

Are Daily Stand-up Meetings Valuable? A Survey of Developers in Software Teams

Viktoria Stray¹(✉), Nils Brede Moe², and Gunnar R. Bergersen¹

¹ University of Oslo, Gaustadalléen 23B, 0374 Oslo, Norway
{stray, gunnab}@ifi.uio.no

² SINTEF, Strindveien 4, 7465 Trondheim, Norway
nils.b.moe@sintef.no

Abstract. The daily stand-up meeting is a widely used practice. However, what is more uncertain is how valuable the practice is to team members. We invited professional developers of a programming forum to a survey and obtained 221 responses. Results show that the daily stand-up meeting was used by 87% of those who employ agile methods. We found that even though the respondents on average were neutral towards the practice, the majority were either positive or negative. Junior developers were most positive and senior developers and members of large teams most negative. We argue that the value of the practice should be evaluated according to the team needs. Further, more work is needed to understand why senior developers do not perceive the meetings as valuable and how to apply the practice successfully in large teams.

Keywords: Daily meetings · Stand up meeting · Daily scrum · Communication · Coordination · Teamwork · Team size · Agile adoption · Agile practices

1 Introduction

Agile methods introduced the daily stand-up meeting (DSM) as a practice to improve communication in software development projects. In Scrum, the meeting is mandatory, time-boxed to 15 min and team members address: (1) what they have done the previous work day, (2) what they will do today and (3) what obstacles are preventing them from making progress [1]. Scrum recommends that the DSM should not be used for discussing solutions to obstacles raised. However, empirical studies have found that spending time in the short meeting on discussing and solving problems is valuable [2, 3].

DSMs are task oriented, generally unrecorded, and members gather to focus on a narrow organizational goal. According to Boden [4], such meetings can be characterized as informal. The practice gives team members an overview of what other team members are doing and is therefore an important mechanism to increase information sharing and team awareness [5]. The meeting is often conducted standing up to keep it brief and avoid lengthy discussions, hence the term “stand-up meeting”. The practice is also called “frequent short meetings” [6], “morning roll call” [7], and “daily Scrum meeting” [1]. The DSM is an important practice for agile teams because it helps the team in monitoring and managing its performance, which is important for the team to

self-manage [8]. Further, such meetings improve access to information that foster employee empowerment [9].

While DSM is a relatively straightforward practice to adopt, it is challenging to implement it successfully. Challenges include finding a suitable time of day, keeping the time limit and whether it should be held daily and standing up [10]. We have previously found DSMs to last 63% longer when team members sit rather than stand during the meeting [5]. Another challenge is members reporting their status to the team leader, resulting in team members not paying attention to each other [8].

Although the DSM is one of the most popular agile practices [11, 12] and the only daily team-based coordination mechanism, the practice has received little research attention. Further, because meeting satisfaction is part of overall job satisfaction [13], it is important to understand what makes this meeting valuable for team members. In a recent, qualitative study of thirteen teams (in Norway, Poland, UK and Malaysia) we found that the attitudes towards DSMs were slightly more positive than negative [5]. However, the level of satisfaction varied within the teams. Therefore, to understand how to implement DSM, it is important to explore satisfaction with the practice on an individual level. This leads to the following research question: “*What are the characteristics of developers perceiving the daily stand-up meeting to be valuable compared to those who do not?*”

Our work also answers a call for more empirical studies on the adoption rate of agile software development methods [14].

2 Method

The target population for this study was professional software developers. Accordingly, we posted the survey on Reddit, which is a social media website that allows scientists to recruit a targeted population [15]. We chose two programming-related subreddits (subforums) that provide news and discussions about computer programming (r/programming, 710 000 subscribers) and web development (r/webdev, 130 000 subscribers). The survey was administered using the Qualtrics software which prevents the survey to be completed more than once by the same respondent. Participation was voluntary. Further, no compensation was offered, which increase the quality of the data because the incentive to cheat is largely reduced [15]. The survey (available from <https://figshare.com/s/a10006dd8f5f26141511>) took about three minutes to complete.

We received 316 responses, of which 243 contained data that could be analyzed. Because we were interested in the opinions of software professionals currently working in teams, we removed students and those not working or not working in a team. In total, 221 responses were used for the reported results. The majority of responses were from the programming forum ($n = 165$). Nearly all the respondents were male (96.8%) with a mean age of 31 years ($n = 204$, $sd = 6.86$). Among the respondents who answered whether their team was distributed ($n = 168$), about two-thirds of the respondents (63.1%) reported working in co-located teams, whereas the remaining had team members distributed across sites (36.9%).

All Likert questions used a five-point scale. All nominal-scale questions were presented with a randomized order of categories because the order of response alternatives can influence results [16]. Some questions were not compulsory, which resulted in missing data for the reported variables. Analyses are reported using the R statistical software [17]. To err on the side of caution, we use two-tailed analysis and chose non-parametric statistical tests. The one-sample Wilcoxon test is used to check for statistically significant differences in distributions. When comparing frequencies between two dichotomous variables that contain count data (i.e., frequencies) we used Fisher's exact test which reports the odds ratio (OR) effect size.

3 Results

In our study, the average number of DSMs conducted per week was five, which suggests that it is a daily meeting. Table 1 shows descriptive statistics. We found no difference regarding the frequency of meetings when it comes to being part of a distributed team or not, or to team size. Among all the respondents, one-third reported to work in teams with two to five members, one-third in teams with six to eight members and one-third in teams with nine or more members. We found a difference of 52% points with an odds ratio of 12.3 for agile teams using DSMs over non-agile teams ($p < 0.001$, 95% confidence interval for OR: 5.4–29.5).

Table 1. Descriptive statistics

	Unit	n	Mean (M)	sd	median
Meetings	Frequency per day, DSMs included	166	1.8	1.2	2
Time in meetings	Hours per day; DSMs included	187	1.4	2.0	1
Time programming	Hours per day	196	6.2	2.3	7
Team size	Members including self	168	7.3	3.9	6
DSM valuable	Likert: Negative (1)–Positive (5)	149	3.0	1.2	3
Programming skill self	Likert: Novice (1)–Expert (5)	177	3.7	0.8	4
Programming skill peers	Likert: Novice (1)–Expert (5)	177	3.6	0.8	4

Overall, 70.6% report that they attend DSMs ($n = 221$). Those who attend and those who do not attend DSMs spend the same amount of hours in meetings (DSMs included) and report similar values for programming skills. However, those who attend DSMs spend almost one hour more each workday on programming ($p = 0.046$, attend: $M = 6.5$ h, $sd = 2.1$; not attend: $M = 5.6$ h, $sd = 2.7$). Further, those who attend DSMs work in larger teams ($p = 0.03$, attend: $M = 6.90$ members, $sd = 4.7$; $M = 7.44$ members, $sd = 3.52$); the median difference was 2 team members. Moreover, the practice is regarded as more valuable by those who attend DSMs than those who do not ($p = 0.002$, attend: $M = 3.1$, $n = 123$, not attend: $M = 2.3$, $n = 29$).

We now report on only those respondents who attend DSMs. While the mean perceived value by these respondents towards the practice was neutral (3.1), only 18.7% chose this middle category on the Likert scale. Most respondents were either positive (44.7%) or negative (36.6%). We coded responses of 4 and 5 as “positive”, responses of 1 and 2 as “negative”, and removed those who responded neutral to be able to better understand differences between these two groups. We found no relation between working in a co-located or distributed team and the perceived value of DSM. However, those positive were significantly younger ($p = 0.008$, positive: $M = 29.6$ years, $n = 49$; negative: $M = 33.5$ years, $n = 42$).

Figure 1 shows the characteristics of the respondents who attend DSMs according to whether they are positive (green, $n = 55$) or negative (red, $n = 45$) towards the meetings they attend. The left part of the figure shows that those positive and negative towards their DSMs spent about the same amount of time in meetings: 83 min for those positive versus 77 min for those negative. However, there was a significant difference in meeting frequency; those positive attended fewer meetings per day (DSMs included) than those negative. Those positive towards DSM report somewhat more time spent on programming per day (24 min) than those negative. Being positive towards DSMs was, to some extent, associated with working in smaller teams. As a post hoc analysis, we investigated differences in attitudes further and found that teams with 12 or more members were most strongly associated with negative attitudes towards DSMs.

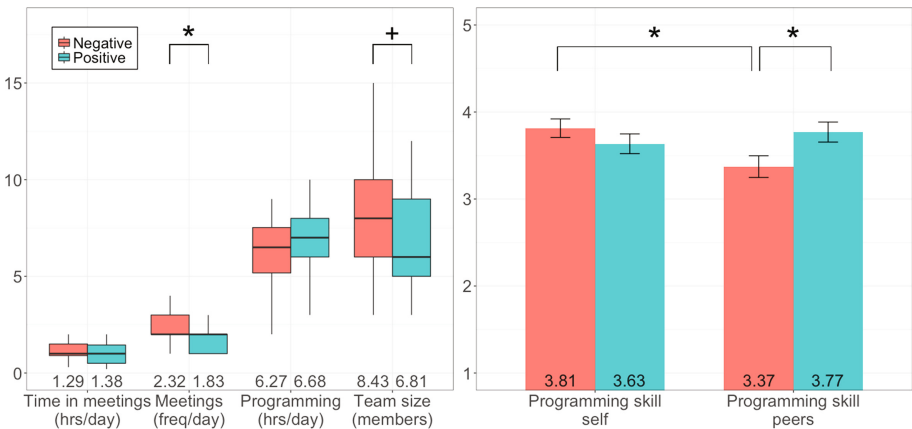


Fig. 1. Characteristics of those positive and negative towards their DSMs being valuable. Significant differences are shown at the top and means are shown at the bottom of the figure (as numbers). Outliers are omitted. Error bars represent the standard errors of the mean. + is $p < 0.10$ and * is $p < 0.05$ (two-sided). (Color figure online)

The right part of Fig. 1 shows a minor difference between how those positive and negative towards DSM rated their own programming skills. However, those positive towards DSM rated the programming skills of their peers as significantly higher compared to how the

negative rated their peers. Further, those negative also rated their own skills as significantly higher than that of their peers, whereas it was, to some extent, the opposite for those positive.

4 Discussion

The main explanation of the widespread use of DSM (70,6%) is the high adoption rate of agile development methods among our respondents. Table 2 shows that the agile adoption rate in our survey is higher than what was found by Rodríguez et al. [11]. Rodríguez et al. did not report the adoption rate of DSM but concluded that it was one of the most widely used practices. The last column in Table 2 shows the adoption rate of DSM in both agile and non-agile teams in our study. VersionOne [12] report the DSM to be the most employed agile practice with an adoption rate of 83%. VersionOne's sample mostly consisted of agile practitioners. In comparison, our DSM adoption rate among those using agile or agile in combination with Lean was 87.3%. Our results indicate that the practice has spread to companies not using agile methods because 35.4% of the respondents who work in non-agile teams also report using DSM. Thus, being agile implies that DSMs are used to a large extent which supports that DSM is a practice that distinguishes agile from non-agile teams [17].

Table 2. Usage rates of agile methods and DSM adoption according to development method

Development method	Agile adoption in our survey	Agile adoption in Rodríguez et al. [11]	DSM adoption in our survey
Agile and/or Lean	73.6%	57.8%	87.3%
<i>Only agile</i>	54.9%	33.6%	89.0%
<i>Agile and Lean</i>	18.7%	21.6%	82.4%
<i>Only Lean</i>	0.0%	2.7%	0.0%
Neither agile nor Lean	26.4%	42.2%	35.4%
Total	100.0%	100.0%	

For our research question, “What are the characteristics of developers perceiving the daily stand-up meeting to be valuable compared to those who do not?”, our results indicate that those positive towards DSM are more junior developers. This inference is supported by age, how they rate their own programming skills and their self-reported skills compared to the perceived skill of their peers. Those positive towards DSM also participate in fewer meetings than those negative. The same variables also indicate that those negative towards DSM are more senior developers. One explanation for why a senior developer regards DSM as less valuable is because seniors may already know what goes on in the team and does not get any new information in the meeting. The personal gain from the meeting is thus reduced. Moreover, being able to have quick problem-solving discussions in the DSM make developers perceive the DSM as more valuable [5]. Senior developers often work on more complex tasks, and it might be that high complexity problems are seldom discussed at the meeting because they require too

much time. It is more likely that the problems a junior developer encounter are more easily solved in a DSM.

A second explanation is that senior developers attend more meetings than junior developers. The DSM then becomes an additional daily interruption, which reduces the satisfaction with such meetings. Perceiving the meeting to have too high frequency negatively affects the attitude towards DSMs [5]. Moreover, meeting load affects employees well-being [18] so companies should be sensitive to the number of meetings the developers have to attend. While it has been claimed that DSMs eliminate the need for other meetings [1], we found no difference between hours spent in meetings for those who attend or do not attend DSMs.

In a self-managing team, the team goal should be more important than the individual goal, and then a developer should rate the DSM value depending on the team needs. One respondent commented: *“I think some people need the daily stand-up format. So even though I personally don’t feel like I need it, I feel it benefits us all to do it because of the different personalities.”* Because we do not know the perspective of the respondent we do not know if the respondents are considering the value from an individual or team perspective, or a mixture of the two views.

We found that larger teams are more likely to have DSMs. Paradoxically, the larger the team, the less is the satisfaction with DSM. Large teams using DSMs should therefore pay special attention to improving the quality of these meetings. In particular, developers were negative towards DSM when teams consisted of 12 or more team members. Previous research also found a negative correlation between the number of meeting participants and the attitude towards DSM [5].

The main limitation of this study concerns the representativeness. Although the distribution of self-reported programming skill in this study is nearly identical to our earlier study of programming skill of developers [19], the sample and target population may differ. For example, it is possible that only those who knew or had a strong (polarized) opinion of DSMs responded to the survey. This may bias results in favor of more respondents reporting using DSM and more variability in opinions than is actually present in the target population. Another potential concern is that we had subjects from two different programming forums, but the results we report still hold when analyzing the data from the two forums separately.

5 Conclusion and Future Work

The present study investigated the perceived value of daily stand-up meetings (DSMs) and reports the adoption rate of the practice. Among those who use agile methods, the majority conducts DSMs. Although it is a common practice, the perceived value of the meeting varies with junior developers being more positive and senior developers more negative towards the DSMs they attend. A possible explanation is that junior developers receive more relevant information and assistance in solving problems during the meeting. In contrast, senior developers often work with larger, more complex and independent tasks that are more difficult to share with team members on a daily basis. Agile teams are expected to be self-managed, and the need of the team should be more important than that of the individual. The value of the practice should, therefore,

be evaluated according to the team needs. Consequently, senior developers should be made more aware that DSMs are beneficial for the junior developers as well as the team as a whole. Another result was that developers in larger teams see the meeting as less valuable than developers in smaller teams. Because the work in large teams is often loosely coupled, the information shared during the meeting may be less relevant for the individuals. Consequently, large teams in particular need to invest resources in improving the practice to make it valuable.

Future work should investigate other criteria of the participants, such as role and domain. Because the perceived value of meetings affects job satisfaction, there is a need to understand why senior developers and large teams do not perceive the meeting as more valuable. The DSM is a widely adopted practice and is an important mechanism for information sharing and team awareness, thus, how to apply the practice successfully in large teams should also be studied.

Acknowledgments. We are grateful to the survey respondents and to the reviewers. This work was supported by the Smiglo project, which is partly funded by the Research Council of Norway under the grant 235359/O30.

References

1. Schwaber, K., Beedle, M.: *Agile Software Development with Scrum*. Prentice Hall, Upper Saddle River (2002)
2. Stray, V.G., Moe, N.B., Aurum, A.: Investigating daily team meetings in agile software projects. In: *The 38th EUROMICRO Conference on Software Engineering and Advanced Applications (SEAA 2012)*, Cesme, Turkey, 17 August 2012
3. Pikkariainen, M., Haikara, J., Salo, O., Abrahamsson, P., Still, J.: The impact of agile practices on communication in software development. *Empirical Softw. Eng.* **13**, 303–337 (2008)
4. Boden, D.: *The Business of Talk: Organizations in Action*. Polity Press, Cambridge (1994)
5. Stray, V., Sjøberg, D., Dybå, T.: The daily stand-up meeting: a grounded theory study. *J. Syst. Softw.* **114**, 101–124 (2016)
6. Rising, L.: Agile meetings. *STQE*, pp. 42–46 (2002)
7. Anderson, D.J.: *Agile Management for Software Engineering: Applying the Theory of Constraints for Business Results*. Prentice Hall, Upper Saddle River (2003)
8. Moe, N.B., Dingsøyr, T., Dybå, T.: A teamwork model for understanding an agile team: a case study of a Scrum project. *Inf. Softw. Technol.* **52**, 480–491 (2010)
9. Allen, J.A., Lehmann-Willenbrock, N., Sands, S.J.: Meetings as a positive boost? How and when meeting satisfaction impacts employee empowerment. *J. Bus. Res.* **69**, 1–8 (2016)
10. Stray, V.G., Lindsjorn, Y., Sjøberg, D.: Obstacles to efficient daily meetings in agile development projects: a case study. In: *The ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM 2013)*, Baltimore, USA, 13 September 2013
11. Rodríguez, P., Markkula, J., Oivo, M., Turula, K.: *Survey on Agile and Lean Usage in Finnish Software Industry*. ACM, New York (2012)
12. VersionOne: *VersionOne 10th Annual State of Agile Report*. <https://versionone.com/pdf/VersionOne-10th-Annual-State-of-Agile-Report.pdf>

13. Rogelberg, S.G., Allen, J.A., Shanock, L., Scott, C., Shuffler, M.: Employee satisfaction with meetings: a contemporary facet of job satisfaction. *Hum. Resour. Manag.* **49**, 149–172 (2010)
14. Stavru, S.: A critical examination of recent industrial surveys on agile method usage. *J. Syst. Softw.* **94**, 87–97 (2014)
15. Shatz, I.: Fast, free, and targeted: reddit as a source for recruiting participants online. *Soc. Sci. Comput. Rev.*, pp. 1–13 (2016)
16. Schwarz, N., Hippler, H.J.: Response alternatives: the impact of their choice and presentation order (1991)
17. Murphy, B., Bird, C., Zimmermann, T., Williams, L.: Have agile techniques been the silver bullet for software development at Microsoft? In: *The Proceedings of the 2013 ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM 2013)*, Baltimore, USA, 7 July 2013
18. Luong, A., Rogelberg, S.G.: Meetings and more meetings: the relationship between meeting load and the daily well-being of employees. *Group Dyn. Theor. Res. Pract.* **9**, 58–67 (2005)
19. Bergersen, G.R., Sjøberg, D., Dybå, T.: Construction and validation of an instrument for measuring programming skill. *IEEE Trans. Softw. Eng.* **40**, 1163–1184 (2014)

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.

