Investigating Development Team Performance: Research Methods

Torgeir Dingsøyr, Tor Erlend Fægri, Tore Dybå, Børge Haugset, and Yngve Lindsjørn

This Web extra complements the July/Aug. 2016 *IEEE Software* Voice of Evidence article "Team Performance in Software Development: Research Results versus Agile Principles," which summarized results from a systematic review of studies of software development teams' performance. This material details our research methods and shows connections between the studies and our findings.

Team Performance

A central concept in studies of teams is their performance.[1] Some studies refer to team performance as the process of conducting teamwork and to team effectiveness as the teamwork's outcome.[2] Some studies also consider learning to be a team-performance indicator.

For our purposes, performance is teamwork's outcome, such as meeting project goals, budget, and schedule; the quality of the software developed; and development effectiveness and efficiency. Our definition of performance also includes team members' motivation to work together, often measured by job satisfaction.

Conducting the Review

A systematic review is characterized by a defined research question, identification of inclusion and exclusion criteria, a search for relevant studies, critical appraisal, data extraction, and synthesis of findings.[3] Figure 1 shows our study selection process.

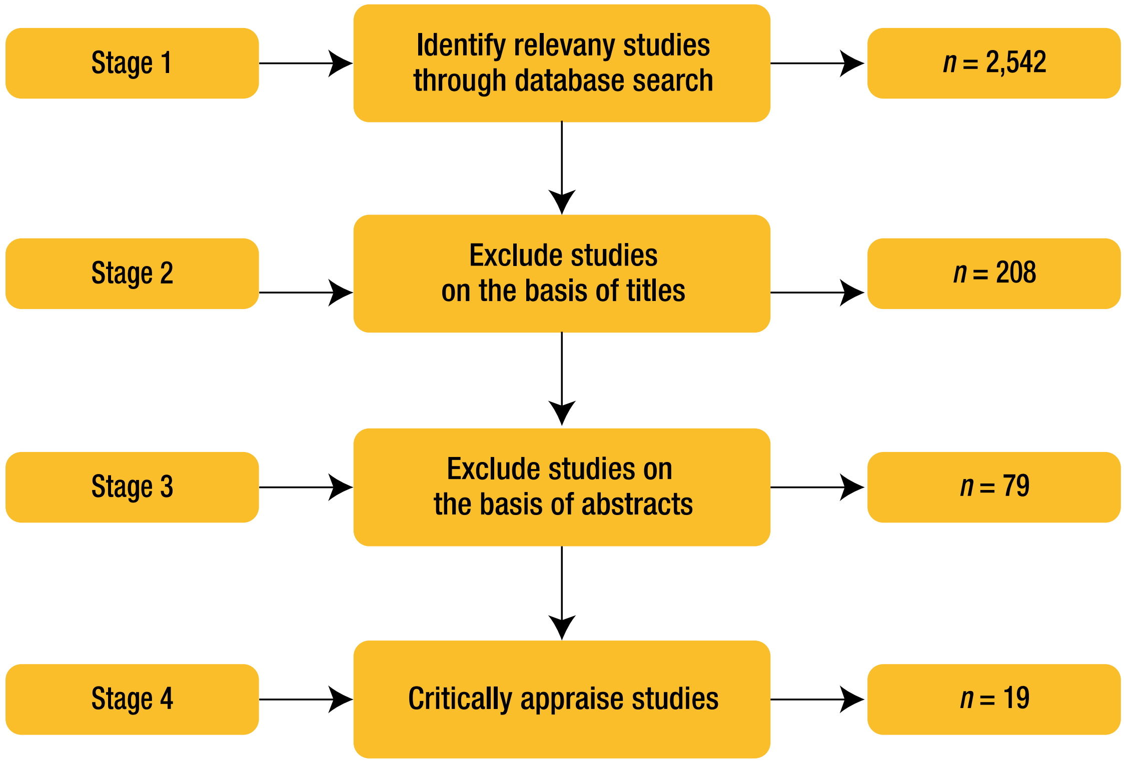

Figure 1. The stages of the study selection process.

Research Question

Our research question was, *what main factors influence software development teams' performance?*

Inclusion and Exclusion Criteria

We included empirical survey studies, published in scholarly journals, of team performance on colocated professional software development projects. We excluded studies of student teams, of specific development practices, and of distributed and global development teams. We examined surveys because they're conducted on industry participants and show causal relationships, and because this limited the number of studies we had to consider. We focused on colocated teams so that we would address only team-related factors, not temporal, geographical, or sociocultural factors.

Searching for Relevant Studies

In October 2011, we searched the ISI Web of Knowledge and the Scopus abstract and citation database of peer-reviewed literature, using this search string:

```
Title = (Team OR group OR teamwork) AND Topic = Software AND Document Type = (Article OR review)
```

Critical Appraisal

The 2,542 citations retrieved during the search process's first stage were imported to a reference-management package and then exported to a spreadsheet, at which point we recorded further decisions on the exclusion of studies.

During the second stage, two of us excluded studies whose title showed they were clearly unrelated to software development teamwork. During the third stage, we read full abstracts and excluded irrelevant studies, many because they were conducted on student or dispersed (not colocated) teams. All text was read by two of us independently and by another of us if the first two disagreed. During the fourth stage, we excluded studies that weren't surveys and that didn't have a performance-related research question or hypothesis. This left the 19 studies we worked with (see Table 1).

Table 1. Primary studies on team performance.

Study	Topic	Reference
S1	Managerial- and team-level control in design teams	J.C. Henderson and S. Lee, "Managing I/S Design Teams: A Control Theories Perspective," *Management Science*, vol. 38, no. 6, 1992, pp. 757–777.
S2	The relationship between design methods and goal orientation on one hand, and team effectiveness on the other	S. Sonnentag et al., "Use of Design Methods, Team Leaders' Goal Orientation, and Team Effectiveness: A Follow-Up Study in Software Development Projects," *Int'l J. Human-Computer Interaction*, vol. 9, no. 4, 1997, pp. 443–454.
S3	How teamwork quality affects performance	M. Hoegl and H.G. Gemuenden, "Teamwork Quality and the Success of Innovative Projects: A Theoretical Concept and Empirical Evidence," *Organization Science*, vol. 12, no. 4, 2001, pp. 435–449.
S4	Software development team flexibility antecedents	Y.Z. Li et al., "Software Development Team Flexibility Antecedents," *J. Systems and Software*, vol. 83, no. 10, 2010, pp. 1726–1734.
S5	How team dynamics and organizational support affect ICT (information and communications technology) project success	R. Gelbard and A. Carmeli, "The Interactive Effect of Team Dynamics and Organizational Support on ICT Project Success," *Int'l J. Project Management*, vol. 27, no. 5, 2009, pp. 464–470.
S6	Coordinating expertise in teams	S. Faraj and L. Sproull, "Coordinating Expertise in Software Development Teams," *Management Science*, vol. 46, no. 12, 2000, pp. 1554–1568.
S7	How collective ownership and coding standards affect team performance	L.M. Maruping, X.J. Zhang, and V. Venkatesh, "Role of Collective Ownership and Coding Standards in Coordinating Expertise in Software Project Teams," *European J. Information Systems*, vol. 18, no. 4, 2009, pp. 355–371.
S8	Behavioral and technical factors in team performance	P.J. Guinan, J.G. Cooprider, and S. Faraj, "Enabling Software Development Team Performance during Requirements Definition: A Behavioral versus Technical Approach," *Information Systems Research*, vol. 9, no. 2, 1998, pp. 101–125.
S9	Influential characteristics of information-systems development-team performance	Y.B. Lu et al., "What Affects Information Systems Development Team Performance? An Exploratory Study from the Perspective of Combined Socio-technical Theory and Coordination Theory," *Computers in Human Behavior*, vol. 27, no. 2, 2011, pp. 811–822.
S10	How team cohesiveness, experience, and capabilities influence team performance	B. Lakhanpal, "Understanding the Factors Influencing the Performance of Software Development Groups: An Exploratory Group-Level Analysis," *Information and Software Technology*, vol. 35, no. 8, 1993, pp. 468–473.
S11	Conflicts in software development teams	D.H. Gobeli, H.F. Koenig, and I. Bechinger, "Managing Conflict in Software Development Teams: A Multilevel Analysis," *J. Product Innovation Management*, vol. 15, no. 5, 1998, pp. 423–435.

S12	How team diversity affects performance	T.P. Liang et al., "Effect of Team Diversity on Software Project Performance," *Industrial Management & Data Systems*, vol. 107, no. 5, 2007, pp. 636–653.
S13	How intragroup conflict affects team performance	S. Sawyer, "Effects of Intra-group Conflict on Packaged Software Development Team Performance," *Information Systems J.*, vol. 11, no. 2, 2001, pp. 155–178.
S14	Team memory in software development projects	H. Keskin, "Antecedents and Consequences of Team Memory in Software Development Projects," *Information and Management*, vol. 46, no. 7, 2009, pp. 388–396.
S15	Knowledge integration in information-systems development teams	B.D. Janz and P. Prasarnphanich, "Freedom to Cooperate: Gaining Clarity into Knowledge Integration in Information Systems Development Teams," *IEEE Trans. Eng. Management*, vol. 56, no. 4, 2009, pp. 621–635.
S16	Team reflexivity in innovative teams	M. Hoegl and K.P. Parboteeah, "Team Reflexivity in Innovative Projects," *R&D Management*, vol. 36, no. 2, 2006, pp. 113–125.
S17	Developing a model that considers team task skills as a facilitator for the more specific application-domain and development-methods skills	C.L. Chan, J.J. Jiang, and G. Klein, "Team Task Skills as a Facilitator for Application and Development Skills," *IEEE Trans. Eng. Management*, vol. 55, no. 3, 2008, pp. 434–441.
S18	Cooperation skills and personality for shared mental models	H.D. Yang, H.R. Kang, and R.M. Mason, "An Exploratory Study on Meta Skills in Software Development Teams: Antecedent Cooperation Skills and Personality for Shared Mental Models," *European J. Information Systems*, vol. 17, no. 1, 2008, pp. 47–61.
S19	Shared mental models and team effectiveness	H.R. Kang, H.D. Yang, and C. Rowley, "Factors in Team Effectiveness: Cognitive and Demographic Similarities of Software Development Team Members," *Human Relations*, vol. 59, no. 12, 2006, pp. 1681–1710.

We assessed these studies for quality based on eight criteria related to clarity of study aims, study design, and study outcome, adapting Tore Dybå and Torgeir Dingsøyr´s criteria for surveys[4] (see Tables 2 and 3). Three of us assessed each article, and we calculated the final quality scores by taking the mode of all three scores. For four of the 19 articles' total of 152 scores, no two assessors agreed, so we used the three assessments' mean value.

Table 2. Quality checklist.*			
Category	**Criterion no.**	**Criterion**	**Things to consider**
Aims	1	Do the authors clearly state their research aims?	Do the authors state research questions (for example, related to time to market, cost, product quality, process quality, developer productivity, and developer skills)? Do the authors state hypotheses and their underlying theories?
Design, data collection, and data analysis	2	Do the authors describe the sample and the target population?	Do the authors explain how they defined and selected the sample and target populations? Do the authors state to what degree the sample is representative of the target population? Do the authors explain why the sample they selected was the most appropriate for providing insight into the type of knowledge the study sought? Do the authors report the sample size and response rate?
	3	Do the authors describe the design of the questionnaire and define the measures?	Do the authors explain how they defined and selected items and measurement scales (for example, domain of concepts, multiple-item scales, units, and counting rules)? Do the authors use quality control methods to ensure the collected data's consistency, completeness, and accuracy? Do the authors perform reliability and validity analyses (for example, Cronbach's alpha, item-scale correlations, and factor analysis)? Do the authors append the questionnaire?
	4	Do the authors define the data analysis procedures?	Do authors justify their choice, describe the procedures, and provide references to descriptions of the procedures? Do the authors report significance levels and effect sizes? Do the authors perform analyses of possible nonresponse bias? Do the authors report or give references to raw data or descriptive statistics?
	5	Do the authors discuss potential researcher bias?	Did the authors develop some or all of the treatments? If yes, do the authors discuss the implications anywhere in the paper? (If the authors developed the treatments [or parts of them] without discussing the implications, the answer to question 5 is "not at all.") Do the authors critically examine their own role, potential bias, and influence during the formulation of research questions, sample recruitment, data collection, and analysis and selection of data for presentation?
	6	Do the authors discuss their study's limitations?	Do the authors discuss external validity with respect to subjects, materials, and tasks? If the study uses novel measures, do the authors discuss the construct validity of the measures?

			Do the authors discuss their findings' credibility?
Study outcome	7	Do the authors state the findings clearly?	Do the authors present results clearly? Do the authors present conclusions clearly? Are the conclusions warranted by the results, and do the authors clearly present the connections between the results and conclusions? Do the authors discuss their conclusions in relation to the original research questions? Do the authors explicitly discuss the study's limitations?
	8	Is there evidence that other researchers and practitioners can use the survey?	Do the authors discuss whether or how the findings could be transferred to other populations or consider other ways in which the research could be used? To what extent do the authors interpret results in the context of other studies, the existing body of knowledge, and theories?
* Each criterion question is answered on a 4-point scale. 3 points: the answer to all questions in the "Things to consider" column is "yes." 2 points: the answer to most (but not all) of the questions is "yes." 1 point: the answer to the minority of the questions is "yes." 0 points: the answer to none of the questions is "yes."			

Table 3. Ratings for the 18 studies.

Study	**Criterion**								**Total score**
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	
S1	3	3	3	3	1	1	2	2	18
S2	3	2	3	2	1	2	2	2	17
S3	3	3	3	3	1	2	3	3	21
S4	3	2	3	2	1	2	3	2	18
S5	3	2	2	2	0	2	3	2	16
S6	3	3	2	2	1	3	3	3	20
S7	3	3	3	3	1	2	3	3	21
S8	3	3	3	3	1	2	3	3	21
S9	3	2	3	2	1	2	3	2	18
S10	3	2	2	2	0	1	2	2	14
S11	2	3	2	3	0	2	2	2	16
S12	3	2	3	2	0	2	3	2	17
S13	3	3	3	3	1	3	3	3	22
S14	2	2	3	2	0	2	2	2	15
S15	3	3	3	3	1	3	3	3	22
S16	3	3	3	3	1	2	3	3	21
S17	3	2	3	3	0	2	3	2	18
S18	3	3	3	3	0	3	3	3	21
S19	3	3	3	2	0	3	3	3	20

The maximum possible score was 32, the average was 18.7, the highest was 22, and the lowest—which lacked discussions of researcher bias (criterion 5) and study limitations (criterion 6)—was 14. These were also the criteria with the lowest overall scores. The studies scored best on clear study aims (criterion 1), as well as description of questionnaire design and definitions of measures (criterion 3). We didn’t exclude any studies based on the quality assessment.

Data Extraction

From each study, we extracted and recorded in a spreadsheet research questions, hypotheses, survey context descriptions, and key information such as the number of teams examined, the number of respondents, team size, the way of measuring performance, and the test type and significance level.

Synthesis of Findings

We derived our propositions in the main article by grouping studies and identifying factors that influenced team performance, according to at least three studies. We then conducted a thematic synthesis of identified findings, based on confirmed and rejected performance-related hypotheses. Tables 4 through 8 show the connection between factors and the studies.

Table 4. Accepted hypotheses linking team coordination and team performance.

Hypothesis	Study no.
Adherence to coding standards improves software project technical quality.	S7
Anticipation capability improves the software development team’s flexibility.	S4
Conventional team factors (expertise, professional experience, administrative coordination, and software development methods) improve team performance.	S6
Generally, increases in both managerial control and team-member control improve information-system design (ISD) team performance.	S1
In general, increases in team-member outcome control improves information-system design team performance.	S1
Organizational support improves the positive relationship between team dynamics and project success.	S5
Team dynamics improves project success.	S5
Using design methods in the software development process improves team effectiveness.	S2

Table 5. Accepted hypotheses linking goal-oriented leadership and team performance.	
Hypothesis	**Study no.**
Mission clarity improves information-system development team performance.	S9
Generally, increases in both managerial control and team-member control improve information-system design team performance.	S1
Team leaders' goal orientation and the use of design methods improve team effectiveness.	S2
Visionary processes improve team performance.	S8

Table 6. Accepted hypotheses and research questions linking team cohesion and team performance.	
Hypothesis or research question	**Study no.**
Collective ownership improves software project technical quality.	S7
Group cohesiveness improves group performance.	S10
How does the conflict-management style relate to success?	S11
Teamwork quality (including cohesion) improves team performance on innovative projects.	S3
What effects do these factors have on packaged-software development-team performance?	S13
What factors most affect the level of intragroup conflict in packaged-software development teams?	S13
What relationships exist between software-team composition and performance?	S12
What relationship do conflict intensity and context have to project success?	S11

Table 7. Accepted hypotheses linking shared mental models and team performance.	
Hypothesis	**Study no.**
Increased memory dispersion improves procedural memory’s positive effect on speed to market.	S14
Mission clarity improves information-systems development-team performance.	S9
Declarative memory lowers development costs.	S14
Team members’ shared mental models have a better effect on software development team effectiveness than demographic (age, tenure, and gender) similarities.	S19
Team-members’ shared mental models improve software development team effectiveness.	S18

Table 8. Accepted hypotheses linking team learning and team performance.	
Hypothesis	**Study no.**
Conventional team factors (presence of expertise, professional experience, administrative coordination, and software development methods) improve team performance.	S6
Cooperative learning in an information systems development team improves performance.	S15
Cooperative learning in an information systems development team improves individual work satisfaction.	S15
Expertise coordination processes (recognizing where expertise is needed, knowing where expertise is located, and bringing expertise to bear) improve team performance.	S6
Expertise coordination processes improve team performance beyond traditional factors.	S6
Team reflexivity improves team effectiveness.	S16
Application domain skills’ impact on software development project management success increases for higher-level team task skills.	S17
Application domain skills’ impact on software development process success increases for higher-level team task skills.	S17
Development-method skills’ impact on software development project-management success increases for higher-level team task skills.	S17
Development-method skills’ impact on software development process success increases for higher-level team task skills.	S17
Software development groups' capabilities improve their performance.	S15
Software development groups' experience improves their performance.	S15

Study Limitation

We investigated only empirical surveys. However, the factors identified correspond to factors identified in general team performance models, based on broad literature reviews.[1,2]

Acknowledgments

We carried out this research as part of the projects TeamIT and Agile 2.0, supported by the Research Council of Norway through grants 193236 and 236759. Agile 2.0 is also supported by Kantega, Kongsberg Defence & Aerospace, Sopra Steria, and Sticos.

References

1. E. Salas, D.E. Sims, and S.C. Burke, "Is There a 'Big Five' in Teamwork?," *Small Group Research*, vol. 36, no. 5, 2005, pp. 555–599.
2. J. Mathieu et al., "Team Effectiveness 1997–2007: A Review of Recent Advancements and a Glimpse into the Future," *J. Management*, vol. 34, no. 3, 2008, pp. 410–476.
3. B.A. Kitchenham, *Guidelines for Performing Systematic Literature Reviews in Software Engineering, Version 2.3*, tech. report EBSE-2007-01, School of Computer Science and Mathematics, Keele Univ., and Dept. of Computer Science, Univ. of Durham, 2007.
4. T. Dybå and T. Dingsøyr, "Empirical Studies of Agile Software Development: A Systematic Review," *J. Information and Software Technology*, vol. 50, no. 9, 2008, pp. 833–859.