David</td>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td>Victor</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>Chase</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>Aeron</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>James</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>Charles</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

The visualization in RDAN and communication information in these networks is useful in studying gaps in awareness and communication that may prompt further investigations of practical or research nature:

**Centrality analysis.** Who are the central people who can bridge the two interrelated networks in terms of awareness and communication? Central people in terms of awareness—who is most aware of the others and whom are most aware of—are key project members in requirements expertise seeking and requirements knowledge dissemination. Exploring questions such as the following: Are the central people those who work on both requirements (R6 and R8 here)? If not, why? Answering these questions has implications with respect to flow of information and coordination between the two inter-related groups of people.

**Analysis of similarities between the awareness and communication networks.** While we found a positive correlation between communication and awareness within RCSNs, is there a correlation between awareness and communication within RDANs? Here, George is most central on both awareness and communication, but is there a global trend? Such a result may indicate that those who are most aware are also best communicators and thus knowledge disseminators to relevant project members. If not true, then other questions arise: What other means of knowledge dissemination besides communication is effective in maintaining awareness?

Because our data was limited (this analysis was not included in our original research design) and did not allow us to run tests for centrality or network similarity, we presented this example to illustrate the potential for study of awareness and communication in RDANs.

### 7. Conclusions

In this analysis of an industrial, distributed software project, we examined collaboration, awareness and distance in requirements-centric social networks. Our findings indicate organic patterns of collaboration involving considerable cross-site interaction, in which communication of changes was the most predominant reason for interaction. Although we did not find evidence that distance affects developers’ awareness of remote team members who work on the same requirements, distance affected how accessible the remote colleagues were.

Our findings that requirements management involves dynamic social networks that are different from those one would draw from the initial task-allocation plan has significant implications for ways in which collaboration should be supported in collocated and distributed development teams. Two directions for developing new awareness mechanisms emerge.
First, we must continue to facilitate general and current awareness, in order to cope with dynamic RCSNs, and awareness information sources are not always electronic. Any awareness mechanism will have to use other sources of information to understand the current status of the project. Requirements management is a complex task involving knowledge acquisition and sharing and it is the task dependencies as well as the technical expertise spread across multiple individuals that makes the effective collaboration and awareness within these RCSNs critical. Although expertise seeking and requesting assistance is typically supported by informal communication in co-located teams, distributed teams lack this communication depth and are left with the inability to know who works on the same requirements, and to whom changes should be propagated for effective cross-site coordination efforts. Second, a collaborative system should be able to facilitate unplanned collaborative work among software developers, as well as initial contact and continued collaboration among team members working on the same and inter-related requirements. Such a system must be able to (1) allow expertise finding when and such that these emerging interactions do occur, and (2) maintain task awareness among the members of RCSNs, without overloading them. Further, tools that create and possibly maintain RCSNs and RDANs automatically could help managers identify gaps in communication and awareness that prompt the need for improvements in process or project communication infrastructure.

7.1. Limitations and future work

While useful as a starting point for research in this direction, our study however was not without challenges and inherent limitations. First, coordination data and details of interactions in RCSNs in particular are hard to capture empirically. We relied on self-reported data based on developers’ recall of interactions. More automated means of collecting this data is desirable and we currently developing prototypes to parse chats and email logs for requirements-related interaction details. Capturing verbal communication remains a challenge.

Further, collecting awareness information is no easier endeavor. Not only measuring awareness is very difficult, but also reliance on self-reported data, again, may become problematic. One, levels of awareness may be different for different respondents. Second, developers may feel apprehensive of reporting lack of awareness when this could be seen as a sign of work underperformance, especially in offshoring relationships where maintaining business partnership is very important. In conclusion, we have provided important evidence for the benefits of examining collaboration and awareness within RCSNs and RDANs. Bringing more insight into what awareness information needs to be made available within one’s RCSNs for coordination and collaboration during the development of requirements in distributed teams is a fruitful area for future research. In our future work, we hope to further analyze awareness, study the effects of various factors on awareness within these networks, and develop tools and processes that can make team members more aware of requirements-related changes and developments in GSD.

Acknowledgements. Many thanks to the study participants and Kate Ehrlich and Urlike Stege for help with the social network analyses.

References