Empirical Software Engineering Group

ESEG

Centro de Informática U·F·P·E

Universidade Federal de Pernambuco

THE EFFECTIVENESS OF PAIR PROGRAMMING

This briefing reports evidence on the effectiveness of pair programming around quality duration and effort based on scientific evidence from a systematic review.

MAIN FINDINGS

- The findings presented in this briefing consider quality as the number of test cases passed or number of correct solutions of programming tasks; duration as the total time taken to complete all tasks considered (all solutions); and effort was reported as twice the duration of each individual in the pair.
- Studies present a small significant positive overall effect of pair programming on quality, a medium significant positive overall effect on duration, and a medium significant negative overall effect on effort.
- Evidence suggests that pair programming is faster than solo programming when programming task complexity is low and also yields code solutions of
- higher quality when task complexity is high.
- The higher quality for complex tasks comes at a price of considerably greater effort, while the reduced completion time for the simpler tasks comes at a price of noticeably lower quality.
- Research results show that the question of whether pair programming is better than solo programming depends on other factors, for example, the expertise of the programmers and on the complexity of the system and tasks to be solved.
- One of the most interesting observations is that the pairing up of individuals seems to elevate the junior pairs up to near senior pair performance. Thus, pair collaboration might compensate for juniors' lack of deep understanding, for example, by inducing an expertlike strategy.
- If you do not know the seniority or skill levels of your programmers, but do have a feeling for task complexity, then employ pair programming either when task complexity is low and time is of the essence, or when task complexity is high and correctness is important.
- When considering the moderating effect of programmer expertise, junior pairs had a small (5%) increase in duration and thus a large increase in effort (111%), and a 73% increase in correctness.
- Intermediate pairs had a 28% decrease in duration (43% increase in effort) and a negligible (4%) increase in correctness.
- Senior pairs had a 9% decrease in duration (83% increase in effort) and an 8% decrease in correctness.
- The juniors benefited from pair programming in terms of increased correctness, the intermediates in terms of decreased duration, while there were no overall benefits of pair programming for seniors.
- When considering the combined moderating effect of system complexity and programmer expertise on pair programming, there appears to be an interaction effect: Among the different treatment combinations, junior pairs assigned to the complex design had a remarkable 149% increase on correctness compared with individuals.
- Intermediates and seniors experienced an effect of pair programming on duration on the simpler design, with a 39% and 23% decrease, respectively.
- However, the cost of this shorter duration was a corresponding decrease in correct solutions by 29% and 13%, respectively.

Keywords:

Pair programming
Meta-analysis

Who is this briefing for?

Software engineers practitioners who want to make decisions about pair programming based on scientific evidence.

Where the findings come from?

All findings of this briefing were extracted from the systematic review conducted by Hannay et al.

What is systematic reviews?

cin.ufpe.br/eseg/systematic-reviews

What is included in this briefing?

The main findings of the original systematic review.

Evidence characteristics through a brief description about the original systematic review and the studies it analized.

What is not included in this briefing?

Additional information not presented in the original systematic review.

Detailed descriptions about the studies analised in the original systematic review.

For additional information about this briefing:

cin.ufpe.br/eseg/briefings

ORIGINAL SYSTEMATIC REVIEW REFERENCE

Jo E. Hannay, Tore Dybå, Erik Arisholm, Dag I.K. Sjøberg, The effectiveness of pair programming: A metaanalysis, Information and Software Technology, Volume 51, Issue 7, July 2009, Pages 11101122, ISSN 09505849, http://dx.doi.org/10.1016/j.infsof.2009.02.001.