

Bridging Knowledge Distribution – The Role of Knowledge Brokers in Distributed Software Development Teams

Alexander Boden

University of Siegen, Germany
alexander.boden@uni-siegen.de

Gabriela Avram

University of Limerick, Ireland
gabriela.avram@ul.ie

Abstract

Software development requires the handling of complex and context specific knowledge to be successful. Hence, efficient knowledge management (KM) counts amongst the most important challenges for any software project, but especially for small enterprises working with distributed teams. One important topic for KM in distributed teams is the role of “bridges” enacted by people who become boundary spanners and facilitate the exchange of knowledge between the sites. In our paper we present empirical findings related to such bridges in the context of two small companies with offshore sites. In doing so, we concentrate on the particular roles these knowledge brokers play in the distributed development practices. We show how small software companies rely on the commitment of particular team members and informal knowledge management practices. The paper concludes with a number of open questions to be addressed by future studies.

1. Introduction

Software development is a creative and knowledge intensive practice. Every software project is more or less unique, depending on multiple factors of which knowledge management (KM) is a very important one. Amongst others, KM has a major influence on team performance and hence on the quality of the resulting software. Being a challenge even for co-located teams, KM can get much more difficult in global contexts. Distributed teams have to cope with a multiplicity of barriers, which can slow down the development pace and result in poor software quality.

One of the KM-related topics in Software Engineering is the role of “bridges” in facilitating collaboration and coordination across distributed locations [1]. These bridges are enacted by people who become “boundary spanners” and fill the structural holes of social networks in distributed organizations. However, the role of bridges is not very well understood yet, as few of

them have been studied “in situ” using other techniques than interviews. We want to contribute to the understanding of Global Software Engineering by comparing the concept of bridges in the Software Engineering literature against our ethnographic data, which includes interviews, on-site observations as well as artifact analyses in the context of small-sized distributed software teams. Based on several examples, we illustrate how small software companies organize their knowledge exchange and which role bridges play for their work practices. We conclude by summarizing challenges for managing knowledge bridges which may need further attention in future studies.

The paper is organized as follows: section 2 discusses briefly the related literature, especially Milewski et al.’s framework. Section 3 describes our methodology, followed by a discussion of our cases in section 4. Section 5 presents our empirical findings introducing a series of knowledge work practices where knowledge brokers play an important role. Section 6 discusses our findings and section 7 concludes the paper.

2. Related Work

Knowledge Management (KM) is “a method that simplifies the process of sharing, distributing, creating, capturing and understanding of a company’s knowledge” [2]. KM as a field is highly interdisciplinary and KM research is known as relying on a broad set of different theories and research methods. In regard to Software Engineering, there has been a dominance of *technocratic* knowledge management approaches supporting a view considering knowledge as being a possession that can be de-contextualized, captured, and disseminated through information systems without a loss of meaning [3]—which is kind of problematic. While this perspective may be partially acceptable for traditional software engineering approaches, the growing field of agile development requires different knowledge management strategies with a stronger fo-

cus on behavioral approaches and knowing-in-action [4].

Agile development methods have a growing impact on software development organizations, especially in the case of small and medium enterprises (SMEs). SMEs tend to rely more on unstructured development methods with flat hierarchies and little specialization, than on formal development models. Hence, they tend to follow informal ways of dealing with knowledge, including less documentation and codification, while focusing on social interaction within teams and collaboration with the customer [cf. 5]. However, in distributed settings it can be very challenging to deal with the related organizational, temporal, spatial, legal, national and cultural barriers.

In regard to distributed collaboration and coordination, Milewski et al. [1] have presented the concept of bridges from a social network perspective. In their view, human actors play key roles in social networks, influencing the fate of software development projects. Different sources name these people “information brokers”, “boundary spanners”, “gatekeepers”, or “cultural liaisons”. These roles are usually not bestowed formally, although their importance has been noticed by both practitioners and researchers. Rather, bridges are facilitated by people who are acting naturally for managing and mediating communication and filling the structural holes in social networks. Usually they work with both sides, visit the remote sites and spend time working there and are often expatriates who have lived in different countries and experienced different cultures. Knowledge brokers rely a lot on their own social skills and on the social relationships they build in time. Their contribution becomes more important when teams are confronted with unusual challenges, like in the ever-changing field of software development.

3. Methodology

For the two case studies presented in this paper, the two researchers adopted similar approaches, relying on qualitative ethnographic methods and an interpretivist paradigm.

The first case study (company A) is still ongoing and was conducted in several phases since 2006 [6, 7]. The different phases consisted of a) 15 explorative interviews with several German SMEs, b) on-site observations, in two German and one Russian company as well as c) an action research approach in one of the German SMEs.

The second case study (company B) was conducted in a small Irish-Romanian company in 2007 [8]. We used ethnographically-informed methods, undertaking visits in both sites, interviewing the managers and two

developers and collecting a number of documents. A new round of interviews were conducted in January 2009 for an update on the company’s situation and practices.

4. Case Studies

Company A is a small German software enterprise engaged in the field of statistics and documentation. Customers are mainly German archives and museums. The company was established in 1980 in Bonn and has approximately 20 employees. In the mid-1990s the company found it increasingly difficult to hire German developers. Hence, based on a positive experience with a very talented Russian developer who did an internship at the company, the owner of company A decided to found a branch in Siberia. Since then, an average of four to eight employees are working for company A in Tomsk, including the former intern. The first project aimed at reengineering an existing product, which had to be rebuilt in C++. Hence, offshoring enabled the company to provide a modern architecture to their existing products, and the cooperation was expanded to several small size projects. Recently, the company has also attempted to enter the Russian market.

Company B was established in January 2006 in Dublin, Ireland. The two managers had worked together for 4 years in a company providing software applications for Telecoms and media companies. One of them had been a project manager and the other (originally from Romania) had been working on his team as senior developer. In January 2006, they decided to set up their own company and hired 4 developers in Dublin. Then, the Romanian manager identified a small company with 5 employees in Bucharest, Romania which they acquired. In December 2007, there were 19 people working in the company’s offices in Romania, and another project manager (beside the Irish manager) in Dublin, with the Romanian manager traveling between Dublin and Bucharest more frequently. Being an Irish-based company made them attractive on the international arena, as Irish companies have the reputation of being stable and reliable, while their development division in Romania was a signal for customers that the company can offer quality work at attractive prices.

5. Findings

In this section, we will briefly present knowledge practices we observed in the two companies, where human actors play the role of knowledge brokers and cultural mediators.

Permanent communication – Sharing a virtual office.

In company B, the permanent communication between the Romanian and the Irish company managers plays a paramount role in running the company; they have been working together for the last 6 years and know each other well, so working over distance is not a problem. Every morning they meet on Skype for an update, and maintain an open communication channel throughout the work day. They have a permanent shared general view on: the situation of each project, the tasks each of the developers is working on and the level of satisfaction of their customers—while documentation is reduced to a minimum. Information on customers is corroborated, used to attain the set business goals and, whenever necessary, shared with the developers. The Romanian manager maintains a comprehensive perspective on the developers' technical skills, on the progress of their tasks and the challenges encountered, but also on their personal situation and social dynamics within the group; she also gives regular updates to her Irish counterpart.

Mutual visits – How is it on the other side? Company B invites Romanian developers for brief visits (2-5 days) to Ireland and other customer sites; these trips are perceived as opportunities to meet counterparts on the customer side and are related to project milestones. As one of them told us, “initially these people did not feel real to me. I was talking to them every day, I was aware of them being delayed in the traffic in some mornings or missing from work because of a flu, but it was difficult to imagine them. After a very short visit to Dublin, everything changed. Now I could put a face to a name, and after my return I’ve done my best to share my impressions with my Romanian colleagues.”

Company A has a similar approach, inviting developers over and giving selected Russian developers the opportunity to work for the company in Germany for longer periods (3-24 months). This practice aims at facilitating the knowledge exchange between the teams. As a Russian visiting developer explained, being on site and aware of customers' problems gives him the opportunity to suggest improvements and innovations based on his deep technical knowledge. Furthermore, the Russian developers in Germany are asked to mediate between the teams, serving as intercultural bridges. As the Russian developer explained: “I am frequently getting requests from (the German manager) or from (the Russian team manager) to improve communication. (...) I am running around, asking people what is the status of different things, what are the difficulties in communication, what are the points where people feel dissatisfied with the other party’s work. (...) It saves a lot of time, effort and emo-

tions that I understand the language, that I can hear their complaints (laughter).”

Weekly Meetings – What is going on in the company? Company A organizes a weekly meeting in order to keep an overview about what is going on in the company, to discuss current developments and problems and to give developers the opportunity to talk about discoveries which may be useful for the team as a whole, like for example a new tool or an innovative technology. The offshore team at Tomsk is organizing a similar event. Both teams write minutes which are meant to provide a short overview for the other team. However, as both developers and project managers reported, the information shared during the meetings and in the minutes exchanged between sites is not enough for keeping up to date—“many explain simply too little.” Hence, instead of relying on this information, the developers and project managers reported they stayed aware of what was going on in the company by talking to people. “Often I simply walk around (...) and ask (...) ‘what are you doing right now?’” In this regard, the minutes of the remote team’s meetings are used as hints for further inquiries to them: “(...) sometimes I can find something new, unknown, or I realize, ‘ah, they are working on the same problem I worked on some time ago!’” In this regard, the short references to what is going on in Tomsk serve as props for direct requests and communication, but not necessarily as a medium for exchanging knowledge directly.

6. Discussion

Despite the global context of their work, the companies in our sample tended to rely on agile development practices [5] in regard to managing knowledge. The related practices we observed in the field were rather unsystematic, and strongly reliant on certain developers or project managers who influence the development activities and maintain an overview about what is going on—partly, because of their formal roles, and partly because of their personal or social inclinations. As Milewski et al.’s study [1] suggested, these roles are neither formal, nor planned, but emerge from tangible necessities of dealing with everyday’s work.

In our case, because of the small size of these teams, knowledge brokers are able talk to each and every person. Hence, they can not only bridge the structural holes between the teams, but—in a way—close them, like in company B. However, because of the very personal character of these relationships, these people are nearly impossible to replace. Their uninterrupted commitment is vital for the cooperation. In a way, the whole offshoring relationship is based on ex-

tending the trust granted initially to one person to include a group of people offshore this person guarantees for. However, bridges can be double-edged swords, as in some circumstances they can become bottlenecks.

The formal procedures of knowledge exchange (for example the weekly meetings minutes) turned out to be less effective for knowledge exchange and were used rather as anchors by the knowledge brokers. Knowledge is not stored in them—instead, they are used to identify gaps and stay aware. Hence, it became apparent that bridges are vital elements in the articulation mechanisms in use. Meetings and artifacts make good coordination mechanisms only in the presence of people who can connect and enact them.

As in SMEs efficient knowledge exchange is mainly accomplished by walking around the office and talking to people, bridging the distance poses several obstacles. In this regard, our observations showed the important role played by: a) permanently open communication channels, b) personal visits for socializing and exchanging knowledge, c) long term collaboration, giving people the chance to build shared understandings over time.

7. Conclusion

Our case studies illustrated some of the knowledge exchange practices of small software companies with distributed sites. In particular, we found a high relevance of personal connections between the teams, and of informal ways of dealing with knowledge.

While Milewski et al focused on social networks—our focus is on knowledge work practices. We are interested in how people (not organizations, not tools) make things work. In this regard, we pursued a classic CSCW concept: the work to make work happen, the “invisible” *articulation work* [6].

Apart of extending the findings of Milewski et al., our observations also reveal several challenges for the management of small software teams working with agile methods, which we believe are not well understood yet—and should be addressed by future studies:

- Is it possible to train someone to become a knowledge broker if the offshoring relationship wasn’t initiated through this person (as in our two cases)? If so, how?

- Is there anything to do regarding the tool support for augmenting this role? Can artifacts be designed to stimulate informal knowledge exchange practices and balance formal and informal communication as well as codified and implied knowledge?

- How can companies avoid overloading these people and turning them into bottlenecks? For example,

can bridges be also motivated to replicate their skills by training other people as bridges?

- Can the role of knowledge brokers be expanded to include liaising with the customers, too?

8. Acknowledgments

The studies reported in this paper were partially supported by: The German Research Foundation (DFG) within the research project ARTOS, and Lero (the Irish Software Engineering Research Centre) within the socGSD cluster project funded by the Science Foundation of Ireland (SFI).

8. References

- [1] A.E. Milewski, M. Tremaine, R. Egan, S. Zhang, F. Kobler, and P. O’Sullivan, “Guidelines for Effective Bridging in Global Software Engineering,” Proceedings of the 2008 International Conference on Global Software Engineering, Los Alamitos, CA, pp. 23-32.
- [2] T.H. Davenport and L. Prusak, Working knowledge: how organizations manage what they know, Boston, Mass: Harvard Business School Press, 1998.
- [3] F.O. Bjørnson and T. Dingsøyr, “Knowledge management in software engineering: A systematic review of studied concepts, findings and research methods used,” Information and Software Technology, vol. 50, Oct. 2008, pp. 1055-1068.
- [4] H. Timonen and M. Jalonens, “A Critical Review of Knowledge Management Literature: Introducing a Practice-based Approach on Knowledge Sharing,” Proceedings of the 2008 European Conference on Knowledge Management, Southampton, pp. 1055-1068.
- [5] Manifesto for Agile Software Development, <http://agilemanifesto.org/>, [last accessed: 20.01.2009].
- [6] A. Boden, B. Nett, and V. Wulf, “Articulation work in small-scale offshore software development projects,” Proceedings of the 2008 international workshop on Cooperative and human aspects of software engineering, Leipzig, Germany, pp. 21-24.
- [7] A. Boden, B. Nett, and V. Wulf, “Coordination Practices in Distributed Software Development of Small Enterprises,” Proceedings of the 2007 International Conference on Global Software Engineering, Munich, pp. 235-246.
- [8] I. Richardson, G. Avram, S. Deshpande, and V. Casey, “Having a Foot on Each Shore - Bridging Global Software Development in the Case of SMEs.” Proceedings of the 2008 IEEE international Conference on Global Software Engineering , Washington, DC, pp. 13-22.