Teamwork Quality and Project Success in Software Development: A Survey of Agile Development Teams

Yngve Lindsjørn , Dag I.K Sjøberg , Torgeir Dingsøyr ,
Gunnar R. Bergersen , Tore Dybå

PII: S0164-1212(16)30187-X
DOI: 10.1016/j.jss.2016.09.028
Reference: JSS 9851

To appear in: *The Journal of Systems & Software*

Received date: 15 April 2015
Revised date: 14 July 2016
Accepted date: 18 September 2016

Please cite this article as: Yngve Lindsjørn , Dag I.K Sjøberg , Torgeir Dingsøyr , Gunnar R. Bergersen , Tore Dybå , Teamwork Quality and Project Success in Software Development: A Survey of Agile Development Teams, *The Journal of Systems & Software* (2016), doi: 10.1016/j.jss.2016.09.028

Highlights

- We studied the effect of teamwork quality on project success in agile software teams.
- We ran a survey with responses from 477 respondents from 71 teams in 26 companies.
- Teamwork quality is perceived to have a small to large effect on team performance, depending of the rater.
- Teamwork quality is perceived to have a large effect on personal success.
- Teamwork quality and its effects are not greater in agile than in traditional teams.

Teamwork Quality and Project Success in Software Development: A Survey of Agile Development Teams

Yngve Lindsjørn[a], Dag I.K Sjøberg[a,b], Torgeir Dingsøyr[b,c], Gunnar R. Bergersen[a], Tore Dybå[b,a]

[a] Department of Informatics, University of Oslo, Norway {ynglin, dagsj, gunnab}@ifi.uio.no

[b] SINTEF, Trondheim, Norway {torgeir.dingsoyr, tore.dyba}@sintef.no

[c] Department of Computer and Information Science, Norwegian University of Science and Technology, Trondheim.

Abstract

Small, self-directed teams are central in agile development. This article investigates the effect of teamwork quality on team performance, learning and work satisfaction in agile software teams, and whether this effect differs from that of traditional software teams. A survey was administered to 477 respondents from 71 agile software teams in 26 companies and analyzed using structural equation modeling. A positive effect of teamwork quality on team performance was found when team members and team leaders rated team performance. In contrast, a negligible effect was found when product owners rated team performance. The effect of teamwork quality on team members´ learning and work satisfaction was strongly positive, but was only rated by the team members. Despite claims of the importance of teamwork in agile teams, this study did not find teamwork quality to be higher than in a similar survey on traditional teams. The effect of teamwork quality on team performance was only marginally greater for the agile teams than for the traditional teams.

Keywords: Agile development, Project management, Team performance, Teamwork quality, Work Satisfaction, Learning.

1 Introduction

Agile methods have been widely used in software engineering over the last decade. Even though agile methods emphasize teamwork more than traditional development methods do (Nerur et al., 2005), there is no thorough investigation of the effect of teamwork quality (TWQ) on project success in agile teams.

Agile development methods are used as an umbrella term to describe a number of development methods (Dingsøyr et al., 2012; Dybå and Dingsøyr, 2008). The agile manifesto[1] advocates "working software over comprehensive documentation", "customer collaboration over contract negotiation", and "responding to change over following a plan". Accordingly, to

[1] http://www.agilemanifesto.org

respond with agility to change, team members should work more closely together, have more frequent communication, be aware of other team members' work efforts, and be able to shift workload between persons. More specifically, the agile manifesto states that the best architectures, requirements, and designs emerge from self-organizing teams; the best communication is face-to-face communication; and business people and developers should work together daily. Collaboration and coordination are also central in the agile literature (Sharp and Robinson, 2010; Strode et al., 2012). In the most popular agile method, Scrum, work is organized in small, cross-functional teams with a facilitator and team members. Team members coordinate their work frequently, such as in daily stand-up meetings (Stray et al., 2016). Vinekar et al. (2006) explain that agile development and traditional development have different views on teamwork. *Agile development* is characterized by collaborative work, which requires multidisciplinary skills, pluralist decision making, high customer involvement, and small teams, while *traditional development* focuses on individual work, specialized skills, managerial decision making, low customer involvement, and larger teams.

Several studies have investigated the effect of teamwork quality (TWQ) on project success in traditional software teams (Hoegl and Gemuenden, 2001; Hoegl et al., 2003; Hoegl et al., 2004; Janz, 1999; Li et al., 2010; Ryan and O'Conner, 2009; Vinod et al., 2009). Hoegl and Gemuenden's frequently cited study (2001), for example, shows the effect of TWQ on team performance and team members´ success for a set of traditional software development teams.

Due to the lack of studies on the effect of TWQ in agile teams, we conducted a survey on this topic by replicating the study of Hoegl and Gemuenden (2001). Our research questions were:

RQ1: *What is the effect of TWQ on the performance of agile software teams?*

RQ2: *What is the effect of TWQ of team members' success in agile software teams?*

RQ3: *How does the effect of TWQ on team performance and team members' success differ between agile and traditional teams?*

The remainder of this article is organized as follows. Section 2 gives an overview of related work and describes the conceptual model of this work. Section 3 outlines the research method. Section 4 reports the results. Section 5 discusses the results, implications, limitations, and future work. Section 6 concludes.

2 Related Work and Conceptual Model

2.1 Teamwork in Software Development

Teamwork is obviously important in software development. In traditional development, the study by Faraj and Sproull (2000) showed a strong relationship between management of expertise and team performance. Another study demonstrated the importance of cooperative learning on project success for software development teams (Janz, 1999). In agile development, a few studies analyzed teamwork using team performance models, such as the one found in Moe et al. (2010). Sharp and Robinson (2010) described how agile development teams enable collaboration, co-ordination, and communication. Another study (Pikkarainen et al., 2008), focused on how agile development methods improve communication, and claimed that Scrum and XP practices improve both formal and informal communication. Maruping et al. (2009) demonstrated that XP practices of collective code ownership and coding standards could lead to increased technical quality of software products. A survey of success factors of agile development found that team capability was one of the factors (Chow and Cao, 2008).

Detailed models that show relationships between various aspects of teamwork quality and team performance have been used in studies of software teams; for example, those described in (Hoegl and Gemuenden, 2001; Salas et al., 2005; Dickinson and McIntyre, 1997; Janz, 1999). In this work, we focus on the factors described by Hoegl and Gemuenden (2001).

2.2 Teamwork Quality (TWQ)

We use the construct of teamwork quality conceived by Hoegl and Gemuenden (2001), which refers only to the quality of interactions. Measures of the task process, the task strategy, and the quality of the performance of the task activities performed by the individual team members are not the subject of this TWQ construct, nor are management activities such as task planning, allocation of resources, or management by objectives.

TWQ is conceptualized as a higher order construct and is based on Hackman's input-process-output model on group behaviour and effectiveness (Hackman, 1987) and derived from McGrath (1964). The six subconstructs of *communication, coordination, balance of member contribution, mutual support, effort,* and *cohesion* cover performance-relevant measures of internal interaction in teams; see Table 1. A more detailed description follows below.

Table 1. The TWQ Construct with Subconstructs

Subconstruct	Description
Communication	Frequency, formalization, and openness of the information exchange.
Coordination	Common understanding when working on parallel subtasks, and agreement on common work-down structures, schedules, budgets, and deliverables.
Balance of member contribution	The ability to employ the team members' expertise to its full potential. Contributions should reflect the team member's specific knowledge and experience.
Mutual support	Team members' ability and willingness to help and support each other in carrying out their tasks.
Effort	Team members' ability and willingness to share workload and prioritize the teams' task over other obligations.
Cohesion	Team members' motivation to maintain the team and accept that team goals are more important than individual goals.

2.2.1 Communication

Pinto and Pinto (1990) describe quality of communication within a team in terms of frequency and formalization of the information exchange. Frequency refers to how often communication occurs among team members and how much time is spent on it. Formalization refers to the degree of spontaneity in the communication. Communication that requires much planning and includes written status reports, etc., is considered formal, while spontaneous communication, such as talking in the doorway, chatting, talking in front of the screen, etc., is considered informal. Ideas and contributions are usually shared, discussed, and evaluated with other team members more quickly and efficiently in informal communication than in formal communication. It is also critical for the quality of communication that team members share their information openly with each other (Gladstein, 1984). Lack of open communication may hinder sharing of knowledge and experience that may be relevant for common tasks. In agile teams, the team members are often placed close together in open-plan offices to stimulate informal and open communication.

2.2.2 Coordination

Malone and Crowston (1994) describe coordination as "managing dependencies between activities." Such dependencies include shared resources, task assignments, and task/subtask relationships. Many activities in task processes are delegated to individual members. Harmonization and synchronization of these individual activities are important for the TWQ and project success (Tannenbaum et al., 1992; Brannick et al. 1995). Teams need to agree on common structures for breaking down work, schedules, and effort needed for the tasks. Coordination means that the teams must develop and agree upon a common task-related goal structure that has sufficiently clear subgoals for each team member. In agile teams, tasks are often selected or delegated when planning a new iteration. In a given iteration, some of the "user stories" (requirements) in the backlog are prioritized. A user story is often divided into several tasks. The workload for the tasks is estimated and each task is designed for or selected by one or more of the team members.

2.2.3 Balance of Member Contribution

The contribution of the task-relevant knowledge and experience of all members to the decision-making processes of the team may benefit the team (Hackman, 1987; Seers et al., 1995). Balanced contribution is critical in software teams with members who have expertise in different areas (core development, GUI development, system architecture, testing, etc.). If only one or even just a few team members dominate the discussions, the others may become less motivated for the work, which in turn may hamper overall team performance. The daily meetings (Stray et al., 2016) in agile teams support such a balance of member contribution.

2.2.4 Mutual Support

In software teams (as well as other teams working with innovative projects), the many inter-dependent tasks and the tight collaboration among individual team members together make cooperation a central issue. A competitive attitude—meaning self-interest at the expense of overall performance of the team tasks—may not benefit the work of the team (Tjosvold, 1998). The team members should be given assistance when needed and should take the other team members' contributions into consideration rather than trying to outdo the other team members. Some agile development methods include collective code ownership, which in turn stimulates mutual support and collaboration.

2.2.5 Effort

Team members should do their best to support the tasks of the team. Hackman (1987) describes conditions that support effort, and says it is important that "interaction among members minimizes social loafing and instead promotes a shared commitment among members to the team and its work." Prioritization of a team's tasks over other tasks is a good indicator of the effort that team members spend on common tasks (Hackman, 1987; Pinto and Pinto, 1990). In a focus group study of what hinders and what fosters effective teamwork in agile teams, prioritizing the team's tasks was perceived as one of the most important factors for achieving better team performance (Dingsøyr and Lindsjørn, 2013).

2.2.6 Cohesion

A common definition of *team cohesion* is "a dynamic process that is reflected in the tendency for a group to stick together and remain united in the pursuit of its goals and objectives" (Mudrack, 1989). Mullen and Copper (1994) distinguish between three aspects of team cohesion: (1) commitment to the team tasks, (2) interpersonal attraction of team members, and (3) group pride/team spirit. In a survey of 31 software teams, team cohesion was found to be the dominating factor when investigating the influence of team cohesion, team experience, and team capability on team performance (Lakhpanel, 1993). In agile teams, the members are often placed close together in office. According to the agile model of development, individuals and their interactions are valued over processes and tools, thus revealing the value of team cohesion.[2]

2.2.7 TWQ in Traditional vs. Agile Development

The TWQ subconstructs take different forms in traditional and agile development. Table 2 highlights some of the differences.

[2] www.agilemanifesto.org

Table 2. TWQ in Traditional and Agile Development

Subconstruct	Traditional teamwork	Agile teamwork
Communication	More formal. Written status reports to project manager.	Less formal. Spontaneous communication (talking in the doorway, chatting, talking in front of the screen).
Coordination	Strong leadership. Project manager makes decisions; estimates, prioritizes, and delegates tasks in particular.	Not strong leadership. Self-organizing teams. The team makes decisions; estimates, prioritizes, and delegates tasks in particular.
Balance of member contribution		In cross-functional teams, it is expected that all team members contribute. Daily meetings support balance of member contribution.
Mutual Support	Hierarchical management does not facilitate mutual support among team members.	Collective code ownership, daily meetings, and retrospective meetings stimulate mutual support and collaboration.
Effort	Less focus on the team per se.	Large team focus, e.g., daily meetings. Facilitator helps protect team members from tasks outside the team.
Cohesion	Hierarchical management and more formal communication may not support cohesion.	Focus on interactions among team members, who often are physically placed together.

2.3 Projects Success in Software Projects

The conceptualization of project success as a multi-variable construct is described in (Gladstein, 1984; Hackman, 1987; Sundstrom et al., 1990; Pinto et al., 1993; Denison et al., 1996). This literature distinguishes between task-related outcomes (e.g., quality of the software product and adherence to cost and budget) and people-related outcomes (e.g., team member satisfaction and viability of the team). In this study, we use the outcome categories of team performance and team members´ success; see Table 3.

Table 3. Team Performance and Team Members' Success

Construct	Subconstruct	Description
Team performance	Effectiveness	Degree to which the team meets expectations regarding quality of the outcome, e.g., functionality, robustness, reliability, and performance. Reflects a comparison of intended versus actual output.
	Efficiency	Degree to which the team meets expectations regarding time, cost, adherence to schedule, and adherence to budget. Reflects a comparison of intended versus actual input.
Team members' success	Work satisfaction	Degree to which team members are motivated to participate in future team projects.
	Learning	Degree to which team members learn social, project management, technical, and creative skills.

2.3.1 Team Performance

Team performance may be defined as the "extent to which a team is able to meet established quality, cost, and time objectives" (Hoegl and Gemuenden, 2001). Many team performance models and teamwork frameworks describe TWQ and its relation to team performance in general, e.g., (Mathieu et al., 2008; Cohen and Bailey, 1997; Rasmussen and Jeppesen, 2006).

Team performance and team effectiveness are often used synonymously in the literature; sometimes team performance is part of team effectiveness, e.g., (Cohen and Bailey, 1997), and sometimes team effectiveness is part of team performance, e.g., (Hoegl and Gemuenden, 2001). Most of the models of team performance (or team effectiveness) originate from management science and psychology (Salas et al., 2007). In this study, *team performance* is described in terms of the subconstructs *effectiveness* and *efficiency*. Effectiveness refers to the degree to which the team meets expectations regarding the *product quality*. The quality of a software product is often measured by the customer, and includes aspects such as functionality, robustness, reliability, and performance. Efficiency refers to the degree to which the team meets expectations regarding *project quality*.

2.3.2 Team Members´ Success

Teams should work in a way that increases the motivation of team members and their ability to engage in future teamwork (Hackman, 1987; Sundstrom et al., 1990). It is obvious that the success of team members increases their motivation for working on future projects of the same team. Collaborating with other team members also provides the opportunity for learning social, management, technical, and creative skills. In some team performance models, e.g., (Janz, 1999), learning is defined as one of the aspects of TWQ, and thus is seen as a contribution to the success of a project—its outcome—and not as a part of the outcome itself.

2.4 Conceptual Model

We investigate the effect of TWQ on two aspects of team outcome: team performance and team members´ success, as shown in the conceptual model of Figure 1.

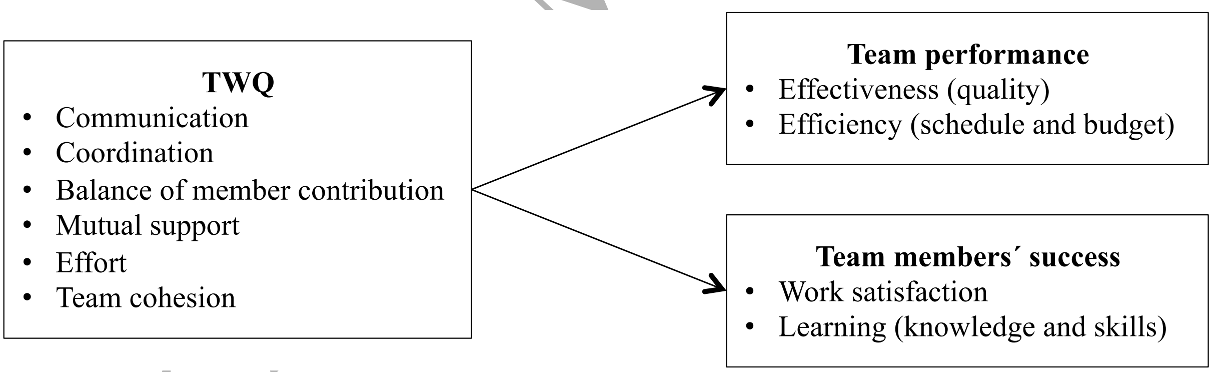

Figure 1: Conceptual model (taken from Hoegl and Gemuenden (2001)).[3]

Hoegl and Gemuenden (2001) give a detailed account of both the theoretical rationale and empirical evidence for the positive relationship between TWQ and both software team performance and team members´ success. The TWQ construct provides a measure of the collaborative team-task process, which focuses on the quality of interactions. In other studies,

[3] Hoegl and Gemuenden (2001) introduce the construct "personal success", defined in terms of work satisfaction and learning. However, the items in the questionnaire start with "the team", "we", or "team members". Consequently, the items indicate references to teams rather than individuals. We therefore use the term "team members' success" instead of "personal success".

cooperation within teams has also been shown to influence both team performance and team members´ success.

We are unaware of any theory or earlier studies that should indicate a difference between traditional and agile development regarding the effect of TWQ on team performance and team members´ success. Nevertheless, we explore such a difference in RQ3.

3 Research Method

This survey was a differentiated replication (Lindsay and Ehrenberg, 1993). Hoegl and Gemuenden (2001) studied traditional teams; we studied agile teams. For simplicity, we will refer to the two surveys as, respectively, the *traditional survey* and the *agile survey*.

3.1 Study Sample

The criteria for participating in our study was that a team had used agile methodology for at least one year, and it had delivered software to a customer at least once. Teams were recruited at the Norwegian Agile Conference in November 2011, which attracted approximately 400 participants from 100 companies. We recruited 71 teams from 26 companies as participants for our survey. These teams included 76 team leaders, 78 product owners, and 323 team members. Twelve companies contributed with only one team in the survey; the other companies contributed with 2 to 11 teams. The companies operated in the application domains of finance, telecommunications, shipping, oil, and consultancy, both within the private sector (75%) and the public sector (25%). They varied from small consulting companies with less than 10 developers to large companies with several hundred developers. Among the recruited teams, 16 were "offshore" teams located in India, China, and Malaysia. Most of the teams used Scrum (69%); the other ones used Kanban (19%) and a mix of Scrum, Kanban, and XP (12%). The Scrum teams used daily stand-up meetings, iteration planning, iteration reviews, and retrospective meetings. The iteration interval was 2.8 weeks on average. Daily stand-up meetings were also used in the Kanban teams. The release interval was 4.3 months on average for all the teams.

Table 4 shows that there were relatively more females among team leaders and product owners (approximately 1 in 3) than among team members (approximately 1 in 6). Some of the team leaders had other job functions in the team (mostly developer), but they answered the survey in the role of team leader. Furthermore, some of the team members had more than one job function in the team. The team members' primary job function was developer (73%), tester (14%), and system architect (7%). Other roles were GUI designer, support staff, configuration manager, and QA responsible.

Table 4. Characteristics of Respondents

Role	N	Age		Gender		Education		Years of experience with			
				M	F	Bachelor	Master	Development		Agile methods	
		Mean	S.D.	Perc.	Perc.	Perc.	Perc.	Mean	S.D.	Mean	S.D.
Team leader	76	36.6	6.7	66.6	33.4	40.0	56.0	10.7	6.7	3.8	1.6
Product owner	78	41.7	8.4	66.7	33.3	34.6	53.8	12.3	8.0	3.4	1.9
Team member	323	34.1	7.8	82.4	17.6	48.6	46.0	8.9	6.9	3.1	1.9
All	487	35.7	8.3	77.2	22.8	44.9	48.9	9.8	7.1	3.3	1.9

One may question whether all the teams that participated in our survey were "agile". This is not a trivial issue because there is no clear definition of what an agile team is. However, we consider the teams in this study as being agile because (1) all the teams stated in the survey that they used Scrum, Kanban, XP, or a hybrid, and (2) the contact persons of the companies that we approached at the Norwegian Agile Conference claimed that the teams that participated in the survey were agile.

3.2 Data Collection

For approximately half of the teams, we visited their workplace to explain the purpose of the survey and to collect answers to a questionnaire. The teams that we were unable to visit (including all the offshore teams) received and submitted the questionnaire electronically or by post.

For each item in the questionnaire (Table 7, Appendix A), the respondents were requested to indicate their agreement with the statement on a scale from 1 (strongly disagree) to 5 (strongly agree) from their personal point of view, rather than from whatever they thought might be the whole team's point of view. The team members responded to all 61 items in the questionnaire, whereas the team leaders and product owners responded only to the 15 items that concerned team performance specifically.

3.3 Investigated Variables

As in the traditional survey, the unit of our study was the team itself. For teams with more than one team member, team leader, or product owner, we use the arithmetic mean of the responses as the value of the team. To make our results comparable with the results reported in the traditional survey, we only included teams for which at least one team member, project leader, and product owner responded. A total of 25 teams were rejected because one or more roles had missing responses, leaving a total of 71 teams for analysis.

Table 5 shows the descriptive statistics for the 14 variables that are used to measure TWQ, team members' success, and team performance as evaluated by, respectively, team members, team leaders, and project leaders. Each variable is represented as the arithmetic mean of the individual items that comprise the variable. All variables can be regarded as normally distributed according to the Shapiro-Wilk test of normality as implemented in SPSS 23, except team leader effectiveness ($p = 0.011$), team leader efficiency ($p = 0.025$), and product owner efficiency ($p = 0.010$). We found only marginal differences in reported results throughout this paper when removing observations that caused lack of normality. To conserve statistical power, we therefore retained those observations.

Table 5. Descriptive Statistics of the Investigated Variables

Construct	Rater	Variable	No of Items	Mean	Std. Dev.	Alpha
Teamwork quality (TWQ)	Team member	Communication	10	3.98	0.26	0.73
		Coordination	4	3.78	0.29	0.72
		Balance of member contribution	3	3.96	0.30	0.58
		Mutual support	7	4.06	0.29	0.85
		Effort	4	3.98	0.34	0.76
		Cohesion	10	3.92	0.28	0.86
Team members` success	Team member	Work satisfaction	3	4.11	0.32	0.84
		Learning	5	4.08	0.30	0.83
Team performance	Team member	Effectiveness_TM	10	3.85	0.33	0.92
		Efficiency_TM	5	3.72	0.44	0.87
	Team leader	Effectiveness_TL	10	3.85	0.45	0.81
		Efficiency_TL	5	3.68	0.55	0.70
	Product owner	Effectiveness_PO	10	3.84	0.41	0.84
		Efficiency_PO	5	3.76	0.66	0.88

Table 5 also reports Cronbach's alpha, which is a statistic for internal-consistency reliability. Cronbach's alpha values were calculated at the team level, that is, on the aggregated values. Nunnally and Bernstein (1994) consider a Cronbach's alpha higher than 0.7 as satisfactory. All variables were thus satisfactory, except *balance of member contribution*, which had an alpha value of 0.58. The correlation matrix for the investigated variables is shown in Table 6.

Table 6. Correlations between Investigated Variables and Differences in Correlations between Studies

Variables	1	2	3	4	5	6	7	8	9	10	11	12	13	14
(1) Communication		-0.23	-0.16					0.14					-0.13	-0.16
(2) Coordination	0.35		-0.27	-0.16				-0.12		0.15			-0.16	
(3) Balance of m.con.	0.57	0.27			-0.12	-0.11	-0.10	0.18	0.11		0.20			-0.19
(4) Mutual support	0.76	0.39	0.69					0.15	0.13		0.21			-0.13
(5) Effort	0.62	0.46	0.53	0.60			0.22		0.13	0.11			-0.21	-0.28
(6) Cohesion	0.75	0.44	0.63	0.77	0.70				0.14	0.11			-0.20	-0.27
(7) Work satisfaction	0.76	0.50	0.62	0.79	0.70	0.82							-0.17	-0.17
(8) Learning	0.66	0.22	0.67	0.67	0.49	0.70	0.71		0.20				-0.13	
(9) Effectiveness_TM	0.50	0.38	0.56	0.57	0.49	0.56	0.71	0.58			-0.12	-0.13	-0.20	
(10) Efficiency_TM	0.37	0.43	0.40	0.41	0.45	0.49	0.55	0.31	0.69			-0.23	-0.14	-0.34
(11) Effectiveness_TL	0.24	0.20	0.39	0.27	0.20	0.19	0.30	0.10	0.42	0.28			-0.28	-0.20
(12) Efficiency_TL	0.13	0.23	0.27	0.14	0.16	0.19	0.24	0.01	0.20	0.28	0.61		-0.27	-0.35
(13) Effectiveness_PO	0.09	0.06	0.02	0.22	-0.07	-0.05	0.07	0.06	0.17	0.06	0.12	-0.03		
(14) Efficiency_PO	0.01	0.02	-0.03	0.04	-0.10	-0.08	0.01	-0.04	0.10	0.03	0.10	0.12	0.68	

Note: The lower triangular part of the matrix shows Pearson's product moment correlations between the investigated variables. Correlations (two-tailed) above 0.23 are significant at $p < 0.05$; correlations above 0.30 are significant at $p < 0.01$; and correlations above 0.38 are significant at $p < 0.001$. N = 71 for all variables. The upper triangular part of the matrix shows differences in correlations between the two surveys. Positive numbers means higher correlations in the agile survey; negative numbers means higher correlations in the traditional survey. Only absolute differences above 0.1 are shown. Differences between rater categories for effectiveness and efficiency are further explained in Section 5.1.

3.4 Statistical Analysis and the Model Tested

Confirmatory statistical analysis was conducted using Structural Equation Modeling (SEM) as implemented in the lavaan package (Rosseel, 2012) using R (R Core Team, 2015). No data was missing. All parameters are estimated using maximum likelihood with the "wishart" option.

SEM allows the specification of a system of equations for two main types of models simultaneously (Anderson and Gerbing, 1998). First, the *measurement model* specifies how a set of variables can be used to represent a concept of interest. A purely data-analytic distinction is whether a variable is observed or latent (Borsboom, 2008). To be considered "observed", data must be directly available (as the 14 variables reported in Tables 5 and 6). In contrast, latent variables are estimated from observed variables plus error, or from aggregations of other latent variables.[4] In this study, the investigated measurement models are as follows: TWQ is represented as a latent variable with six observed variables where factor loadings can vary (i.e., a congeneric model). There are four other latent variables: team members' success and, the team performance reported by, respectively, team members, project leaders, and product owners. Each of these four latent variables is represented by two observed variables with equal factor loadings (i.e., a tau-equivalent model). Measurement error in all the five models is specified to be uncorrelated.

Second, the *structural model,* specifies how concepts are supposed to be related, as in linear regression. However, a difference is that in linear regression, the analysis is performed on observed variables, whereas in a structural model, the analysis is performed on latent variables. The structural model that we investigate has four paths, from TWQ to each of the four dependent latent variables.

The testing of model fit for the specified measurement and structural models is covariance based; that is, a covariance matrix generated on the basis of the models is compared with the covariance matrix of the actual data. Differences between these two covariance matrices will in turn inform the question of whether the data fits the specified model. The covariance matrix for the agile survey was calculated from the variables reported in Tables 5 and 6. The covariance matrix for the traditional survey was calculated using the correlation matrix, means and standard deviations reported in Hoegl and Gemuenden (2001). We report model fit by the Root Mean Square Error of Approximation (RMSEA) and its 95% confidence interval. RMSEA values below 0.05 indicate a close model fit; values around 0.08 indicate an acceptable model fit; and values above 0.10 indicate an unacceptable model fit.

3.5 Model Fit

The confirmatory factor analysis of the measurement model for TWQ indicated an almost close model fit ($\chi^2[9] = 10.73$, $p = 0.30$, RMSEA [95% confidence interval] = 0.052 [0.000–0.150]).[5] However, as indicated by the wide confidence interval for RMSEA, one cannot

[4] We use only reflective measurement models in this work, where errors are associated with the observed variables.

[5] The alternative tau-equivalent model indicated unacceptable model fit for TWQ ($\chi^2[14] = 23.73$, $p = 0.05$, RMSEA = 0.10 [0.005–0.167]); the congeneric model of TWQ is therefore analyzed throughout this paper.

claim with sufficient confidence that the model fits because the upper 90% confidence interval (0.150) is above an unacceptable value (i.e., > 0.10). Note that the Kaiser criterion and Cattell's scree plot were acceptable: one component could be extracted with an eigenvalue above 1 and there was a clear "elbow" in the eigenvalue-component plot. However, these two criteria are more akin to heuristics and are more easily satisfied than the confirmatory tests we report.

All factor loadings for the involved constructs were significant ($p < 0.001$). The overall model fit of the investigated (measurement and structural) model was otherwise somewhat worse than for the measurement model of TWQ alone ($\chi^2[71] = 100.64$, $p = 0.012$, RMSEA = 0.077 [0.038–0.110]). In addition to low statistical power for rejecting a poorly fitting model, there were also problems with highly correlated indicators resulting in a nonpositive definite matrix during estimation, see, e.g., (Wothke, 1993) and negative error variance. By removing the two latent variables of team members' success and project owner performance (along with their four indicator variables), these problems were resolved and the overall model fit improved ($\chi^2[34] = 38.26$, $p = 0.28$, RMSEA = 0.042 [0.000–0.100]), with negligible changes to regression weights and factor loading for the remaining variables.

4 Results

Section 4.1 reports the results for Research Questions 1 and 2. Section 4.2 reports the results for Research Question 3.

4.1 Relationship between TWQ and the Dependent Variables

Figure 2 shows the results for the investigated model. The investigated variables are represented as rectangles and the constructs are represented as ellipses (i.e., latent variables). Arrows with no origin shows error variance, and arrows from latent variables to observed variables show the standardized factor loadings; all factor loadings are significant at $p < 0.001$. Arrows from TWQ to the four dependent latent variables show the (structural) path coefficients. In the figure, *coordination* had the highest error variance (0.78) and the lowest factor loading on TWQ (0.47); the lowest structural path coefficient was from TWQ to team performance as rated by product owner (0.06). Note that the path coefficients are estimated and standardized so that an increase of one standard deviations in the independent variable will result in an increase in standard deviation as given by the estimated coefficient. For example, the estimated coefficient of 0.997 (rounded out to 1.00 in the figure) between TWQ and team members' success imply that an increase of 1 SD in TWQ will with a 95% confidence interval result in an expected increase of 0.95–1.05 SD in team members' success.

Regarding Research Question 1, TWQ significantly affects team performance when performance is rated by team members ($p < 0.001$) and team leaders ($p = 0.010$). The effect is large for the rating by team members ($R^2 = 0.466$) and medium for the rating by team leaders

(R^2 = 0.104). TWQ has no effect on team performance when performance is rated by the product owners ($p = 0.593$, R^2 = 0.004).

Regarding Research Question 2, TWQ significantly affects team members' success, which was rated by only team members ($p < 0.001$). The effect is large, almost unity (R^2 = 0.994).

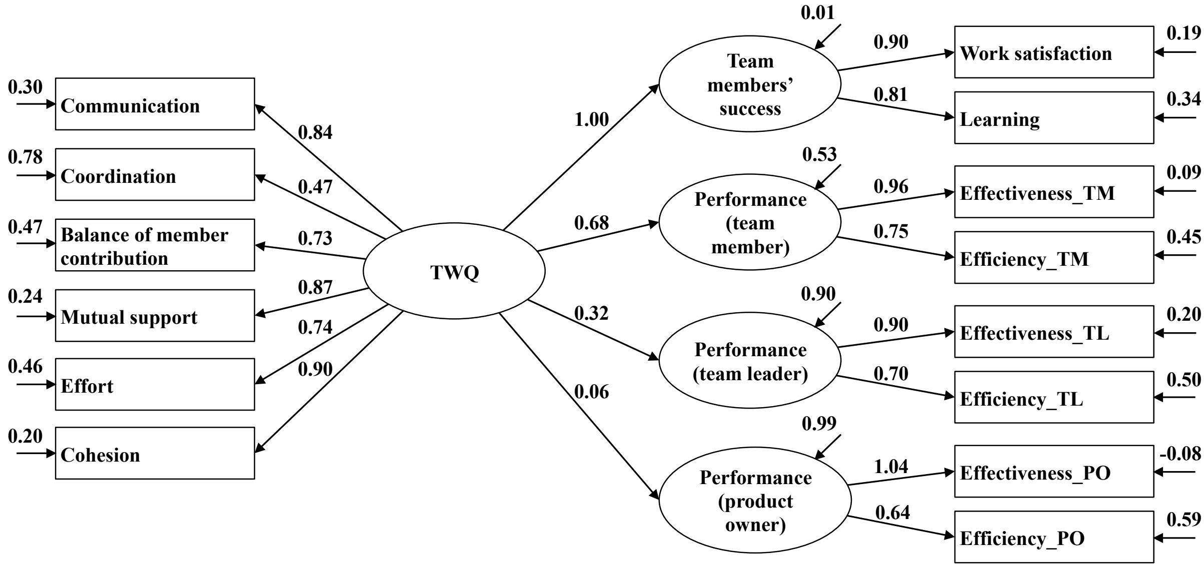

Figure 2: Standardized factor loadings, (structural) path coefficients, and error variances for the investigated model.

4.2 Differences Between Traditional and Agile Teams

Using the model described in Section 3.5, Hoegl and Gemuenden's (2001) data displayed an unacceptable confirmatory model fit (χ^2[71] = 224.90, RMSEA = 0.123 [0.105–0.141]. Nevertheless, the factor loadings of TWQ in the two surveys are highly similar; the largest difference is that the data from the agile survey has a lower loading for *coordination* (0.47) than the data from the traditional survey (0.62).

The results from the two surveys also show some minor differences in the standardized structural coefficients for the path from TWQ to the four dependent variables. Figure 3 shows that the coefficients in the agile survey are higher for team members' success (R^2 = 0.994), performance rated by team members (R^2 = 0.466), and performance rated by team leaders (R^2 = 0.104) than in the traditional survey but lower for performance rated by product owners (R^2 = 0.004). The figure also shows that the standard error is larger the smaller the regression weight is, in both surveys.

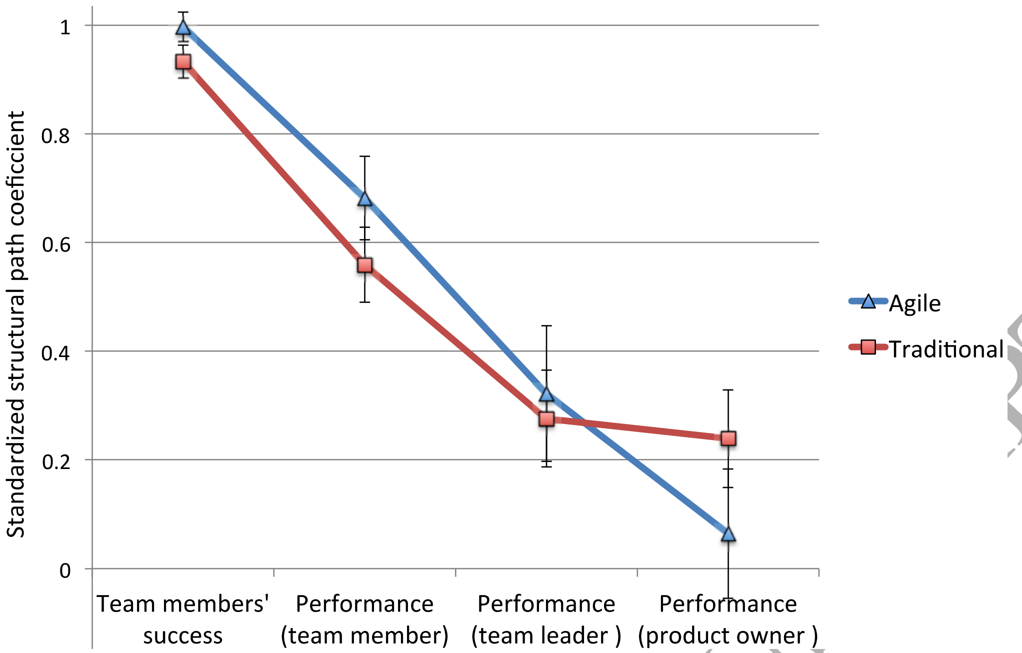

Figure 3: Standardized path coefficients from TWQ to the four dependent variables; the whiskers show the standard error of measurement.

5 Discussion

This section discusses the differences in evaluations of team performance, implications for practice and theory, limitations, and future work.

5.1 Differences in Evaluation of Team Performance among Raters

This survey revealed large differences among team members, team leaders, and product owners in how team performance is evaluated. Figure 4 shows the correlations between the three categories of raters, which can also be found in Table 6. The leftmost radar chart shows the correlations for product quality, the rightmost chart shows project quality. Overall, the figure shows that the agile survey has weaker agreements among raters than in the traditional survey. One may expect that close communication between team members, team leaders, and product owners in agile development will lead to consensus in the evaluation of team performance. On the other hand, the traditional plan-driven approach involves more documentation and reporting, which may make it easier to have a shared view of team performance.

There may be several reasons for the differences between the rater groups. Regarding *product quality*, product owners, and to some extent team leaders, may consider the product more from the customer's point of view (functionality, usability, etc.) than do team members, who may emphasize code qualities of the product (maintainability, testability, etc.), which are invisible to the customer. Fig. 4 shows that the agile and the traditional survey both have highest agreement between team members and team leaders regarding product quality (r =

0.42 and *r* = 0.54, respectively). Consensus between product owners and the two other raters is low in the agile survey (*r* = 0.17 for team member and *r* = 0.12 for team leader). The consensus is higher in the traditional survey (*r* = 0.37 and *r* = 0.40, respectively).

Regarding *project quality*, product owners and team leaders may have a better overview than team members of lead time and cost besides development (overall management costs, infrastructure costs, etc.). Particularly in agile teams, team members tend to focus more on costs only within the current iteration or release of the system, which may explain that the correlation between product owners and team members are much less in the agile survey (*r* = 0.03, i.e., non-existent) than in the traditional survey (*r* = 0.37).

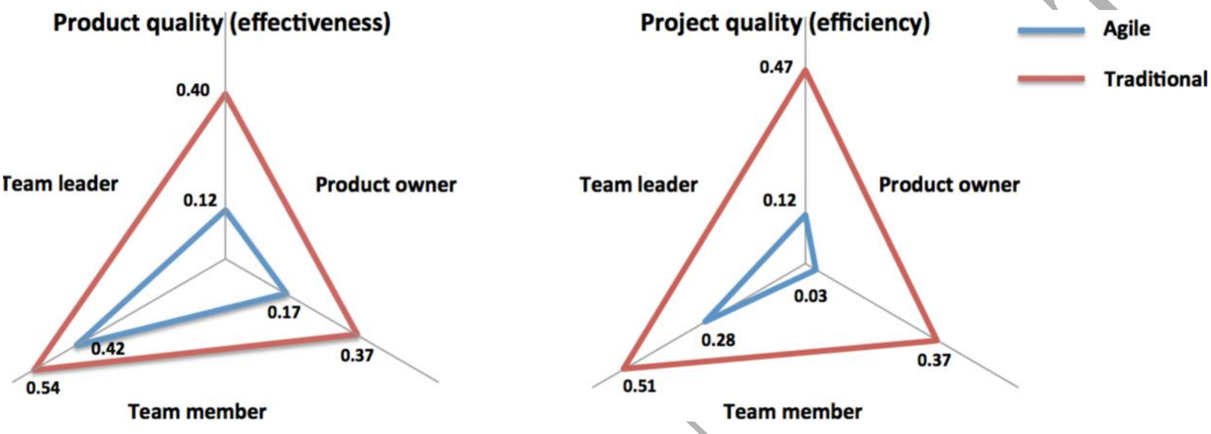

Figure 4. Correlations between raters' evaluation of team performance.

The fact that the team members rated both TWQ and team performance may have created *implicit models* among team members as proposed by Gladstein (1984): "It appears that individuals have implicit models of how certain modes of group process 'should' benefit performance and attribute good outcomes to the group when the appropriate process has been instituted." The presence of implicit models may have induced a bias that may explain differences in the rating of team members versus the other raters. In particular, if team members consider TWQ to be high, they may also consider performance to be high (and vice versa). More generally, MacKenzie and Podsakoff (2012) showed in a meta-analysis that the correlations between the independent and dependent variables inflated from 133 to 304 percent when the same rater evaluated both.

The team leaders and product owners did not evaluate TWQ, but the team leaders worked more closely with the team than did product owners. Consequently, the team leaders might have had a better understanding of TWQ than the product owners, which might have caused them to also have an implicit model that teams with high TWQ also have high team performance. Still, their implicit model would have been "weaker" than that of the team members. Product owner evaluations of team performance are much less likely to be influenced by implicit models, given their limited knowledge of the TWQ.

Hoegl and Gemuenden (2001) explain another implicit model. Because managers (product owners in our survey) lack detailed information about relevant performance measures, they "evaluate the outcomes based on their general impression of the expertise of the team leader or other team members, rather than solely considering actual performance" (Hoegl and Gemuenden, 2001). In other words, the managers evaluate team performance high if they consider the expertise in the team to be high. Furthermore, Cohen and Bailey (1997) observed, "Team members tend to rate the team's performance high if the team has engaged in healthy internal processes, such as collaboration and resolution of conflict. Managers, who may be less intimate with the group's internal dynamics, rate a team highly according to more external factors, like the amount of communication the group has with external agents" (Cohen & Bailey, 1997). In agile teams, product owners can be seen as an external agent and thus evaluate the team performance according to how much the team communicates with the product owner and other external agents.

5.2 Implications for Practice

A practical implication of the differences in evaluation of performance in the agile survey concerns whose perception of performance should be taken into account when making efforts to improve performance. For example, team members may focus more on internal code quality; product owners, who are supposed to represent the perspective of the customers, may focus more on usability and other more easily inspected parts of the system, in addition to project lead time and cost. Early in the project, when efforts are being made to improve development processes and to achieve successful projects, stakeholders need to clarify those aspects of performance the team should optimize.

Which aspects of performance that will be optimized have consequences for the importance of TWQ. If product quality in a project is most important, TWQ should be emphasized, but if the main focus is to meet expectations regarding time and cost, TWQ appears less important. This is hardly a surprise. Generally, one needs to consider the trade-offs within the "magic triangle": product quality, time, and cost.

Mutual support is the investigated variable of TWQ with the largest effect on team performance (Table 6); that is, quick resolution of conflicts, constructive discussions, respect for suggestions and contributions made by other team members, the ability to reach consensus, and good cooperation are considered particularly important in agile teams. One explanation for the importance of mutual support is that there is no leader who can deal with conflicts and manage other problems that may occur in agile, self-organized teams. Such teams may be more vulnerable to lack of mutual support than teams with a traditional management style. Therefore, agile teams should be particularly concerned with developing measures (such as involving an unbiased third party, emphasizing giraffe language, etc.) for dealing with conflicts and handling a lack of mutual respect.

Generally, given that agile teams are self-organizing and have less focus on plans and documents than traditional teams, we had expected that TWQ was more important for team performance than in traditional teams. However, we found only small differences between the two surveys regarding the importance of TWQ. The similarity in the mean values of the TWQ variables themselves was also unexpected to us (the values were actually a bit higher in the traditional survey) given the focus on teamwork in agile development. An explanation may be that while TWQ has in reality increased, the expectations in today's agile teams are higher than in traditional teams over a decade ago, resulting in similar values. Another explanation might be the restriction of range (Shadish et. al, 2002) in the response scale of these variables. In the traditional study, the values were already close to 4 on a scale with 5 as a maximum.

5.3 Implications for Theory

Our results have several implications for theory. First, although a theoretical distinction is possible between TWQ and team members' success, we found no empirical distinction between the two concepts. The variables involved in measuring the two concepts were correlated to the point that they are almost the same (i.e., correlated by unity) after controlling for measurement error. One explanation for the high correlation is rater bias (cf. Section 5.1).

Second, a related point concerns the extent to which the structural relations are correctly specified in the theoretical model. The elimination of rival models that can explain the available data equally well is a major challenge in research. Although we had no reason a priori to dispute the direct links from TWQ to the four dependent variables, we believe there are also other alternatives with merit. For example, team members' success could mediate the relation between TWQ and team performance. Moreover, it is clear that at least for the agile teams that the relation with TWQ and product owner performance is low to none.

A third point relates to the expected effect of TWQ on performance when performance is defined both in terms of project and product quality. Project quality, including schedule and budget performance, may in some situations be negatively correlated with TWQ. For example, a dictator management style certainly affects TWQ negatively—at the same time, it *might* get things done faster.

5.4 Limitations

The comparison between the two surveys might be affected by methodological differences. In the traditional survey, data was collected *after* project completion while it was collected *during* the project in the agile survey. Still, there are good reasons for ongoing data collection in a project. First, the participants involved can more easily report day-to-day affairs rather than recalling what occurred some time ago. Second, the survey respondents might be less influenced by how other team members, leaders, and others perceive and express their opinions about the outcome of the project. Another difference between the two surveys is that, in the traditional survey, all the teams worked exclusively on one project, while in the agile survey, half of the teams worked on several projects that involved other teams. Furthermore, the traditional survey had approximately four respondents per team on average. The agile survey had approximately seven respondents per team on average.

To compare our results with those of the traditional study, we followed the procedure of the traditional survey by aggregating the opinions of several respondents of a team into a single response. The traditional survey investigated whether such an aggregation led to bias but found nothing. We did not investigate whether there was such a bias in our study.

One might question to what extent the teams were "agile" in our survey and "traditional" in the traditional survey. In Section 3.1, we justified the identification of agile teams for our survey. Since the participants of the survey by Hoegl and Gemuenden (2001) were not asked explicitly about development methods, we need to justify the identification of the teams in that survey as having a traditional approach. There are two strong indications. First, the waterfall model or a similar plan-driven model with a sequential approach was the most common development model before 2001. The agile manifesto was not formulated before

2001, and the first book on Scrum was published in 2001 (Schwaber and Beedle, 2001). Second, the context of the traditional survey was large organizations. The teams were recruited from four German software laboratories, which varied in size from 100 to 500 software developers. All four laboratories were part of larger organizations, two of which were independent operations of the same U.S. parent company. In the relatively unlikely case that these teams did have an agile approach, it is still interesting that the results of our Norwegian survey obtained very similar results to a German survey conducted 15 years earlier.

The response rate at the company level was 26%. At the team level, the response rate was about 30%; that is, from the range of 200 to 220 agile teams in these companies (for a few of the companies, we were not informed of the exact number of teams), we managed to recruit 71 teams. To what extent these teams are representative of agile software teams within or outside Norway is an open question. It may be that companies that attend the Norwegian agile conferences have more positive attitudes toward agile development than other companies. Consequently, the agile teams of this survey might evaluate TWQ, team members´ success, and team performance higher than agile teams in companies that show less interest in agile development.

We are unaware of any survey in software engineering that claims that its sample is representative of a given industry. Still, the more companies that are represented in the sample, the less the likelihood is that a specific culture of a company will bias the results. The traditional study collected responses from 145 teams in three companies. We collected data from 26 companies, that is, an average of 2.7 teams per company. We included most teams (11) from the largest company, which is a public administration agency. There may still be a bias in our results towards certain companies but to a much lesser extent than may be the case in the traditional survey.

5.5 Future Work

In the survey, we collected additional data about the respondents and the teams that may be used to identify differences among various subgroups with respect to TWQ and its effect. We intend to investigate, for example, the effect of offshore versus local teams, public versus and private sector, application domains (telecommunications, consultancy, shipping, and oil), agile practices (e.g., daily stand-up meetings), and level of team interaction with the product owner.

We have established a relationship with a large organization with many development teams. In that company, we will further investigate the effect of TWQ by collecting a more refined and more objective set of performance data than is possible to obtain in a survey.

Another topic for further work is to refine and possibly simplify some of the constructs in the survey. In our opinion, some items of TWQ have a dubious linear relation to project quality and, thus, team performance. One example is “there is frequent communication within the team.” Team performance will not improve if the team members communicate above a certain threshold because they cannot communicate all the time and still perform well. Further, the statement "our team was able to reach consensus regarding important issues" may also be problematic, because consensus might be detrimental for project schedule. A curvilinear or even a quadratic relation may be more appropriate in these situations. A further problem is that team performance is defined in terms of both product quality and project quality. These two variables are often negatively related in a trade-off function; that is, one can increase quality by spending more time, or reduce quality to save time. On the other hand, the best performing teams will both deliver better product and project quality, something that should be taken into account in future work where team performance is operationalized; see (Bergersen et. al, 2011).

6 Conclusions

The present survey found TWQ and team performance to be highly related when team members rated these two concepts. Furthermore, the correlation between TWQ and team members' success—their work satisfaction and learning—approach unity. One interpretation is that the team members consider TWQ and team members' success as indistinguishable concepts. The team leaders' perception of team performance had a medium correlation with TWQ. In contrast, no effect of TWQ on team performance was found when product owners rated team performance. The effect of TWQ on team performance was higher for product quality (in particular regarding team members and team leaders) than for project quality.

Despite the emphasis on TWQ in the agile community, in the traditional and the agile surveys alike, both the evaluation of TWQ itself and its effect on team performance and team members´ success were similar. However, the agile survey showed lower agreement among the raters regarding evaluation of team performance than was the case in the traditional survey.

An implication of this survey is that the quality of teamwork is a major factor in improving team performance, especially for improving the quality of the team's product. Note that when trying to optimize team performance, one needs consensus of whose view of team performance should be considered. For the future, we recommend that more research efforts be made to validate the TWQ construct and to advance the measurement of team performance.

Acknowledgments

We thank Geir Amsjø, Nils Brede Moe, Viktoria Stray, and the anonymous reviewers for valuable comments. This work was partly funded by the Research Council of Norway through the projects TeamIT (Grant 193236/I40) and Agile 2.0 (Grant 236759).

References

Anderson, J. C., Gerbing, D. W., 1988. Structural equation modeling in practice: A review and recommended two-step approach. Psychological Bulletin Vol. 103, No. 3, pp. 411–423. DOI:10.1037/0033-2909.103.3.411.

Bergersen, G. R., Hannay, J. E., Sjøberg, D. I. K., 2011. Inferring skill from tests of programming performance: Combining time and quality. Empirical Software Engineering and Measurement (ESEM), pp. 305–314. DOI: 10.1109/ESEM.2011.39

Borsboom, D., 2008. Latent variable theory. Measurement: Interdisciplinary Research and Perspectives. Vol. 6, No. 1–2, pp. 25–32. DOI:10.1080/15366360802035497.

Brannick, M. T., Prince, A., Prince, C., Salas, E., 1995. The measurement of team process. Human Factors, Vol. 37, No. 3, pp. 641–651. DOI:10.1518/001872095779049372.

Chow, T., Cao, D. B., 2008. A survey study of critical success factors in agile software projects. Journal of Systems and Software, Vol. 81, No. 6 pp. 961–971. DOI:10.1016/j.jss.2007.08.020.

Cohen, S., Bailey, D., 1997. What makes team work: Group effectiveness. Research from the shop floor to the executive suite. Journal of Management, Vol. 23, No. 3, pp. 239–290. DOI:10.1177/014920639702300303.

Denison, D.R., Hart, S.L., Kahn, J.A., 1996. From chimneys to cross-functional teams: Developing and validating a diagnostic model. The academy of Management Journal, Vol. 39, No. 4, pp. 1005–1023. DOI: 10.2307/256721.

Dickinson, T.L., McIntyre, R.M., 1997. A conceptual framework of teamwork measurement. Team performance assessment and measurement: Theory, methods, and applications, Psychology Press, pp. 19–43.

Dingsøyr, T., Lindsjørn, Y., 2013. Team performance in agile development teams: Findings from 18 focus groups, XP2013, Lecture Notes in Business Information Processing. Vol. 149, pp. 46–60. DOI: 10.1007/978-3-642-38314-4_4.

Dingsøyr, T., Nerur, S., Balijepally, V., Moe, N. B., 2012. A decade of agile methodologies: Towards explaining agile software development. Journal of Systems and Software, Vol. 85, No. 6, pp. 1213–1221. DOI:10.106/j.jss.2012.02.033.

Dybå, T., Dingsøyr, T., 2008. Empirical studies of agile software development: A systematic review. Information and Software Technology, Vol. 50, No. 9–10, pp. 833–859. DOI:10.1016/j.infsof.2008.01.006.

Faraj, S., Sproull, L., 2000. Coordinating expertise in software development teams. Management Science, Vol. 46, No. 12, pp. 1554–1568.

Gladstein, D.L., 1984. Groups in context: A model of task group effectiveness. Administrative Science Quarterly, Vol. 29, No. 4, pp. 499–517.

Hackman, J. R., 1987. The design of work teams. Handbook of organizational behavior. Prentice-Hall, Englewood, Cliffs, NJ, pp. 315–342.

Hoegl, M., Gemuenden, H.G., 2001. Teamwork quality and the success of innovative projects: A theoretical concept and empirical evidence. Organization Science, Vol. 12, No. 4, pp. 435–449. DOI:10.1287/orsc.12.4.435.10635.

Hoegl, M., Parboteeah, K.P., Gemuenden, H.G., 2003. When teamwork really matters: task innovativeness as a moderator of the teamwork-performance relationship in software development projects. Journal of Engineering and Technology Management, Vol. 20, No. 4, pp. 281–302. DOI:10.1016/j.jengtecman.2003.08.001.

Hoegl, M., Weinkauf, K., Gemuenden, H.G., 2004. Interteam coordination, project commitment, and teamwork in multiteam R&D projects: A longitudinal study. Organization Science, Vol. 15, No. 1, pp. 38–55. DOI:10.1287/orsc.1030.0053.

Janz, B.D., 1999. Self-directed teams in IS: Correlates for improved systems development work outcomes. Information & Management, Vol. 35, No. 3, pp. 171–192. DOI:10.1016/S0378-7206(98)00088-3.

Lakhpanel, B., 1993. Understanding the factors influencing the performance of software development groups: An exploratory group-level analysis. Information and Software Technology, Vol. 35, No. 8, pp. 468–473. DOI:10.1016/0950-5849(93)90044-4.

Li, Y, Chang, K.C., Chen, H.G., Jiang J. J., 2010. Software development team flexibility antecedents. Journal of Systems and Software, Vol. 83, No. 10, pp. 1726–1734. DOI:10.1016/j.jss.2010.04.077.

Lindsay, R.M., Ehrenberg, A.S.C., 1993. The design of replicated studies. The American Statistican, Vol. 47, No. 3, pp. 217–228. DOI:10.1080/00031305.1993.1047598.

MacKenzie, S.B., Podsakoff, P.M., 2012. Common method bias in marketing: causes, mechanisms, and procedural remedies. Journal of Retailing, Vol. 88, No. 4, pp. 542–555. DOI: 10.1016/j.jretai.2012.08.001.

Malone, T.W., Crowston, K., 1994. The interdisciplinary study of coordination. ACM Computing Surveys, Vol. 26, No. 1 pp. 87–119. DOI:10.1145/174666.174668.

Maruping, L. M., Zhang, X. J., Venkatesh, V., 2009. Role of collective ownership and coding standards in coordinating expertise in software project teams, European Journal of Information Systems, Vol. 18, pp. 355–371. DOI: 10.1057/ejis.2009.24.

Mathieu, J., Maynard, M.T., Rapp, T., Gilson, L., 2008. Team effectiveness 1997–2007: a review of recent advancements and a glimpse into the future. Journal of Management, Vol. 34, No. 3, pp. 410–476. DOI:101177/0149206308316061.

McGrath, J.E., 1964. Social Psychology: A Brief Introduction. New York: Holt, Rinehart and Winston.

Mudrack, P.E., 1989. Defining group cohesiveness. A legacy of confusion. Small Group Research, Vol. 20, No. 1, pp. 37–49. DOI:10.1177/104649648902000103.

Moe, N. B., Dingsøyr, T., Dybå, T., 2010. A teamwork model for understanding an agile team: A case study of a Scrum project, Information and Software Technology, Vol. 52, pp. 480–491.

Mullen, B., Copper, C., 1994. The relation between group cohesiveness and performance: An integration, Psychological Bulletin, Vol. 115, No. 2, pp. 210–227.

Nerur, S., Mahapatra, R.K., Mangalaraj, G, 2005. Challenges of migrating to agile methodologies. Communications of the ACM, Vol. 48, No. 2.

Nunnally, J.C., Bernstein, I.H., 1994. Psychometric theory (3rd ed.), New York: McGraw-Hill.

Pikkarainen, M., Haikara, J., Salo, O., Abrahamsson, P., Still, J., 2008. The impact of agile practices on communication in software development. Empirical Software Engineering, vol. 13, pp. 303–337.

Pinto, M. B., Pinto J. K., 1990. Project team communication and cross functional cooperation in new program development. Journal of Product Innovation Management, Vol. 7, No. 3, pp. 200–212. DOI:10.1016/0737-6782(90)90004-X.

Pinto, M. B., Pinto J. K., Prescott J.E., 1993. Antecedents and consequences of project team cross-functional cooperation. Management Science, Vol. 39, No. 10, pp. 1281–1297. DOI:10.1287/mnsc.39.10.1281.

R Core Team, 2015. R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL: http://www.R-project.org/.

Rasmussen, T., Jeppesen, H., 2006. Teamwork and associated psychological factors: A review. Work & Stress, Vol. 20, No. 2, pp. 105–128. DOI:10.1080/02678370600920262.

Rosseel, Y., 2012. lavaan: An R package for structural equation modeling. Journal of Statistical Software Vol. 48 No. 2, pp. 1–36.

Ryan, S., O´Conner, V., 2009. Development of a team measure for tacit knowledge in software development teams. Journal of Systems and Software, Vol. 82, No. 2, pp. 229–240. DOI:10.1016/j.jss.2008.05.037.

Salas, E., Sims, E.S., Burke, C.S., 2005. Is there a "big five" in teamwork? Small Group Research, Vol. 36, No. 5, pp. 555–599. DOI:10.1177/1046496405277134.

Salas, E., Stagl, K.C., Burke, C.S., Goodwin, G.F., 2007. Fostering team effectiveness in organizations: toward an integrative theoretical framework. Nebraska Symposium on Motivation, Vol. 52, pp. 185–243.

Schwaber, K., Beedle, M., 2001. Agile software development with Scrum. Prentice Hall, New Jersey.

Seers, A., Petty, M.M., Cashman, J. F., 1995. Team-member exchange under team and traditional management: A naturally occurring quasi experiment. Group & Organization Management, Vol. 20, No. 1, pp. 18–38. DOI:10.1177/1059601195201003.

Shadish, W.R., Cook, T.D., Campbell, D.T., 2002. Experimental and quasi-experimental designs for generalized causal inference, Boston: Houghton Mifflin.

Sharp H., Robinson H., 2010. Three "C"s of Agile practice: Collaboration, co-ordination and communication, XP2010, pp. 61–85. DOI:10.1007/978-3-642-12575-1.

Stray, V., Sjøberg, D.I.K., Dybå, T., 2016. The daily stand-up meeting: A grounded theory study. Journal of Systems and Software, Vol. 114, pp. 101–124. DOI: 10.1016/j.jss.2016.01.004.

Strode, D.E., Huff, S.L., Hope, B., Link, S., 2012. Coordination in co-located agile software development teams. Journal of Systems and Software, Vol. 85, No. 6 pp. 1222–1238. DOI:10.1016/j.jss.2012.02.017.

Sundstrom, E., De Meuse, K.P., Futrell, D., 1990. Work teams: Applications and effectiveness. American Psychologist, Vol. 45, No. 2, pp 120–133.

Tannenbaum, S.I., Beard, R.L., Salas, E., 1992. Team building and its influence on team effectiveness: An examination of conceptual and empirical developments. Issues, Theory, and Research in Industrial/Organizational Psychology. Advances in Psychology, Vol. 82, pp. 117–153. DOI:10.1016/S0166-4115(08)62601-1.

Tjosvold, D., 1998. Cooperative and competitive goal approach to conflict: Accomplishments and challenges. Applied Psychology, Vol. 47, No. 3, pp. 285–313. DOI:10.1111/j.1464–0597.1998.tb00025.x.

Vinekar, V., Slinkman, C., W., Nerur, S., 2006. Can agile and traditional systems development approaches coexist? An ambidextrous view. Information Systems Management, Vol. 23, No.3, pp. 31–42. DOI:10.1201/1078.10580530/46108.23.3.20060601/93705.4

Vinod, V., Dhanalakshmi, J., Sahadev, S., 2009. Software team skills on software product quality. Asian Journal of Information Technology, Vol. 8, No. 1, pp. 8–13.

Wothke, W., 1993. Nonpositive definite matrices in structural modeling. In testing structural equation models, Bollen, K.A., and Long, J. S. (Eds.), Newbury park:Sage , pp. 256–293.

Appendix A – Questionnaire

Table 7 - Items in Questionnaire

Construct (no of Items)	Item no., Statement
Teamwork Quality (38) Communication (10)	1. There is frequent communication within the team 2. The team members communicate often in spontaneous meetings, phone conversations, etc. 3. The team members communicate mostly directly and personally with each other 4. There are mediators through whom much communication is conducted (*) 5. Relevant ideas and information relating to the teamwork is shared openly by all team members 6. Important information is kept away from other team members in certain situations (*) 7. In the team there are conflicts regarding the openness of the information flow (*) 8. The team members are happy with the timeliness in which they receive information from other team members 9. The team members are happy with the precision of the information they receive from other team members 10. The team members are happy with the usefulness of the information they receive from other team members
Coordination (4)	11. The work done on subtasks within the team is closely harmonized 12. There are clear and fully comprehended goals for subtasks within our team 13. The goals for subtasks are accepted by all team members 14. There are conflicting interests in our team regarding subtasks/subgoals (*)
Mutual Support (7)	15. The team members help and support each other as best they can 16. If conflicts come up, they are easily and quickly resolved 17. Discussions and controversies are conducted constructively 18. Suggestions and contributions of team members are respected 19. Suggestions and contributions of team members are discussed and further developed 20. The team is able to reach consensus regarding important issues 21. The team cooperate well
Effort (4)	22. Every team member fully pushes the teamwork 23. Every team member makes the teamwork their highest priority 24. The team put(s) much effort into the teamwork 25. There are conflicts regarding the effort that team members put into the teamwork (*)
Cohesion (10)	26. The teamwork is important to the team 27. It is important to team members to be part of the team 28. The team does not see anything special in this teamwork (*) 29. The team members are strongly attached to the team 30. All team members are fully integrated in the team 31. There were many personal conflicts in the team (*) 32. There is mutual sympathy between the members of the team 33. The team sticks together 34. The members of the team feel proud to be part of the team 35. Every team member feels responsible for maintaining and protecting the team
Balance of member Contribution (3)	36. The team recognizes the specific characteristics (strengths and weaknesses) of the individual team members 37. The team members contribute to the achievement of the team's goals in accordance with their specific potential 38. Imbalance of member contributions cause conflicts in our team (*)
Team members' success (8) Work Satisfaction (4)	39. So far, the team can be pleased with its work 40. The team members gain from the collaborative teamwork 41. The team members will like to do this type of collaborative work again 42. We are able to acquire important know-how through this teamwork
Learning (4)	43. We consider this teamwork as a technical success 44. The team learn important lessons from this teamwork 45. Teamwork promotes one personally 46. Teamwork promotes one professionally
Team Performance (15) Effectiveness (10)	47. Going by the results, this teamwork can be regarded as successful 48. All demands of the customers are satisfied 49. From the company's perspective, all team goals are achieved 50. The performance of the team advances our image to the customer 51. The teamwork result is of high quality 52. The customer is satisfied with the quality of the teamwork result 53. The team is satisfied with the teamwork result 54. The product produced in the team, requires little rework 55. The product proves to be stable in operation 56. The product proves to be robust in operation
Efficiency (5)	57. The company is satisfied with how the teamwork progresses 58. Overall, the team works in a cost-efficient way 59. Overall, the team works in a time-efficient way 60. The team is within schedule 61. The team is within budget

(*) = reverse coded item

Bio Yngve Lindsjørn Yngve Lindsjørn received his MSc degree in computer science from the University of Oslo in 1987. He worked 10 years as a researcher at Norsk Regnesentral from 1987 to 1997. He has 13 years of industry experience as project manager and as a manager (CEO) of a software company within the IT industry. From 2009 to 2014 he was project manager for a research project investigating the effect of teamwork within and across software development teams. From 2010 he has worked at the Department of Informatics, University of Oslo. His research interests include software development methods and socio-technical factors influencing software project success.

Bio Dag I.K. Sjøberg Dag I.K. Sjøberg received the MSc degree in computer science from the University of Oslo in 1987 and the PhD degree in computing science from the University of Glasgow in 1993. He has five years of industry experience as a developer and group leader. In 2001, he formed the Software Engineering Department at Simula Research Laboratory and was its leader until 2008, when it was number 1 in a ranking by the Journal of Systems and Software. Since 1999 he has been a full professor of software engineering at the University of Oslo. Sjøberg was an Associate Editor of *Empirical Software Engineering from 2002 to 2009 and an* Associate Editor of *IEEE Transactions on Software Engineering* from 2010 to 2014. His main research interests are the software life cycle, including agile and lean development processes, skill assessment, and empirical research methods in software engineering. Sjøberg is a member of IEEE and ACM.

Bio Torgeir Dingsøyr Torgeir Dingsøyr focuses on software process improvement and knowledge management as chief scientist at the SINTEF research foundation. In particular, he has studied agile software development through a number of case studies, co-authored the systematic review of empirical studies, co-edited the book Agile Software Development: Current Research and Future Directions, and co-edited the special issue on Agile Methods in the Journal of Systems and Software. He wrote his doctoral thesis on Knowledge Management in Medium-Sized Software Consulting Companies at the Department of Computer and Information Science, Norwegian University of Science and Technology, where he is an adjunct professor.

Bio Tore Dybå Tore Dybå received the MSc degree in electrical engineering and computer science from the Norwegian Institute of Technology in 1986, and the Dr. Ing. degree in computer and information science from the Norwegian University of Science and Technology in 2001. He has eight years of industry experience from Norway and Saudi Arabia. He is a chief scientist and research manager at SINTEF ICT and a professor at the University of Oslo. For the period 2001-2012, he was ranked as the top scholar worldwide in agile software development by the Journal of Systems and Software. He was on the editorial board of Empirical Software Engineering from 2007 to 2013. Since 2011 he has been editor of the Voice of Evidence column in IEEE Software, and since 2013 he has been on the editorial board of Information and Software Technology. His research interests include evidence-based software engineering, software process improvement, and agile software development.

Bio Gunnar R. Bergersen Gunnar Rye Bergersen received the MSc degree in computer science (2001) and the PhD degree in software engineering (2015) from the University of Oslo. He has worked as a software developer, project leader, and manager for over 15 years and is currently the CEO of Technebies, a startup based on his work at Simula Research

Laboratory. Bergersen is employed part time at the University of Oslo, first as a researcher since 2011 and then as an associate professor since 2016. His research interests concern software development, particularly the acquisition and measurement of programming skills.