Team Composition in Distributed Software Development

Roderick van Cann, Slinger Jansen, Sjaak Brinkkemper

Cooresponding author:
Slinger Jansen

Universiteit Utrecht
PO Box 80.089
3508 TB Utrecht
The Netherlands

Telephone: +31 (30) 253 1807
Email: S.Jansen@cs.uu.nl

Acknowledgements
We would like to thank the following companies for their support and for providing useful research data: CoMakeIT, Deloitte, InnoVites, Itude, Logica, and Microsoft. Finally we would also like to thank all companies and other parties involved in the sponsoring of the research trip to India.

ABSTRACT
Team composition greatly influences project and organizational performance. With literature lacking a simple team composition method that can be used in an outsourcing situation, we propose a method to assist project managers by answering our research question: 'What is an optimal outsourcing team composition method in a distributed software development environment?' This method provides project managers with a guideline containing easy steps to compose their optimal outsourcing team. We evaluated our method with data collected from six companies in the Netherlands and India.

Keywords: team composition method; distributed software development; outsourcing.

INTRODUCTION

The composition of a team greatly influences the project and organizational performance (Baykasoglu, Dereli & Das, 2007; Wi, Oh, Mun & Jung, 2009; Chi & Cheng, 2009), which makes it an essential aspect of outsourcing for companies that depend on the productivity and success of their teams.

While there are a number of authors (Ahn, DeAngelis & Barber, 2007; Wi et al., 2009; Baykasoglu et al., 2007; Tseng, Huang, Chu & Gung, 2004; Korvin, Shipley & Kleyle, 2002; Zakarian & Kusiak, 1999; Venkatamuni & Rao, 2010) that have developed different kinds of models, using for example fuzzy logic, to evaluate the most appropriate people for a certain team, there is hardly any literature that describes a simple method or model when it comes to team composition.

The research question we will answer in this paper is: *What is an optimal outsourcing team composition method in a distributed software development environment?*

This research question is answered by introducing a team composition method. This method will help project managers with the composition of their software development team in an outsourcing environment.

We define an outsourcing team as all people directly involved in the outsourcing activities. Lasser and Heiss (2005) state that '*Geographic dispersion makes teambuilding (e.g.*

establishing trust) more difficult and may induce language and cultural barriers that hamper effective communication.'(p.718) As stated in our research question, we will focus specifically on the outsourcing of software development. An essential aspect of any team is that they actually work as a team and also view themselves as a team (Wi et al., 2009; Roland, Yttredal & Moldskred, 2008). If a team is composed properly, they will be able to make faster and safer decisions (Roland et al., 2008), which would lead to better results for the company.

Having a well-composed team is not only critical to the company that develops the software, but also to the company that outsources their software development. Since software companies usually have a high employee turnover, often up to 15 or even 20 percent, customers want to know what they are getting into before spending a lot of time and resources to transfer their knowledge to the employees of the supplier (Padmanabhan, 1996). According to Padmanabhan (1996), knowledge transfer is mainly done during the initial phases of a project, when for instance the requirements specification and the high-level design are developed. While it is impossible to eliminate employee turnover, a well composed team can reduce the impact employee turnover can have on knowledge retention in a team. The knowledge that there is a well-composed team in place could therefore be a reassurance to the company issuing the outsourcing.

While some of the existing team composition methods focus almost solely on the personal characteristics of the employees (i.e. Korvin at al., 2002; Baykasoglu et al., 2007; Ahn et al., 2007; Wi et al., 2009), others do not consider employee characteristics at all in their methods (i.e. Tseng et al., 2004; Venkatamuni & Rao, 2010; Zakarian & Kusiak, 1999). According to Roland et al. (2008), the functional team roles are often the basis for the composition of a team. Team members are, according to Roland et al. (2008), usually '*selected based on professional qualifications and back ground*' (p.3). During this research, we will focus on the functional roles that should be present in these teams, while not neglecting the importance of personal characteristics. The project characteristics, often disregarded during the evaluation of potential candidates according to Wi et al. (2009), will also be taken into account. Considering we focus on the composition of an outsourcing team, we will take into account the number of people required for the project, and where people should be located, either on the onshore or offshore location.

This chapter is organized as follows: in the next section, we will discuss the related work, including the description of a number of case studies of team compositions that can be found in literature. Next, we will describe the different team composition methodologies that can be found in literature, and apply a method engineering approach based on fragment comparison on five team composition methods, followed by our proposed method in the next section. Then we will evaluate our proposed method using companies involved in software development outsourcing activities. Following the evaluation, we will provide a discussion section with regard to our research findings and our proposed method. Finally, we will draw some conclusions based on our research.

RELATED WORK

A team consists of a group of people that are linked together and have a common purpose (Roland et al., 2008). An outsourcing team, therefore, is the team that is responsible for the outsourcing activities, and can consist of people working onshore as well as offshore. In addition to the common purpose indicated by Roland et al. (2008), in a globally distributed team (GDT), team members work at one or multiple locations, are distributed over multiple countries and rely more on technology for their communication than on face-to-face communication (Mattarelli & Gupta, 2009).

Tsai, Moskowitz and Lee (2003) claim that *'the failure of software development projects is often a result of inadequate human resource project planning'* (p.167). One of the issues with regard to human resource planning for software development projects they identified is the scarcity of resources. They indicate that qualified people, who have both knowledge and experience related to the current project and the required technical skills, are usually scarce and therefore can be sought-after to work on multiple projects. Tsai et al. (2003) therefore suggest that instead of scheduling activities for a project, resources should be scheduled. Besides project cost and duration, having knowledgeable personnel is an essential element for project success that should be taken into account when selecting human resources for a project (Tsai et al., 2003).

Team building is considered an important responsibility of the outsourcing supplier by Padmanabhan (1996). This often makes the outsourcing customers worried about the knowledge retention and performance of the established team (Padmanabhan, 1996). A crucial property that the supplier typically implements in their team is *'a degree of redundancy of skill-sets to allow for staff turnover'* (Padmanabhan, 1996, p.10). For teamwork to lead to project success, team members should have knowledge of what their fellow team members are working on (Leinonen, Järvelä & Häkkinen, 2005; Cohen & Thias, 2009).

Even though they argue that a collocated team is better than a distributed team, Cohen and Thias (2009) give the following key aspects of project success:

- Having the team actually see itself as a team, instead of a group of individuals;
- Having the right skills in the team;
- Having a good partnership between the customer and the developers;
- Have short development iterations; and
- Have the focus on quality.

Cohen and Thias (2009) claim one of the biggest problems with offshore developers, is that they do not have a sense of ownership, or share the same goal as the onshore team. This claim is partly supported by Bulka, Gaston and desJardins (2007), when they mention that local performance is not equal to global performance, meaning that when a part of a team is trying to optimize their performance, this does not automatically mean that the entire team performance is optimized. Therefore, making sure to establish a well-composed offshore, or distributed, team is critical. Such a team should be able to overcome these issues mentioned by Cohen and Thias (2009).

Four different types of teams that can exist in a company are identified by Cohen and Bailey (1997). Of these four possible types, two are relevant for the outsourcing of software development. These two types are the work teams and project teams, for the development activities that are outsourced continuous or on a project basis, respectively. Mattarelli and Gupta (2009) have also identified a number of different types of teams, with differences in having the options of including people from the client company, and having someone to mediate between the onsite and offshore teams.

The opinion of researchers with regard to the influence the team size has on project success varies. While some authors claim to have found evidence that team size is positively related to performance (Campion, Medsker & Higgs, 1993; Magjuka & Baldwin, 1991), other authors have found no direct relation between the team size and the performance of a team (Smith et al., 1994). Scott and Cross (1995) and Cohen and Bailey (1997) are of opinion that team size does play a role and should definitely be taken into account during the team composition phase. Besides the team size, the tasks that the team will have to carry out should also be included in a team composition method (Chi & Cheng, 2009). According to Mattarelli and Gupta (2009), all GDTs consist of a larger offshore team in comparison with the onsite team. McGrew, Bilotta and Deeney (1999) found a significant correlation in their research between communication within the team and the team size. They studied different teams by performing Capability Maturity Model level two audits. In their study, McGrew et al. (1999)

found a correlation between the communication and the audit score, indicating that communication can have a great impact on the team.

Outsourcing intensity

Competitive outsourcing against low cost can be seen as an optimization problem, where companies need to find an optimal balance between the availability of onsite experts and labor costs, since this usually is the biggest part of the total project costs when talking about software development (Lasser & Heiss, 2005). This can mean that companies need to be involved in distributed development to be able to meet customer requirements. In these cases they cannot simply *'assigning tasks to the best experts available, independent of where they are located'* (p.719), due to coordination costs that increase significantly when virtual teams are distributed among multiple locations (Lasser & Heiss, 2005).

Outsourcing can be done in different forms, which can, according to Lasser and Heiss (2005), be classified in fifteen stages, increasing the level of outsourcing with each subsequent stage. These stages describe the intensity of outsourcing that is done by the company, and start with a one person project in stage one, ending with outsourcing everything, including business responsibilities, in stage 15. The total costs are the lowest in stage 15, but the disadvantage of this stage is that it provides less flexibility with regard to the team composition compared to the other stages (Lasser & Heiss, 2005). They mentions that the higher the level of outsourcing, the higher the *'maturity of knowledge networking, including internal on-demand support, and consulting serviced offered by experts from other sites'* (p.719) is required to be, to successfully perform the activities. They calculated that companies usually drop below the "offshoring cost barrier" at stage nine. This means that from that stage on, the total costs are less compared to performing all the activities in-house. This stage includes outsourcing non-critical work, development and some of the key functions. When taking into account the collaboration maturity, the off-shoring cost barrier is located at stage 10, which also includes sub-project management.

A relevant point Lasser and Heiss (2005) make in their paper is that *'there is no such a thing as a "universal optimal form of collaboration"'* (p.726). They mention that the balance between cost and effectiveness should be sought for each project separately. This should include *'the trade-off between flexibility in team composition and crosssite communication effort'* (p.726).

While these stages of Lasser & Heiss (2005) give a good overview of the possibilities of different levels of outsourcing, we believe fifteen stages is too much in order to maintain a simple method that can be easily applied. The level of outsourcing in itself is however critical when talking about a method for team composition in a distributed software development environment.

Selection options

Four different options for team formation are, according to Karduck (1994), determined by the option of selecting individuals or selecting teams, the option of deliberate team formation, where people are selected to accomplish a certain objective, or spontaneous team formation, where people come to participate on a certain project. Karduck (1994) suggests creating information networks with regard to the topic of interest, the past projects and the characteristics of the individuals that are candidate for team membership. The selection will then be based on the required expertise and the existing relationships between the individuals.

Scott and Cross (1995) mention five different methods for selecting student teams in classroom settings. These methods are:

- Choose teams randomly;
- Let students choose team members;

- Choose team members to balance academic achievement and performance profiles;
- Choose team members based on results from psychological profiles; and
- Choose team members based on a mixture of these methods.

Assigning students randomly to teams can lead to newness and freshness in newly formed teams. On the other hand, it risks placing the best, or the worst, students in a single team, thereby limiting their abilities. Team abilities are also in risk of being limited when students are allowed to form their own teams, since they *'tend to work well with people they already know'* (Scott & Cross, 1995, p.296). Since Scott and Cross (1995) discuss teams for short academic projects, they indicated that for small teams, consisting of two or three students, letting them choose their own team members delivers the most optimal result. For larger teams, however, they believe the balancing of both academic achievement and performance profiles, and psychological profiles, is the best approach. They believe this has a positive effect on the team performance, although they indicate that these approaches do not take into account whether the team members can work well together.

Deibel (2005), who also researched student teams, found that interaction between team members is a key aspect of group learning, and collaboration has a positive impact on the socialization and information processing abilities of the students. This exposes them to the different ideas students can have and encourages their cognitive growth (Deibel, 2005).

To summarize what we believe to be the most influential criteria for team composition as describe above from literature, we provide a few of bullet points that should be considered during the team composition activities.

- Assume intense coordination between individuals in a dynamic working environment: since we are focusing on a distributed software development environment.
- The level of outsourcing: how much, and what, will be done onshore in comparison with offshore. This process should be kept simple.
- The size of a team: with small groups, the collaboration between individuals is vital, while with larger teams, the complementation of knowledge and skill plays a major role.
- People should be selected individually, not per team, to ensure the most flexibility in team composition.

Case studies in literature

There are a few case studies to be found in literature that describe the team composition in certain outsourcing or software development situations. We will discuss four of these case studies here to be able to derive some general information from these case studies. All of these case studies were considered to be successful.

A perhaps exceptional example is the case study described by Boland and Fitzgerald (2004), where an existing development team with less than 20 developers is being distributed between the United Stated and Ireland. This project was considered a success due to good communication between the two sides and a single software manager (Boland & Fitzgerald, 2004).

In a highly distributed case study done by Leinonen et al. (2005), where 19 employees from a single organization are distributed over six countries in three continents. The biggest drawback of working in a virtual team was identified as being the lack of awareness of what other team members are working on. They state that recent studies showed that the outcome of the collaboration is dependent on *'how the participants manage the content of the problem and the social relations between individuals'* (Leinonen et al., 2005, p.316).

Kussmaul, Jack and Sponsler (2004) describe a case study where two to three team members onshore execute the analysis, high-level design, and coordination tasks, but also participate in some low level implementation and testing. The offshore part of the team, that consists of five to ten members, works on the low-level design, implementation and testing, but sometimes also participate in analysis and design. In this case study, the onshore team is responsible for resolving *'open issues from the daily meeting, so that the offshore team can continue work the next day'* (Kussmaul et al., 2004, p.150).

A large case study from Sutherland, Viktorov, Blount and Puntikov (2007) involves over 50 developers working in scrum teams. These scrum teams are split between the U.S. and Russia, with the scrum of scrums and the product owner located onshore. These scrum teams where almost equally divided between the U.S. and Russia, with per team about three to five people in the U.S., and four or more in Russia. Sutherland et al. (2007) indicated that the Scrum master, basically the project manager, was located in the U.S., but a technical lead team member was situated in Russia to lead the local developers. This case study indicates that large teams can also be successfully distributed.

While it is difficult to draw some preliminary conclusions from these case studies, we can observe that about half of the development teams in these cases seem to be distributed more or less equally between the onshore and offshore locations, and the other half having a larger offshore team in comparison with the onshore team. We have not found a clearly described case study where the onshore team is larger than the offshore team.

TEAM COMPOSITION METHOD DESCRIPTIONS

Current literature describes a few methods that are developed for team composition. We have modeled these methods to provide a high level overview of the activities in these methods. This provides us with the opportunity to indicate the advantages and deficits of existing team composition methods in literature.

The first method, which can be found in Figure 1, is described by Ahn et al (2007). This method only focuses on three dimensions that they find to be of influence for potential team members. These dimensions are:

- Reliability, the tendency of a potential team member to fulfill the commitment;
- Quality, the quality of service that the team member provides; and
- Availability, the tendency to work as a team member.

For each of the potential team members, these three dimensions are rated. Weights are then applied on the three dimensions, depending on the importance of these dimensions for the project in question. The fit of each of the potential team members is then calculated as the weighted sum of these three dimensions. The team members with the highest scores are then selected to work on the team.

This method focuses mainly on projects where a reward is given when it is finished on time and projects where the project leader is penalized when the time limit is exceeded. The level of the reward depends on the quality of the work. Probably because of this focus, this method only discusses these three dimensions. We believe that this method lacks a number of attributes, such as the expertise or experience of potential team members, the functions of the team members, and the project characteristics.

The second method we discuss is described by Wi et al. (2009) and focuses mainly on personal and network knowledge of potential team members. This method, that can be found in Figure 2, has a strong focus on taking the project characteristics in consideration for the evaluation of potential team members. Wi et al. (2009) suggest using project keywords to search on publications of potential team members. Based on these publications, a personal knowledge score is computed based on the amount of keywords that are present in the publication, to indicate the relevance of the topic, and when the publication was made, to indicate whether the authors' knowledge is recent and up to date. Besides the personal knowledge score, they suggest to build a social network by looking at the publications of the co-authors of relevant publications. By using the same attributes as with the personal knowledge score, extended with the number of coauthors, a social network score is also computed. These two scores together make up a personal score for that particular team member for the project. The people with the highest scores are then allocated to the team, and the person with the highest score among them is made project manager.

Our biggest arguments against this method is that, especially in the information technology (IT) field, the project manager should not necessarily have the most knowledge on the subject, but should mainly have good communication and leadership skills. Another argument is that this method cannot be properly implemented in companies, since most employees never publish anything and therefore cannot be selected for a team when using this method. While there are companies that publish documents internally, not all knowledge is always documented in co-authored documents.

Three methods that can be found in literature, Tseng et al. (2004), Venkatamuni & Rao (2010), and Zakarian & Kusiak (1999), are quite similar. The methods are modeled in Figure 3, 4 and 5 respectively. The three methods start with the same three activities, being the identification of the customer requirements and the identification of the engineering characteristics, which are mapped against the corresponding customer requirements in a matrix.

After these initial three activities, Tseng et al. (2004), Figure 3, add weights to both the customer requirements and he engineering characteristics in the matrix, after which these weights are used to order the matrix on importance. During this ordering, the items in the matrix are also grouped. The potential teams that are available are then selected, and their group density indexes (GDI’s) calculated, based on their characteristics. These GDI’s are then mapped to, a part of, the matrix. The team whose GDI matches the section of the matrix best is selected. This mapping to the matrix and selecting the best fitting potential team is continued until the entire matrix is mapped to a potential team.

The first three activities described above – the identification of the customer requirements and engineering characteristics, which are mapped to one-another in a matrix –, are the same for the method of Venkatamuni and Rao (2010) in Figure 4. After these three activities, they identify the team functions that should be present, based on the engineering characteristics. They create a separate matrix to map the team functions to the engineering characteristics. Weights are then added to the customer requirements, which influences the weights added to the engineering characteristics, which in turn influence the weights added to the team functions. Both matrixes are then ordered based on the weights that were added. Venkatamuni and Rao (2010) then suggest the selection of team members, based on the ordered matrixes, which show which team functions are the most critical for the completion of the project.

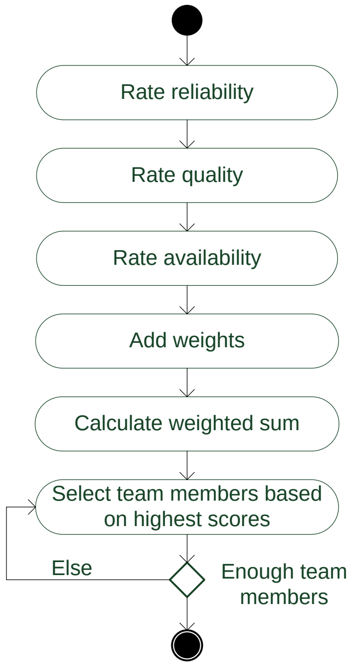

Figure 1: Ahn et al. (2007) method

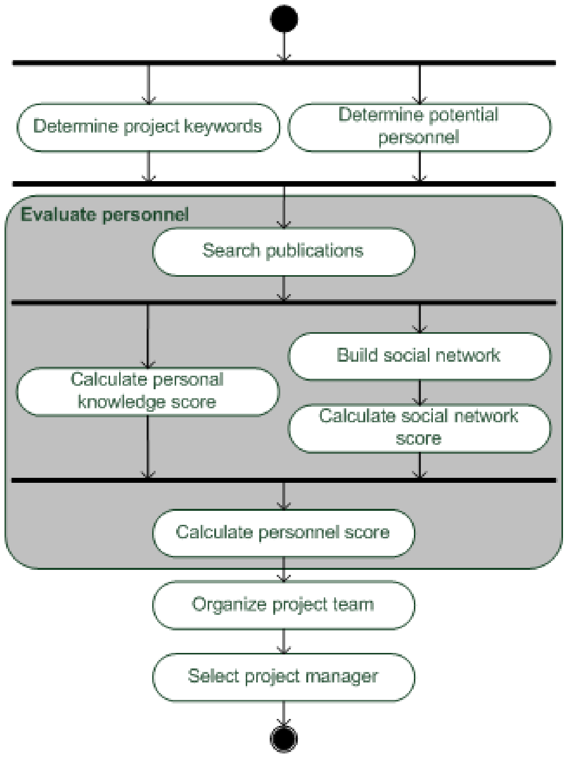

Figure 2: Wi et al. (2009) method

As can be seen in Figure 5, the process described by Zakarian and Kusiak (1999) is quite similar to that of Venkatamuni and Rao (2010), which is modeled in Figure 4. The first seven activities are exactly the same, the seventh being the ordering of the matrixes based on the weights. From there on, Zakarian and Kusiak take a slight detour of the method of Venkatamuni and Rao. Zakarian and Kusiak (1999) also group the matrixes in the seventh step, and then suggest creating a hierarchical model to visualize the relationship between the project, customer requirements, engineering characteristics, and team functions. They use a mathematical programming model to calculate the optimal team member composition, which includes a priority level for each of the team member functions. This programming model takes into account the number of projects a team member can be involved with at any giving time, the priority of the engineering characteristics for this project, and the number of teams that should be selected. The outcome of this mathematical programming model shows the team member functions that should be selected for the project.

We believe the main flaw in the methods of Tseng et al. (2004) and Zakarian & Kusiak (1999), Figures 3 and 5 respectively, is the complexity of the calculations they use. Tseng et al. (2004) also allocate groups to a project instead of individuals, which severely limits the team composition options. Tseng et al. (2004) indicated that the relationship between the customer requirements and the system characteristics are difficult to identify in real world cases. The selection of team members based solely on group fit is not optimal, since team members differ in their capabilities (Tseng et al., 2004).

Venkatamuni and Rao (2010) select their team members based only on the ordered matrixes, which provides them with the most critical team functions, but lacks any information on the individuals that should fill those functions. The method of Zakarian and Kusiak (1999) does not even select actual team members, but only select team functions based on the outcome of their mathematical programming model.

Finally, none of these three methods take into account any information with regard to the knowledge or past experience of potential team members. We believe this to be crucial information when striving for project success and a well-functioning team.

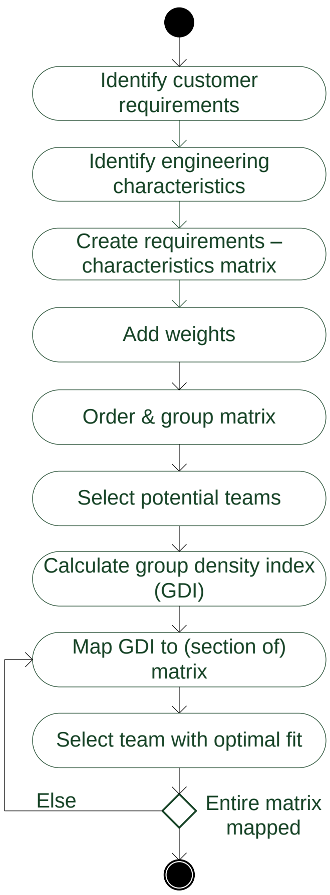

Figure 3: Tseng et al. (2004) method

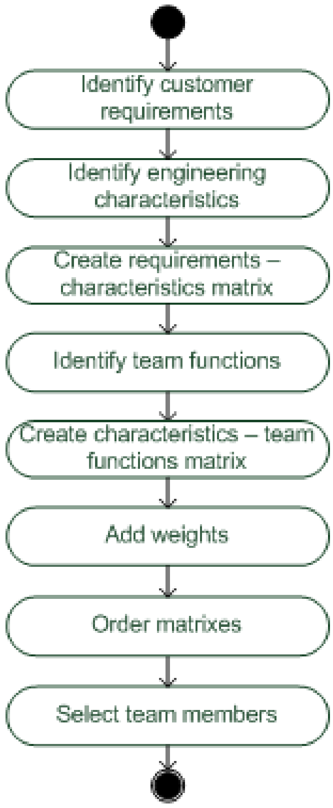

Figure 4: Venkatamuni & Rao (2010) method

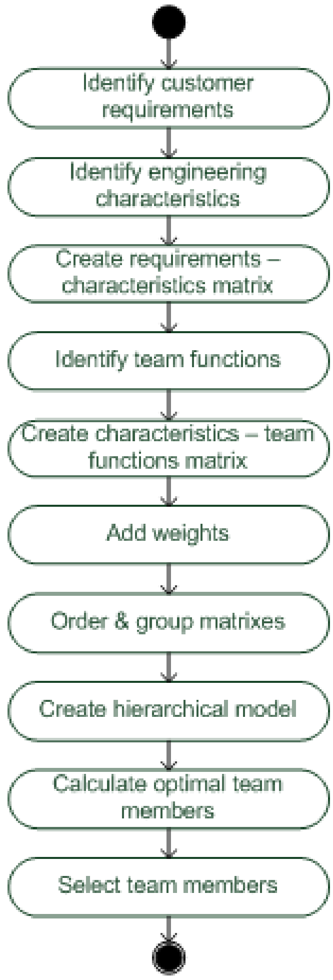

Figure 5: Zakarian & Kusiak (1999) method

METHOD COMPARISON

All five of the methods described above lack certain aspects that are crucial in an outsourcing situation, which makes them unlikely candidates to use when composing an outsourcing team. One of these aspects is for instance the location where the team member with a certain function should be located. This depends on a number of properties, like the project, the level of outsourcing and the preferences of the companies involved. We have used these five team composition methods, modeled in figures 1 to 5, to perform a method comparison. Table 1 below shows the result of this comparison. In this method comparison matrix, we use the same indicators as Hong, Goor and Brinkkemper (1993) indicated in their proposed formal approach. These indicators are: '=', the method activity is similar to the super method activity; '<', the super method activity does less than the method activity; '>', the super method activity does more than the method activity; and ‘><’, parts of the method activity overlap with super method activity.

We did not indicate any differences in the naming of the activities between the different methods; we judge the relationships solely on the actual activity.

	Ahn et al. (2007)	Wi et al. (2009)	Tseng et al. (2004)	Venkatamuni & Rao (2010)	Zakarian & Kusiak (1999)
Identify project properties					
Determine project keywords		=			
Determine potential personnel		=	=*		
Identify customer requirements			=	=	=
Identify engineering characteristics			=	=	=
Identify team functions				=	=
Personal properties					
Search publications		=			
Build social network		=			
Rate reliability	=				
Rate quality	=				
Rate availability	=				
Data manipulation					
Create matrix			=	=	=
Add weights	=		=	=	=
Calculate weighted sum	=				
Calculate social network score		=			
Calculate personal knowledge score		=			
Calculate personnel score		=			
Order & group matrix			=	=**	=
Create hierarchical model					=
Team member selection					
Calculate optimal team members					=
Calculate group density index (GDI)			=		
Map GDI to (section of) matrix			=		
Select team members	=	=	=*	=	=

* = selects teams instead of individuals

** = only orders the matrix

Table 1: Method comparison matrix

The method comparison matrix in Figure 1 shows that the super method contains 22 different activities, which can be divided in four different groups: Identify project properties (five activities), Personal properties (five activities), Data manipulation (seven activities), and Team member selection (four activities), leaving one visualization activity without a group. The number of activities does not differ much between the five methods: Ahn et al. (2007) have six activities, Wi et al. (2009) have eight activities, Tseng et al. (2004) nine, Venkatamuni & Rao (2010) seven, and Zakarian & Kusiak have nine activities.

From the four different groups of the super method, the data manipulation seems to be more present than the personal properties. Not only does this group contain more activities, it is the only group, besides Team member selection, where all five methods perform activities. The adding of weights to the data appears to be one of the most used activities, with four out of five methods using this activity in their method.

OUTSOURCING TEAM COMPOSITION METHOD

Since the methods described in the previous section do not fit the needs of a simple and comprehensive method that can be used for team composition in an outsourcing situation, we propose the following method, as depicted in Figure 6. This method is assembled from method fragments originating from the above mentioned methods from literature, together with some data from industry. To model the method, we used the method engineering approach as described by Weerd and Brinkkemper (2009). We aimed to use the best method fragments from literature and complement these with method fragments that were missing for the outsourcing situation. In trying to keep this method simple, we aim to be able to make this method useful for project managers and upper management. They should be able to use this method without having a thorough understanding of, and having to rely on, complicated mathematical formulas or computer programs to be able to properly compose their outsourcing team.

Our proposed outsourcing team composition method, Figure 6, assumes that a number of aspects of the outsourcing activities are already in place. We assume, for instance, that the outsourcing partner has already been selected and contracts have been handled. Knowledge about the project should be available, just as the company preferences on, for example, the level of outsourcing that is preferred in the current situation. This method should be used for each project that will be outsourced separately, since it relies heavily on the project characteristics.

The method does not contain an activity to select a level of outsourcing, what functions a company should be outsourcing, but solely focuses on getting the right amount of people to work on the project activities on both sides of the outsourcing relationship. The level of outsourcing, and thereby also the decision as to the location of team members, which is done in a later stage of the method, should be based mainly on the company preferences and project characteristics.

This method starts with a number of activities to identify the properties that are related to the project. The first two activities are the identification of the project characteristics and the available team members for this project, which will both be documented. The project characteristics will contain properties such as the following: the time limit, the size of the project, the impact the project is expected to have on the business and the priority level of the project. Of the available team members, it is imperative to know what kind or knowledge they possess, what their previous experiences are with regard to the topic and the techniques that will be used and the

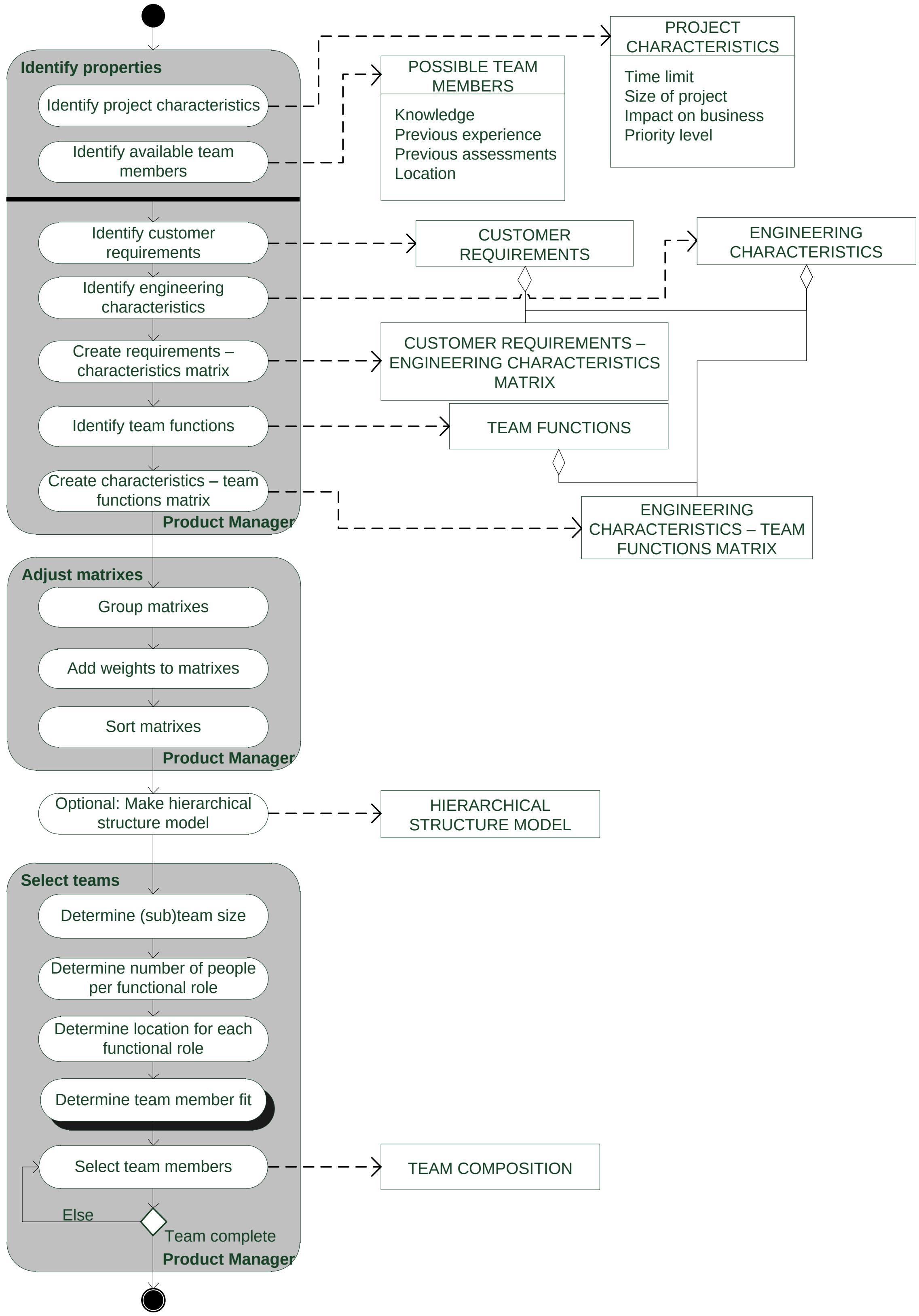

Figure 6: Outsourcing team composition method

previous assessments of their work and attitude. At which location these people are located is also of importance, especially when using the method to compose a distributed team.

Next, the customer requirements are thoroughly identified and documented. Based on these customer requirements, the engineering characteristics are identified during the next activity. The engineering characteristics describe what knowledge is necessary to fulfill the customer requirements. The customer requirements will be mapped to the engineering characteristics in a matrix. The engineering characteristics are the subjects of which knowledge is needed for the realization of the customer requirements. Based on these engineering characteristics, the team functions are identified. This is done by looking at what kind of people at the company possess these certain knowledge elements. The engineering characteristics are then also placed in a separate matrix, where they will be mapped to the team functions.

With the result up to this point of two matrices, the next phase of activities can be started, where some adjustments will be made to these matrixes. The first activity groups the data in the matrixes. This grouping activity starts by grouping the matrix that maps the engineering characteristics to the team functions. The team functions are grouped together, and the engineering characteristics are grouped to match those groups of team functions. This grouping of engineering characteristics is then copied to the first matrix. The customer requirements in this matrix are then grouped to match the engineering characteristics. Basically, this step involves the grouping of the team functions, and then simply keeping the connections to the other elements, the engineering characteristics and the customer requirements, intact.

The next activity adds weights to the customer requirements in the first matrix, from which the weights of the engineering characteristics can be derived. After the weights of the engineering characteristics are copied to the second matrix, the weights can also be added to the team functions.

The final step in the adjustment of the matrixes involved the sorting of the matrixes on the weights added to the different axis, while still keeping the groupings intact.

The following activity, which is optional, involves the creation of a hierarchical structure model. This activity is optional since it only provides a visualization of the activities up to this point and does not provide any additional information with regard to the team composition. On the top level of the structure model is the project goal, followed by the customer requirements on the second level, and the engineering characteristics on the third level. The engineering characteristics on the third level are in turn connected to the team functions on the fourth and final level, providing a hierarchical model that visualizes the structure of the matrixes.

The final section of activities involves the actual selection of the team members. This section starts with determining the team size. The preferred team size is dependent mainly on the company preferences. This can mean that at a single location, the option is chosen to create multiple teams, which in term can also influence the number of people that are needed. This is done in the next activity, the determining of the number of people that are needed for each of the different team functions. This is done based on the weights on the team functions and customer requirements, but also on the team size and the project characteristics, since a shorter time frame would suggest more people might be needed to complete the project on time. For each of these team functions, it will then be determined at which location these should be placed. This could all be at one location, but if a team function will be filled by multiple people, these could also be distributed among multiple locations. Aspects that influence the choice of the location are the project characteristics, but mainly the company´s preferences and the level of outsourcing that is envisioned for this project.

The next step is a complicated activity, which we do not describe in detail. It involves determining the team member fit on the project. In this step, the possible team members that are identified in the beginning of the method should be evaluated on their knowledge, experience, previous assessments, location, and personal characteristics. Which of these aspects of the team members are the most critical will be different for each project, but will also vary between different project managers that have to make the decisions regarding the selection of personnel. It is imperative not to take this step lightly, since the fit of a team member within the team can have a great impact on the team performance.

The final activity involves the actual selection of the team members. This is based on all the available knowledge, especially the team member fit that is determined in the previous activity. This activity is repeated until the entire outsourcing team is complete.

EVALUATION

In this section, we will evaluate our method by using the literature described earlier, and company input that is collected during four company presentations in their India offices, and two questionnaires sent out in the Netherlands.

Literature

To evaluate our method using literature, we will compare our method comparison with our proposed method.

Starting with the first section where we identify properties, project characteristics are marginally covered by Wi et al. (2009), when they determine the project keywords. Our activity however goes a lot deeper than that. Available team members are identified by both Wi et al. (2009) and Tseng et al. (2004), while Ahn et al. (2007) have three activities of what we consider to be part of determining potential team members, namely rating their reliability, quality and availability. Besides the activity of determining potential team members, these activities are also related to the activity that determines team member fit.

The next three activities from our method, identifying customer requirements, engineering characteristics, and create a matrix with these two elements, is in accordance with the methods of Tseng et al. (2004), Venkatamuni and Rao (2010), and Zakarian and Kusiak (1999). The identification of the team functions and the creation of the method to map these against the engineering characteristics is the same in our method as in the methods of Venkatamuni and Rao (2010) and of Zakarian and Kusiak (1999).

When looking at the next section of our method, where the matrixes are adjusted, we find that four other methods also add weights to their gathered data. These methods are those of Ahn et al. (2007), Tseng et al. (2004), Venkatamuni and Rao (2010), and Zakarian and Kusiak (1999). The ordering, or sorting, of the matrixes is also done in the methods of these latter three, where Tseng et al. (2004) and Zakarian & Kusiak (1999), in the same activity also group their matrixes. We have separated the grouping activity from the sorting in our method. The creation of a hierarchical model, which is optional in our method, is present in the method of Zakarian & Kusiak (1999).

The final activity, the actual selection of team members, occurs in all method we have modeled in this chapter.

Company input

All six companies that provided input indicated, to a greater or lesser extent, that communication is critical to project success. Without enquiring about communication, most of the companies explicitly mentioned its importance, while the others suggested it. The majority of the companies (four out of six) have indicated that they let their employees travel to the other teams to ensure good communication between the different (sub) teams. This clearly indicates the importance of a well composed team.

The project characteristics have an influence on the team composition. Logica mentions that the team composition depends on the scope of the project, while CoMakeIT indicated that process and product are more important than the technology that is used when looking at team member selection. CoMakeIT also stated that besides technical knowledge, team members are required to also have industry knowledge, indicating that the selection of potential team members, as well as determining the team member fit, are essential activities. This is supported by Logica, who claim that besides the skill level, it is also critical for a team member to poses domain knowledge. That is why Logica registers team member experience in their system.

While the influence of team size is being debated in literature, at CoMakeIT, they believe using micro scrum teams makes it easier to extend the core team and provides a better cohesion and knowledge sharing environment within the team. Starting with a small team that can be extended is also the preferred option of InnoVites. While this will probably not be the best situation for every company, it does indicate that every project manager should put some thought in what would be the optimal team size for their project. Itude indicated that the team size depends on the project.

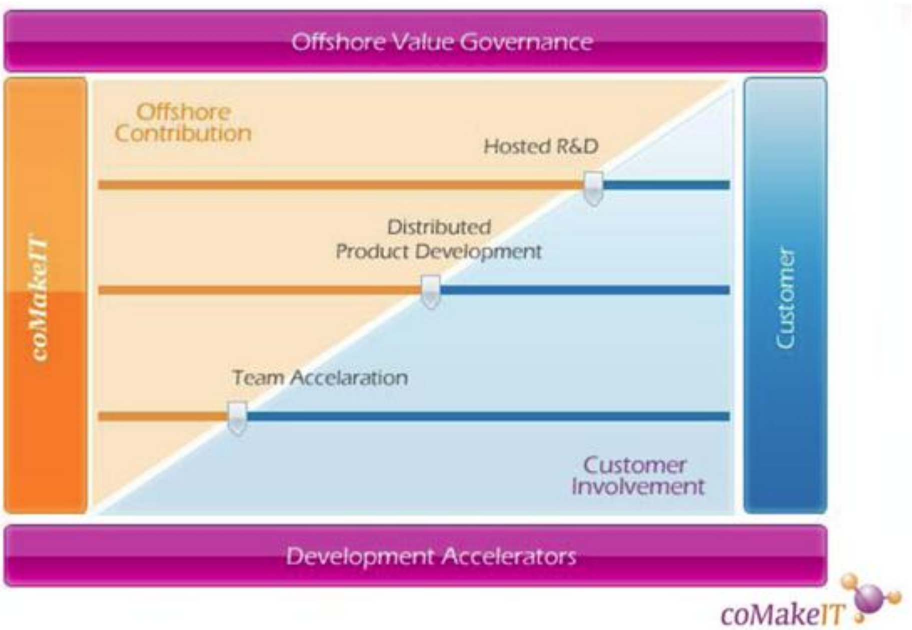

Figure 7: CoMakeIT Engagement Model

A number of companies have mentioned some of their ideas on the location of functional roles. Logica and CoMakeIT, for instance, have all of their developers located in India. Logica does have certain aspects onsite, such as the execution of the requirements phase, due to its criticalness, and the customer contact. CoMakeIT has their customer relations and sales in the Netherlands, as well as some local sales people in India. Microsoft indicated to prefer co-located team in certain aspects of their development process, and Itude mentioned certain location decisions based on the fact that it is the only way for them to guarantee the quality. This indicates a company can have different reasons to make certain decisions with regard to the location of functional roles or even individuals. CoMakeIT has developed a model, Figure 7, to differentiate between three levels of outsourcing they provide. This model shows, in a simple way, how each of the different levels changes the distribution of the location of team members. Since each project is different, and each company has their own preferences, the level of outsourcing, and thereby also the location of team members, can differ per project.

DISCUSSION

Prior to the creation of the method, we sent out a short questionnaire to find out how Dutch companies go about in composing their outsourcing team. We received two completed questionnaires after having contacted eight companies. The other four companies provided their input by information sessions in India, which were followed by a Q&A session. Since two Dutch and four Indian companies is not a broad foundation to base the method on, the proposed method is mainly based on existing literature.

To ensure that the team that results from the method actually performs adequately, product managers can choose to test the team on a small test project. This process can identify any potential problems that can then be resolved before the team actually starts to work on the outsourcing project.

While the method includes an activity where the fit of an individual is determined, we do not go into detail on how this should be done, since we believe this is an extensive topic. This does not mean this step should be ignored, but that it is a complex activity, which is often influenced by personal ideas and preferences. Since this is a critical step in the team composition, we decided to mention this step in the method, even though we do not go into detail on how this step should be performed. For examples of the influence of team member fit, see for instance Roland et al. (2008), Deibel (2005), and Leask, Hawe and Chapman (2001). We believe that is an area of further research, just as the validation in a company environment.

CONCLUSION

We propose a method for the composition of an outsourcing team. This method describes a number of activities that will help project managers in composing their software development team that is best suited for their specific outsourcing situation. An outsourcing team includes all people involved in the outsourcing activities, on both sides of the outsourcing activities, meaning onshore as well as offshore. The company data we collected from the questionnaires made it clear that team composition is done consciously, and does not include ad-hoc decision making. Since communication is found to have a great impact on team performance, in both literature and companies, a proper team composition is vital to project success.

The outsourcing team composition method proposed is mainly based on literature, and evaluated by using information from two Dutch companies that are involved in outsourcing software development to India, and four companies in India.

Further research in the area of team composition on outsourcing situations is necessary. The actual usefulness of our proposed method should be further investigated, for instance by performing case studies. This method can also be evaluated by adapting it to a project that will be executed in a classroom setting, where it will be used to form the student teams. We also invite researchers to investigate possible changes or extensions of this method to improve it for the benefit of the industry. It would be interesting to investigate if there are any differences in the team composition between near-shore and far-shore, or offshore, outsourcing. Also the determination of team member fit should be further investigated.

REFERENCES

Ahn, J., DeAngelis, D. and Barber, S. (2007) Attitude Driven Team Formation using Multi-Dimensional Trust, in *IEEE/WIC/ACM International Conference on Intelligent Agent Technology (*Silicon Valley, USA, 2007). 229-235.

Baykasoglu, A., Dereli, T. and Das, S. (2007) Project Team Selection Using Fuzzy Optimization Approach, *Cybernetics and Systems: An International Journal 38:* 155–185.

Boland, D. and Fitzgerald, B. (2004) Transitioning from a Co-Located to a Globally-Distributed Software Development Team: A Case Study at Analog Devices Inc, in *The 3rd International Workshop on Global Software Development. Co-located with ICSE 2004 International Conference on Software Engineering (*Edinburgh, Scotland, 2004). 4-7.

Bulka, B., Gaston, M. and desJardins, M. (2007) Local strategy learning in networked multi-agent team formation, *Autonomous Agents and Multi-Agent Systems 15*(1): 29-45.

Campion, MA., Medsker, GJ. and Higgs, AC. (1993) Relations between work group characteristics and effectiveness: Implications for designing effective work groups, *Personnel Psychology 46:* 823-850.

Chi, YL. and Chen, CY. (2009) Project teaming: Knowledge-intensive design for composing team members, *Expert Systems with Applications 36:* 9479–9487.

Cohen, SG. and Bailey, DE. (1997) What Makes Teams Work: Group Effectiveness Research from the Shop Floor to the Executive Suite, *Journal of Management 23*(3): 239-290.

Cohen, B. and Thias, M. (2009) The Failure of the Off-shore Experiment: A Case for Collocated Agile Teams, in *2009 Agile Conference (*Chicago, USA, 2009). 251-256.

Deibel, K. (2005) Team Formation Methods for Increasing Interaction During In-Class Group Work, in *ITiCSE 2005 (*Monte de Caparica, Portugal, 2005). 291-295.

Hong, S., Goor, G. van den and Brinkkemper, S. (1993) A Formal Approach to the Comparison of Object-Oriented Analysis and Design Methodologies, in *26th Hawaii International Conference on System Sciences 4 (*Hawaii, USA, 1993). 689-698.

Karduck, A. (1994) TeamBuilder: a CSCW tool for identifying expertise and team formation, *Computer Communications 17*(11): 777-787.

Korvin, A. de, Shipley, MF. and Kleyle, R. (2002) Utilizing fuzzy compatibility of skill sets for team selection in multi-phase projects, *Journal of Engineering and Technology Management 19*: 307-319.

Kussmaul, C., Jack, R. and Sponsler, B. (2004) Outsourcing and Offshoring with Agility: A Case Study, in C. Zannier, H. Erdogmus and L. Lindstrom (ed.) *Extreme Programming and Agile Methods - XP/Agile Universe 2004*, Berlin: Springer, pp. 147–154.

Lasser, S. and Heiss, M. (2005) Collaboration Maturity and the Offshoring Cost Barrier: The Trade- Off between Flexibility in Team Composition and Cross-Site Communication Effort in Geographically Distributed Development Projects, in *IEEE International Professional Communication Conference* (Limerick, Ireland, 2005). 718-728.

Leask, J., Hawe, P. and Chapman, S. (2001) Focus group composition: a comparison between natural and constructed groups, Australian and New Zealand Journal of Public Health 25(2): 152-154.

Leinonen, P., Järvelä, S. and Häkkinen, P. (2005) Conceptualizing the Awareness of Collaboration: A Qualitative Study of a Global Virtual Team, *Computer Supported Cooperative Work 14*: 301–322.

Magjuka, RJ. and Baldwin. TT. (1991) Team-based employee involvement programs: Effects of design and administration, *Personnel Psychology 44*: 793-812.

Mattarelli, E. and Gupta, A. (2009) Offshore-onsite subgroup dynamics in globally distributed teams, *Information Technology & People 22*(3): 242-269.

McGrew, JF., Bilotta, JG. and Deeney, JM. (1999) Software Team Formation and Decay: Extending the Standard Model for Small Groups, *Small Group Research 30*(2): 209-234.

Padmanabhan, S. (1996) Software Project Outsourcing, *Informatik 4*: 9-11.

Roland, S., Yttredal, O. and Moldskred, IO. (2008) Successful Interaction Between People, Technology and Organisation – A prerequisite for Harvesting the Full Potentials From Integrated Operations, in*SPE Intelligent Energy Conference and Exhibition (*Amsterdam, the Netherlands, 2008).

Scott, TJ. and Cross, JH. (1995) Team Selection Methods For Student Programming Projects, in *8th SEI CSEE Conference (*New Orleans, USA, 1995). 295-303.

Smith, KG., Smith, KA., Olian. JD., Sims, HP., Jr., O'Bannon, DP. and Scully, JA. (1994) Top management team demography and process: The role of social integration and communication, *Administration Science Quarterly 39*: 412-438.

Sutherland, J., Viktorov, A., Blount, J. and Puntikov, N. (2007) Distributed Scrum: Agile Project Management with Outsourced Development Teams, in *Hawaii International Conference on Software Systems* (Hawaii, USA, 2007).

Tsai, HT., Moskowitz, H. and Lee, LH. (2003) Human resource selection for software development projects using Taguchi's parameter design, *European Journal of Operational Research 151*: 167-180.

Tseng, TL., Huang, CC., Chu, HW. and Gung, RR. (2004) Novel approach to multi-functional project team formation, *International Journal of Project Management 22*: 147–159.

Venkatamuni, T. and Rao, R. (2010) Reduction of product development time by team formation method in lean manufacturing, *Indian Journal of Science and Technology 3*(5): 578-582.

Weerd, I. van de, and Brinkkemper, S. (2009) Meta-Modeling for Situational Analysis and Design Methods, in Syed, MR. and Syed, SN., (ed.) Handbook of Research on Modern Systems Analysis and Design Technologies and Applications, London: Information Science Reference, pp.35-54.

Wi, H., Oh, S., Mun, J. and Jung, M. (2009) A team formation model based on knowledge and collaboration, *Expert Systems with Applications 36*: 9121–9134.

Zakarian, A. and Kusiak, A. (1999) Forming teams: an analytical approach, *IIE Transactions 31*: 85-97.