Exploring Cross-Site Networking in Large-Scale Distributed Projects

Aivars Sablis, Darja Smite, and Nils Brede Moe

Citation:
Sablis A., Smite D., Brede Moe N. (2018) Exploring Cross-Site Networking in Large-Scale Distributed Projects. In: Kuhrmann M. et al. (eds) Product-Focused Software Process Improvement. PROFES 2018. Lecture Notes in Computer Science, vol 11271. Springer
DOI: 10.1007/978-3-030-03673-7_23

Exploring Cross-Site Networking in Large-Scale Distributed Projects

Aivars Sablis[1], Darja Smite[1,2], and Nils Brede Moe[1,2]

[1] Blekinge Institute of Technology, Karlskrona, Blekinge 371 33, Sweden
aivars.sablis@bth.se
darja.smite@bth.se
[2] SINTEF, Trondheim, Norway
Nils.B.Moe@sintef.no

Abstract. Context: Networking in a distributed large-scale project is complex because of many reasons: time zone problems can make it challenging to reach remote contacts, teams rarely meet face-to-face which means that remote project members are often unfamiliar with each other, and applying activities for growing the network across sites is also challenging. At the same time, networking is one of the primary ways to share and receive knowledge and information important for developing software tasks and coordinating project activities.
Objective: The purpose of this paper is to explore the actual networks of teams working in large-scale distributed software development projects and project characteristics that might impact their need for networking.
Method: We conducted a multi-case study with three project cases in two companies, with software development teams as embedded units of analysis. We organized 20 individual interviews to characterize the development projects and surveyed 96 members from the total of 14 teams to draw the actual teams networks.
Results: Our results show that teams in large-scale projects network in order to acquire knowledge from experts, and to coordinate tasks with other teams. We also learned that regardless of project characteristics, networking between sites in distributed projects is relatively low.
Conclusions: Our study emphasizes the importance of networking. Therefore, we suggest that similar companies should pay extra attention for cultivating a networking culture in the large to strengthen their cross-site communication.

Keywords: Large-scale . Distributed . Software Development . Knowledge Networks . Coordination Networks.

1 Introduction

Large-scale distributed software development has shown a strong growth rate the last years. However, still, the success of organizing distributed work has been varying a lot [1]. The key reasons for varying success are the challenges

with communication and coordination in distributed development [2], and managing dependencies in large-scale projects [3].
Large-scale software development means working with a product that has millions of lines of code, multiple subsystems, and very long product lifecycles. No single developer or even a development team possess all the expertise needed for conducted for solving their development tasks [1]. Further, the necessity to divide development work between several teams in large-scale development creates dependencies between teams. Therefore, teams need to rely on their network within the project to solve dependencies and to solve complex tasks. "Who you know" affects "what you know", and a large network is more likely to help to solve development problems and handle dependencies between teams faster. As a consequence, networking - the exchange of tacit or explicit information and knowledge between project members, is of paramount importance [4]. At the same time, wrong assumptions about network and decision structure could lead to poorly made decisions impacted by a lack of necessary knowledge and even to devastating results [5] in extreme scenarios.
In a distributed large-scale project networking is complex, because time zone issues makes it hard to reach the remote partner or team, and growing the network across sites is problematic. In distributed projects, teams rarely meet face-to-face, and the lack of familiarity makes networking across sites challenging. Lack of familiarity with other sites means that, in practice, remote locations often work in isolation, especially when following modularized ways of working with a clear split of responsibilities. But what happens when work tasks are interdependent, or knowledge needed to complete a task resides at a remote location in another timezone? To the best of our knowledge, little is known about the importance of networking in large-scale distributed projects, including how teams rely on their networks, and the important actors in these networks. In-depth empirical cases of networking and coordination in large-scale distributed projects, in which work is shared and not modularized, are scarce. Motivated by the importance of knowledge networks in large-scale distributed software development we suggest the following research question:
RQ: What characterizes networking in large-scale distributed projects?

The remainder of the paper is organized as follows. Section 2 outlines related work. In Section 3, we describe our research methodology. In Section 4, we present our findings from the cases and cross-case analysis, which are further discussed in Section 5. Finally, Section 6 concludes the paper.

2 Related Work

In this section, we describe related research on need of networking in large-scale projects and how project members share task-related product knowledge and how coordinate the tasks with others.

2.1 Networking needs in large-scale distributed projects

Networking needs in large-scale distributed software development is affected by formidable challenges mainly in two fronts - scale and distribution.
First, large-scale software projects are challenging because of the challenges of managing portfolios and customer requirements, while at the same time coordinating and distributing work to many teams [6]. Because of the scale, many tasks such as systems management, verification and validation, integration etc. are centralized and performed by specialized support roles. This creates task interdependencies between teams and supporting roles, requiring task coordination and alignment to deliver the final product [7]. Similarly, the development tasks may have interdependencies, requiring teams to mutually coordinate.
Second, sheer expertise needed to do quality work in the product [1] is huge and often leads to knowledge fragmentation and necessity for coordination. Large-scale software development means working with a product that has millions of lines of code, multiple subsystems, and often a very long product lifecycle. That often will lead to the point where a single developer or even a development team does not possess all the expertise needed for their task [6], [1]. To address this need for task knowledge, inevitably networks of knowledgeable contacts are created between different teams and roles in the project. The importance of these networks grows along with the growth of the product.
To successfully develop tasks, software developers have to rely on both their own and their team members knowledge and the knowledge and information pulled from their social network outside the team [8]. Well-connected members have efficient access to valuable knowledge and coordinate task dependencies with others more effectively.
The complexity of large-scale distributed software projects has been traditionally minimized by chunking the work into manageable pieces (moduls) and minimizing the interaction between the teams and sites [9]. This often results in knowledge differentiation, which reduces the burden of knowing everything. Experiences show that modularization is insufficient since modules are rarely isolated and cross-module interfaces require cross-team and cross-site interaction [10].

2.2 Experts in Networks

Networking - the exchange of knowledge and coordination between project members, is of paramount importance in large-scale projects. Research on organizational learning has demonstrated the implications of knowledge and information coordination on performance [11], and the importance of social relationships when acquiring such information [12]. We know that social network perspective can help understand collective information exchange in organizations [11].
Networks similar to those in large-scale distributed development, in which some know more than others, have also been previously addressed. Social capital has been often discussed together with social networks. Social capital is described

as "advantages that individuals or groups have because of their location in social network" [12]. Current research on social capital determines that those who are better positioned or active in a network are able to solve tasks with better quality and also influence others [12], [13]. These are usually more senior contacts, or formal and informal experts, and are recognized as contacts that are sought by the teams in software development [8]. Similarly, certain project members have to be more active in networking due to the necessity to coordinate task-related dependencies. In the large-scale distributed environment there might be hundreds of people, thus understanding how to establish and cultivate valuable contact networks is both challenging and of paramount importance. Also, not all information and knowledge plays the same role. While two persons in a network might be connected, it is critical for them to exchange relevant information about a task for their connection to be beneficial (the one to capitalize upon) [12]. Finding the right experts and creating connections with them in the network requires time and effort [13].

3 Methodology

Since in-depth empirical cases of networking and coordination in large-scale distributed projects are scare, we used an exploratory approach in our research study [14] to understand to what characterizes networking in large-scale distributed projects. We conducted an embedded case multi-case study where the unit of analysis was a development team in large-scale distributed software development project [14]. In our exploratory research we relied on theoretical sampling to select cases. We have studied large-scale software development projects in two companies, Company A (Project A and Project B) and Company B (Project C), and we selected projects with overlapping and complementary characteristics for the case study to explore what characterizes networking in large-scale software development projects.

3.1 Data collection

Qualitative data was collected through individual interviews with project members in each of the companies. Interviews followed a structured guide, where management representatives and persons occupying supporting roles were asked directly about project characteristics to clarify team responsibilities, and the type and complexity of team tasks in the studied projects. All interviews in the three projects were recorded and later transcribed. Most interviews were organized on onshore sites, but in two Project A and Project C also in offshore sites. Most of the interviews were organized face-to-face, and all held in English. In total, we organized 20 individual interviews, see Table 1.
Quantitative data was collected using a questionnaire. We decided to use a survey as it is a common method for empirical evidence in social network analysis [15] and also is recognized in software engineering [16] as a means of

Table 1. Data collection

Activity	*Location*	*Time*	*Participants*	*Researchers*
Company A - Project A				
Survey	Sweden	Apr2014	3 development teams, 16 participants	A1,A2,A3
	China	Mar2014	3 development teams, 19 participants	A1,A2,A3
Interviews	Sweden	Aug2013	7 interviews	A2,A3
Company A - Project B				
Survey	Sweden	Nov2014	2 development teams, 17 participants	A1,A3
	China	Nov2014	2 development teams, 12 participants	A1,A3
Interviews	Sweden	Nov2014	4 interviews	A2,A3
Company B - Project C				
Survey	Sweden1	Feb2014	2 development teams, 9 participants	A1,A2,A3
	Sweden2	Feb2014	1 development teams, 5 participants	A1,A2,A3
	India	Jul2014	1 development group, 18 participants	A1
Interviews	Sweden1	Aug2013	6 interviews	A2,A3
	Sweden2	Feb2014	3 interviews	A1,A2

* This column refers to the authors participation in the data collection; the first author is A1, the second author is A2, and so forth.

systematic relation data collection [15]. We conducted a survey with 14 teams. The respondents were asked to report their knowledgesharing network in a free recall manner. We have specifically inquired to identify persons whom the respondents transfer the project-related knowledge to, or retrieve the knowledge from, as well as the nature and content of the knowledge transferred or retrieved. This survey instrument is described in detail in [8]. The survey responses were collected in two ways: with the help of a web-based tool (Project A onshore site, Project B offshore site) and manually using paper printouts (Project A offshore site, Project B onshore site, Project C onshore sites and offshore site). Participants in paper-based surveys were given around 30 minutes to fill in responses for as many contacts as they felt needed in free-recall format. Participants in the web-based survey were given two days to fill in responses for as many contacts as they felt needed in a free-recall format. Longer time was required due to the need of the participants to allocate time in their daily work schedule. We completed the list of recalled contacts and clarified these with company representatives to clarify the roles of the reported contacts in the company and their office location. In total members of 14 teams participated in the study for a total of 96 participants, see see Table 1. Actual team sizes ranged from 5 to 18.

3.2 Data analysis

To understand what characterizes networks in large-scale projects, we seek an understanding of networking in terms of contacts and connections, and the content flowing through the network. In particular, we want to know who are the key contacts in the network in two categories, those who share task-related product knowledge with others and those who help coordinate the tasks with others.
Project characteristics: We use data gathered in interviews to classify different project characteristics. Each of these characteristics have potential influence on the way information exchange is done in projects social networks.
Networks: All reported connections were classified into three categories based on the reported content: task-related product knowledge, task coordination (coordination of task interdependencies and alignment-related coordination) and administrative information (the rest of connections). We calculated the number of connections between project sites and within each site. Further, we created graph diagrams to visualize connection categories, surveyed team members, their reported contacts and their affiliation with onshore or offshore sites.
Key contacts in network: We used multiple social network analysis measures to understand who the key contacts in network are. We did this, because there is no single measure that would precisely determine the central contacts in a network, because centrality can be attributed to different aspects [17]. Thus, our aim is to combine multiple measures and therefore triangulate different aspects of network "centrality". We calculate three measures - Degree, Betweenness and Centrality [18].
- Degree (Degr.) measures the number of connections each contact has. The more connection a contact has, the larger the degree. In our case, for Task knowledge we are using Out Degree capturing contacts transferring knowledge to others. The more knowledge a contact transfers, the greater is the value of that contact in the network [19][18].
- Closeness centrality (Clos.) - measures how many steps are required to access every other vertex (in this case: contact) from a given vertex. Larger closeness centrality measures might indicate that the contact is in a relatively central position in the network, which increases the chance of exchanging the information in a timely manner [19][18].
- Betweenness (Betw.) - measures the number of geodesics (shortest paths) going through a vertex. Larger betweenness scores might indicate that the contact holds a relatively unique position in the network in relation to the ability to transfer knowledge to different sub-groups in that network, which creates opportunities for information control benefits [19][18].
We rank each of these three measures and aggregate them in the final score and are calculated separately for task-related product knowledge and task coordination connections. Role-wise, we classified all network contacts according to a previously defined schema [20] into five different categories: Line Management (LM), Product Management (PM), Feature Product Development (FDS) and Development Support (DS) and Developers and Testers from teams (OD).

4 Results

In this Section, we start with a presentation of the three cases. Then we visualize the networks and present the main network measures and share the key findings related to overall structures of project networks. Lastly, we present our findings of key contacts in each of the cases.

4.1 Project cases

All three selected projects are large-scale projects delivering complex software-intensive sub-systems or entire systems, all development efforts are distributed across two or more sites. The main characteristics, similarities and differences between the projects are described in Table 2.

Table 2. Project cases

	Project A	*Project B*	*Project C*
Product type, maturity	Sub-system of a large software product	Sub-system of a large software product	A software solution
Methodology	Agile	Agile	V-model
Size and complexity	Very complex, contains several millions LOC	Very complex, contains several millions LOC	Large, safety-critical and very complex embedded software system
Product architecture	Contains multiple components, layered architecture, interacts with other sub-systems	Contains multiple components, layered architecture, interacts with other sub-systems	Contains multiple interrelated modules
Project distribution	5 teams in Sweden, 8 teams in China, 2 homeshore teams in China, 2 teams in Korea	4 teams in Sweden and 5 teams in China	6 teams in the main site in Sweden, 1 team in the secondary Swedish location, 1 group in India
Team setup	Cross-functional agile feature teams	Cross-functional agile feature teams	Disciplined component teams

4.2 Networks

Project A network is visualized in Figure 1. In Table 3 we have calculated all connections by type within and between sites. Results show that in Project A site networks are clustered. Most of the knowledge and task coordination connections is done within sites. Further results show that compared to onshore teams, offshore teams with less experience have more knowledge connections (45.5% of all reported connections, as seen in Table 3), thus indicating that offshore teams are pulling the knowledge necessary for task completion. When it comes to task coordination, these primarily occur among the onshore teams and supporting roles (48.7% of all reported connections).

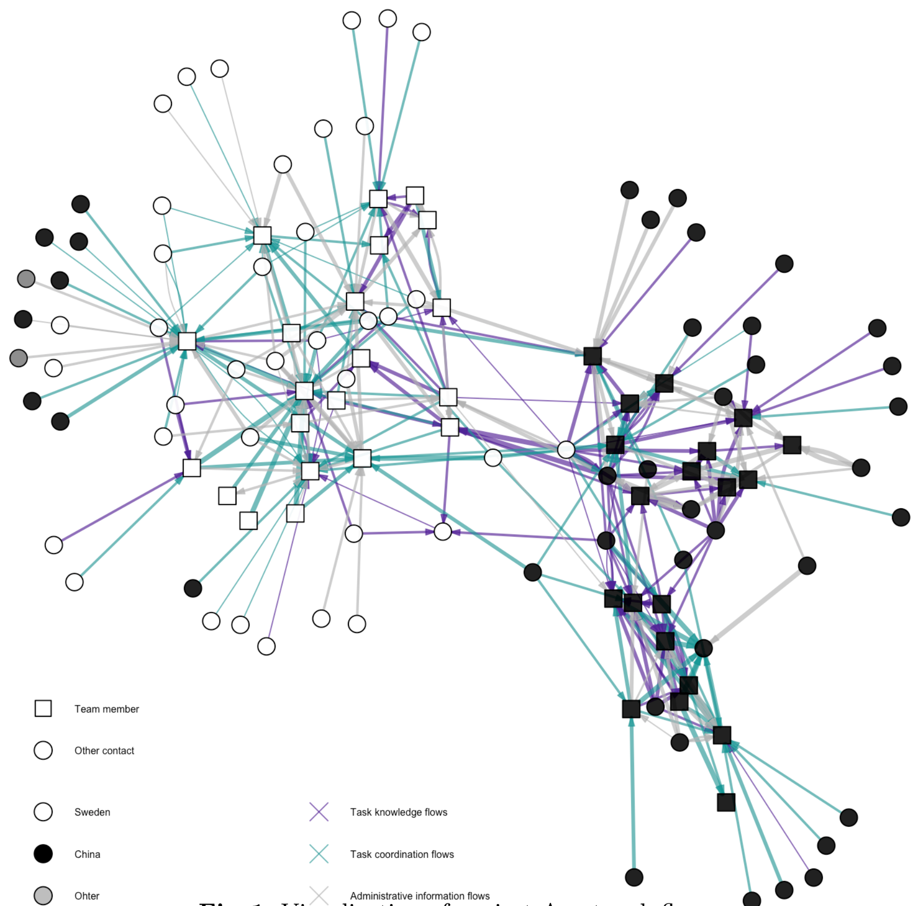

Fig. 1. Visualization of project A network flows

Table 3. All connections between and within sites - Project A

From	To	Know.	Know. %	Coord.	Coord. %	Admin.	Admin. %	Total
Sweden	Sweden	32	27.40	57	48.70	28	23.90	117
Sweden	China	11	57.90	7	36.80	1	5.30	19
China	Sweden	2	18.20	8	72.70	1	9.10	11
China	China	61	45.50	40	29.90	33	24.60	134

Project B network is visualized in Figure 2. In Table 4 we have calculated all connections by type within and between sites. Results show that in Project B site networks are also clustered. Both knowledge and task coordination connections occur within a site. Further, the results show that the Swedish site has equal distribution of communication for both knowledge exchange and task coordination, reported at 37.1% and 39.0% respectively. While the results show that task coordination scores high in the offshore site (48.2% of all reported connections), it is largely tunneled by a single key contact, as discussed in next section.

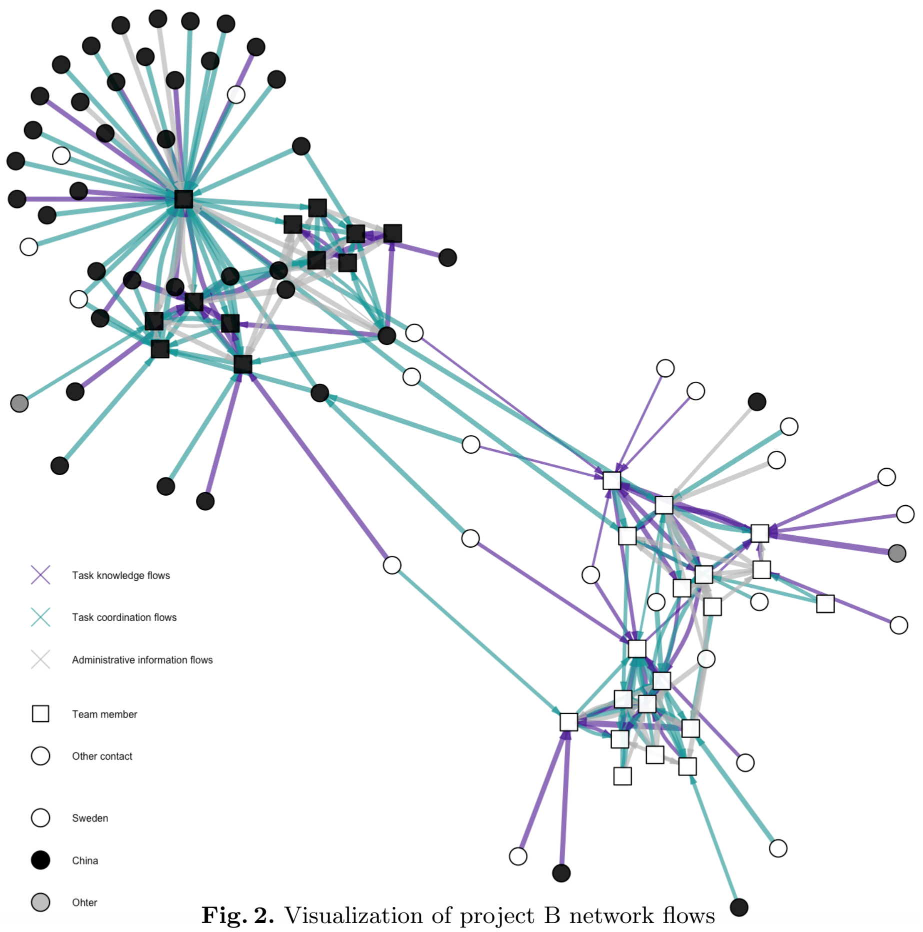

Fig. 2. Visualization of project B network flows

Project C network is visualized in Figure 3. In Table 5 we have calculated all connections by type within and between sites. Results show that similarly to other projects, site networks in Project C are clustered, and there is a low number of connections between sites. There are more connections between the two

Table 4. All connections between and within sites - Project B

From	To	Know.	Know. %	Coord.	Coord. %	Admin.	Admin. %	Total
Sweden	Sweden	39	37.10	41	39.00	25	23.80	105
Sweden	China	1	10.00	9	90.00	0	0.00	10
China	Sweden	1	25.00	2	50.00	1	25.00	4
China	China	29	26.40	53	48.20	28	25.50	110

Swedish locations, while the offshore site in India is connected via few connections apart, primarily tunneled through one key contact. In one of the Swedish locations (1) and the offshore site the largest percentage of classified connections are knowledge connections.

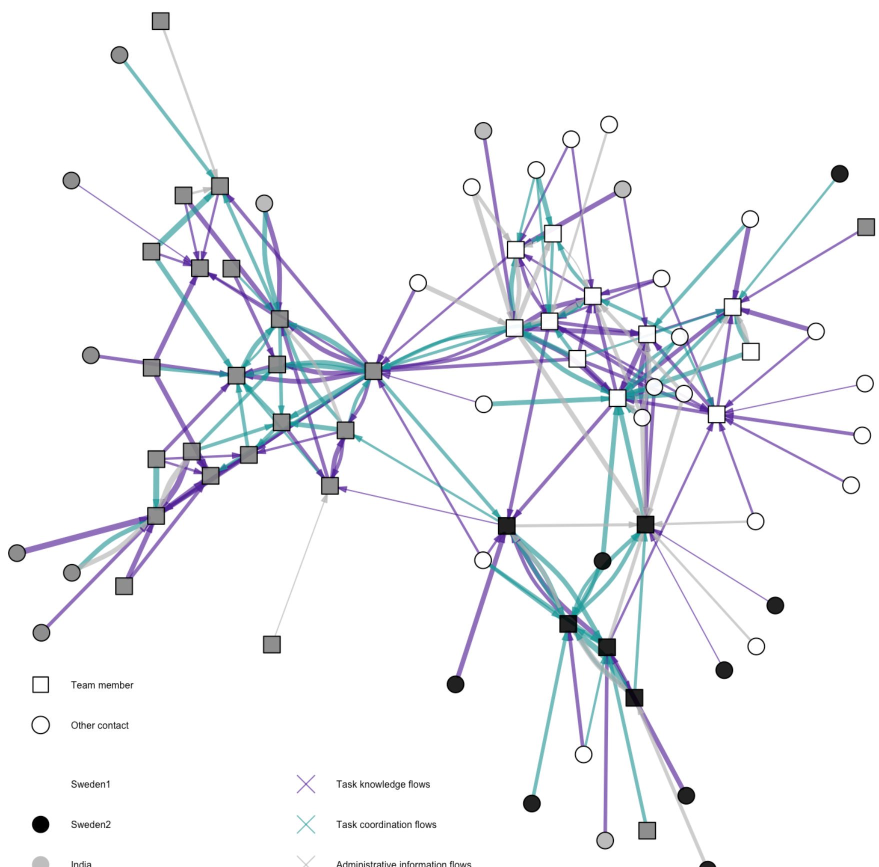

Fig. 3. Visualization of project C network flows

Table 5. All connections between and within sites - Project C

From	To	Know.	Know. %	Coord.	Coord. %	Admin.	Admin. %	Total
India	India	36	53.70	25	37.30	6	9.00	67
India	Sweden1	1	100.00	0	0.00	0	0.00	1
India	Sweden2	0	0.00	2	100.00	0	0.00	2
Sweden1	India	7	70.00	3	30.00	0	0.00	10
Sweden1	Sweden1	34	45.30	26	34.70	15	20.00	75
Sweden1	Sweden2	5	38.50	3	23.10	5	38.50	13
Sweden2	India	1	50.00	1	50.00	0	0.00	2
Sweden2	Sweden1	2	40.00	3	60.00	0	0.00	5
Sweden2	Sweden2	7	29.20	12	50.00	5	20.80	24

4.3 Key contacts in networks

Table 6. Top 7 key contacts for knowledge in each project

	id	Role	Role Cat.	Location	Degr.	Rank	Betw.	Rank	Clos.	Rank	Total Rank
Project A	s56	Technical Expert / Architect	PM	China	9.00	2	0.17	3	24.88	2	1
	s51	Technical Expert / Architect	PM	Sweden	10.00	1	0.00	13	29.19	1	2
	s84	Technical Expert / Architect	PM	Sweden	3.00	9	0.47	1	19.98	8	3
	s59	System Manager	FDS	China	9.00	2	0.00	13	23.22	3	4
	s38	Technical Expert / Architect	PM	China	9.00	2	0.00	13	22.69	4	5
	s05	Technical Expert / Architect	PM	China	7.00	5	0.06	11	20.39	6	6
	s80	Technical Expert / Architect	PM	Sweden	5.00	6	0.00	13	20.39	6	7
Project B	s57	Technical Expert / Architect	PM	Sweden	4.00	1	0.00	14	21.22	1	1
	s56	Technical Expert / Architect	PM	Sweden	3.00	2	0.31	7	20.38	8	1
	s07	Developer	OD	Sweden	2.00	5	0.00	14	20.73	3	3
	s06	Developer	OD	Sweden	2.00	5	0.00	14	20.69	4	4
	s48	System Manager	FDS	Sweden	3.00	2	2.55	1	19.97	21	5
	s05	Developer	OD	Sweden	2.00	5	0.69	5	20.32	16	6
	s58	Technical Expert / Architect	PM	Sweden	2.00	5	0.00	14	20.36	9	6
Project C	s20	Developer	OD	Sweden2	3.00	5	1.17	11	24.48	5	1
	s14	Team Leader	OD	Sweden1	4.00	3	0.56	14	24.19	7	2
	s36	Expert	PM	Sweden1	3.00	5	0.00	19	24.71	3	3
	s28	Developer	OD	Sweden1	4.00	3	0.00	19	24.21	6	4
	s25	Developer	OD	Sweden1	3.00	5	5.87	1	23.45	23	5
	s15	Developer	OD	Sweden1	6.00	2	1.63	9	23.69	20	6
	s57	Technical Coordinator	FDS	India	8.00	1	5.83	2	19.99	29	7

Key contacts in Project A network: Results show that the main role categories for transferring knowledge are Product Management (six contacts) and Feature Development Support (one contact), listed in Table 6 In this project, all Technical Experts/Architects are designated and formally appointed roles. Further, three key contacts are from the onshore site and four from the offshore site. Results show that the knowledge that is transferred by the key contacts are the knowledge about the architecture and design of the product and other related sub-systems. In a few cases, key contacts are also experts in other, related sub-systems and then key knowledge transferred is about the other sub-system. Our results show that the main role categories for task coordination are developers from teams and development support roles, listed in Table 7. Further, five

Table 7. Top 7 key contacts for coordination in each project

	id	Role	Role Cat.	Location	Degr.	Rank	Betw.	Rank	Clos.	Rank	Total Rank
Project A	s06	Developer	OD	Sweden	14.00	1	1.10	1	15.15	8	1
	s42	Developer	OD	China	8.00	6	0.49	3	19.81	2	2
	s34	Test Support Specialist	DS	China	11.00	3	0.13	5	15.61	7	3
	s104	Developer	OD	Sweden	6.00	8	0.62	2	16.89	5	4
	s21	Developer	OD	Sweden	12.00	2	0.00	13	18.13	3	5
	s76	Tester	OD	Sweden	9.00	5	0.15	4	14.08	10	6
	s87	Developer	OD	Sweden	10.00	4	0.00	13	17.49	4	7
Project B	s45	Scrum Master	LM	China	25.00	1	17.35	1	35.35	15	1
	s48	System Manager	FDS	Sweden	6.00	4	1.81	10	70.97	3	2
	s12	Developer	OD	Sweden	4.00	6	2.45	7	40.22	9	3
	s03	Developer	OD	Sweden	4.00	6	1.38	13	56.70	5	4
	s57	Technical Expert / Architect	PM	Sweden	3.00	9	10.51	4	36.63	13	5
	s77	Tester	OD	China	10.00	2	1.84	9	34.77	16	6
	s07	Developer	OD	Sweden	3.00	9	1.70	11	44.35	7	7
Project C	s63	Developer	OD	India	5.00	4	2.10	6	47.52	2	1
	s62	Developer	OD	India	4.00	5	1.54	7	47.33	3	2
	s23	Developer	OD	Sweden2	7.00	2	5.32	3	32.66	11	3
	s57	Technical Coordinator	FDS	India	4.00	5	6.17	1	32.55	12	4
	s76	Developer	OD	India	7.00	2	0.83	12	47.24	4	5
	s25	Developer	OD	Sweden1	10.00	1	0.00	15	42.15	6	6
	s15	Developer	OD	Sweden1	4.00	5	4.26	4	24.34	16	7

key contacts are from the onshore site and two key contacts are from offshore site. Results show that designated team members are responsible for task coordination with different supporting roles regarding responses to team development task proposals, progress of testing and test results, test environment tools and coordinating progress.

Key contacts in Project B network: Results show that the main role categories for transferring knowledge are Product Management (three contacts) and Developers from teams (three contacts), listed in listed in Table 6. While in this project Technical Experts-Architects are designated and formally appointed roles, this role is only one of many responsibilities without clear dedicated time and complained about being overloaded. Interestingly, all key contacts for task coordination that are reported are from onshore site. Results show, that the knowledge that is transferred by the key contacts is about product behavior, legacy, system-specific and testing-related knowledge.

Our results also show that in this project, task coordination is done by a variety of roles, including Technical Expert/Architects, Scrum Masters and Developers from teams, listed in Table 7. However, all of these key contacts are a part of teams, being part-time members. Further, five key contacts are from the onshore site and, two key contacts are from the offshore site. Results show that the key contacts coordinate information related to testing and test results, troubleshooting progress, tips and tricks. Also, in this project, two of the key contacts appear in both categories.

Key contacts in Project C network: Results show that the main role categories for transferring the knowledge are experienced developers from teams or team leaders (five contacts), while also showing that formally recognized experts (Expert and Technical Coordinator) are the key contacts, listed in Table 6.

Further, five key contacts are from prime location site, one key contact is from another Swedish site, and one is from offshore site. Main knowledge transferred from key contacts are specific issues with a respective module, such as framework architecture, protocols, and module functionality.
Our results show that the main role categories for task coordination are developers from teams or team leaders (five contacts). Interestingly, four of key contacts are from the offshore site, listed in Table 7. Reported coordination content from these contacts is code reviews, testing procedures, and general process related issues, while contacts from both Swedish sites reported work coordination and coordination about specific issues.
Also, in this project, three key contacts appear in both categories, showing that experienced developers from teams are both key contacts for task knowledge and task coordination.

5 Discussion

We have studied three large-scale distributed projects and explored the im- portance of networking in this context, including the content of network connections, and the important actors in these networks. Establishing and cultivating valuable contact networks is both challenging and of paramount importance for enabling team coordination [7] and knowledge exchange [1] in large-scale distributed projects, in which networks consist of hundreds of people spread across multiple geographic locations. In the following, we discuss three cases in the light of our research question:
What characterizes networking in large-scale distributed projects?

5.1 Networking in large-scale distributed projects

Our work contributes to the debate on how to set up communication and coordination in large-scale distributed projects [2], [9] which argues that cross-site coordination is challenging and that information is likely to circulate within one physical location. Our findings from studying networks in three large-scale distributed development efforts confirms that networking between sites is relatively low (see Table 3, 4, 5) and site networks are clustered (illustrated in Figures 1, 2, 3). In all three companies, despite the differences in project characteristics, cross-site coordination was limited. This was intentional. In Projects A and B, the company intended to assign interdependent feature tasks within the same location, and make tasks allocation across site as independent as possible. In Project C teams were responsible for components, and relatively interdependent components were assigned to teams within a short distance (separate floors or cities within the same country). Evidently, cross-site coordination between the two Swedish site was relatively high, even though lower than within the same location. Our results indicate that sameshore networking is thus possible, and in Project C was fostered by extensive traveling and face-to-face visits, and a long history of mutual collaboration. When it comes to knowledge exchange, we found that remote offshore locations in all three cases depended on the onshore competence, which was the main purpose for cross-site networking. In Project A dedicated experts were found to share their knowledge with teams from all sites. In Project B, where formally recognized experts were assigned to this role as only one of many of responsibilities, knowledge exchange between sites was very limited. In Project C, knowledge exchange was tunneled by a technical expert, who was familiar with the Swedish contacts due to his long history in the company, and accumulated contact network during onsite visits.

Finally, we suggest that software companies have to be aware that because cross-site knowledge exchange and task coordination are challenging, they require dedicated resources, extensive travel and time to bridge the geographical and temporal distances. Similarly to our case companies, software companies may decide whether to invest in networking or use approaches that limit the need for cross-site communication and coordination. In our cases, companies tried to minimize dependencies, and this was confirmed in the results. However, to realize such a structure and still ensure even knowledge levels in all sites and coordination across sites the experts are important. When software companies decide to invest in networking, it is the experts that require dedicated time and need to be available for all sites. The companies need enough experts, so the experts do not become a bottleneck or become vulnerable if an expert leaves.

5.2 Key contacts in network

Finding the right contacts and creating valuable connections in large-scale projects takes time and effort [13]. Previous research suggests that some contacts excel in networking more than others, and thus may gain a better position in a network [12]. Based on our analysis of the key contacts (summarized in Tables 6 and Table 7), we suggest that key contacts for knowledge exchange are either dedicated formal experts or informal experts (experienced team members). In Project A and B, there were dedicated technical experts to connect the offshore sites (i.e. the Swedish sites with the offshore sites in China), while the knowledge exchange across the two Swedish locations in Project C was done by very experienced team members who have visited each other often or have a long history of working together.

When it comes to task coordination, the network is not as centralized and instead is more evenly connected. At the same time, in all cases, we have observed coordination champions, i.e. few team members with a large network of contacts with whom they coordinate teams task dependencies, including supporting roles and contacts from other teams.

Interestingly, in Projects B and C, we found that some of the contacts appear as key for both knowledge exchange and coordination. This might be because few people actually traveled across sites, and established remote connections.

Furthermore, when discussing our findings with the cases, the project representatives revealed that these emerging contacts, who once gained the key position in the project network, remain acting as the key cross-site coordinators, and often complain about being overloaded. We, therefore, suggest that software companies shall be aware of the actual needs for knowledge exchange and coordination across sites, and actively support the teams by dedicating a sufficient number of experts and coordination champions and give them time to network.

5.3 Implications and limitations

One may wonder what can we expect in other large-scale distributed projects? The generalizability of our work is, of course, impacted by our sampling strategy and the cases included in our investigation. We studied networks in three large-scale development projects in two companies, different in project structure and characteristics. We suspect that the intensity of the networks and the key positions in each individual case will depend on the various project characteristics, such as the task allocation approach, modularization initiatives, formal choice of technical experts and coordination champions. At the same time, we can with large certainty say that cross-site coordination on a large distance in the majority of such cases will be limited, as also shown in previous research [4], [6], [2]. In our work we have specifically distinguished between knowledge exchange flows and coordination flows, which helped us gain a better understanding of the content and the needs for coordination in a large-scale project. Our results show that the importance of one or the other type of connections can be impacted by the maturity of sites and teams, ways of working (including project methodology) and project architecture and distribution. However, further research is needed to establish better explanations of the underlying reasons and causality. Specifically, we would like to suggest looking finding more cases to compare networking in agile versus non-agile projects.

6 Conclusions

In this paper, we shared the findings from an exploratory study of the role of networking in three large-scale distributed development projects. In response to our research question (What characterizes networking in large-scale distributed projects?), we identified limited cross-site networking, and that networks in large-scale distributed projects are geographically clustered in all three cases. At the same time, we learned that cross-site networking can be established, if the geographic distance is not large, and the sites have a long history of collaboration. We also found that knowledge networks are centered around formal and informal experts, while coordination networks are more equally connected and are performed by mutual coordination among the teams. Further research is needed to better understand the impact of networking on team performance in large-scale projects.

References

1. Moe, N.B., Šmite, D., Hanssen, G.K., Barney, H.: From offshore outsourcing to insourcing and partnerships: four failed outsourcing attempts. Empirical Software Engineering 19(5), 1225–1258 (2014)
2. Herbsleb, J.D., Mockus, A.: An empirical study of speed and communication in globally distributed software development. IEEE Transactions on software engineering 29(6), 481–494 (2003)
3. Flyvbjerg, B.: What you should know about megaprojects and why: An overview. Project management journal 45(2), 6–19 (2014)
4. Dingsøyr, T., Moe, N.B.: Research challenges in large-scale agile software development. ACM SIGSOFT Software Engineering Notes 38(5), 38–39 (2013)
5. Garner, J.T.: It's not what you know: A transactive memory analysis of knowledge networks at nasa. Journal of technical writing and communication 36(4), 329–351 (2006)
6. Dingsoyr, T., Smite, D.: Managing knowledge in global software development projects. IT Professional 16(1), 22–29 (2014)
7. Van de Ven, A.H., Delbecq, A.L., Koenig Jr, R.: Determinants of coordination modes within organizations. American sociological review pp. 322–338 (1976)
8. Šmite, D., Moe, N.B., Šāblis, A., Wohlin, C.: Software teams and their knowledge networks in large-scale software development. Information and Software Technology 86, 71–86 (2017)
9. Mockus, A., Weiss, D.M.: Globalization by chunking: a quantitative approach. IEEE software 18(2), 30–37 (2001)
10. Cataldo, M., Bass, M., Herbsleb, J.D., Bass, L.: On coordination mechanisms in global software development. In: Global Software Engineering, 2007. ICGSE 2007. Second IEEE International Conference on. pp. 71–80. IEEE (2007)
11. Borgatti, S.P., Cross, R.: A relational view of information seeking and learning in social networks. Management science 49(4), 432–445 (2003)
12. Burt, R.S.: The network structure of social capital. Research in organizational behavior 22, 345–423 (2000)
13. Nahapiet, J., Ghoshal, S.: Social capital, intellectual capital, and the organizational advantage. In: Knowledge and social capital, pp. 119–157. Elsevier (2000)
14. Runeson, P., Host, M., Rainer, A., Regnell, B.: Case study research in software engineering: Guidelines and examples. John Wiley & Sons (2012)
15. Marsden, P.V.: Survey methods for network data. The SAGE handbook of social network analysis 25, 370–388 (2011)
16. Sjoberg, D.I., Dyba, T., Jorgensen, M.: The future of empirical methods in software engineering research. In: Future of Software Engineering, 2007. FOSE'07. pp. 358–378. IEEE (2007)
17. Borgatti, S.P., Everett, M.G.: A graph-theoretic perspective on centrality. Social networks 28(4), 466–484 (2006)
18. Freeman, L.C.: A set of measures of centrality based on betweenness. Sociometry pp. 35–41 (1977)
19. Borgatti, S.P., Jones, C., Everett, M.G.: Network measures of social capital. Connections 21(2), 27–36 (1998)
20. Šāblis, A., Šmite, D.: Agile teams in large-scale distributed context: Isolated or connected? In: Proceedings of the Scientific Workshop Proceedings of XP2016. p. 10. ACM (2016)