Explaining Pair Programming Session Dynamics from Knowledge Gaps

Franz Zieris[1], Lutz Prechelt[2]

This is an extended abstract of the paper with the same title [ZP20a] which was presented at the 42nd International Conference on Software Engineering (2020).

Keywords: pair programming; qualitative analysis; grounded theory methodology

1 Background, Data, and Research Method

Pair programming (PP) has many purported benefits, including higher code quality, faster progress, and knowledge transfer between developers. Despite a lot of research on the effectiveness of PP, the question when it is useful or less useful remains unsettled: A meta-analysis found mere tendencies and a lot of between-study variance [Ha09]; a large controlled experiment could not determine consistent moderating effects of task complexity and developer expertise [Ar07]. Even though the feasible experimental setups tend to be highly unrealistic, there have been only few qualitative studies which looked at the actual PP process in industrial contexts (e. g., [Pl15]).

We follow Straussian Grounded Theory Methodology [SC90] to understand how pair programmers actually transfer knowledge. We analyze 26 recordings of industrial PP sessions from 9 companies which we selected in the manner of *theoretical sampling* from the *PP-ind* session repository [ZP20b]. For *open coding*, we build on our own prior work [ZP14; ZP16] that identified various phenomena related to within-session knowledge build-up and transfer. We validate our findings with practitioners from four companies.

2 Results

We identify two different types of required knowledge and explain how different constellations of knowledge gaps in these two respects lead to different session dynamics:

- Industrial pairs mostly deal with gaps in project-specific *system understanding,* or S knowledge. They address any differences in their respective system understanding first before building up new system understanding together.

[1] Freie Universität Berlin, Institut für Informatik, Takustr. 9, 14195 Berlin, Deutschland zieris@inf.fu-berlin.de
[2] Freie Universität Berlin, Institut für Informatik, Takustr. 9, 14195 Berlin, Deutschland prechelt@inf.fu-berlin.de

doi:10.18420/SE2021_48

- Differences in *general software development knowledge*, or G knowledge, hardly hamper the PP process. Rather, such a difference is an opportunity to transfer knowledge—which pairs only do after they dealt with their S knowledge gaps.
- Building up lacking G knowledge together in a PP session appears to be difficult.
- Pair constellations with *complementary knowledge* allow both partners to contribute S knowledge and G knowledge, respectively, which makes pair programming a particularly effective practice.
- Software developers may use our findings when forming pairs (e. g., by choosing a partner and/or amending the goal of the session such that differences in their knowledge levels play out favorably) or as a means of reflecting after a session (e. g., whether the *right* knowledge gaps were addressed or which were newly identified).

References

[Ar07] Arisholm, E.; Gallis, H.; Dybå, T.; Sjøberg, D. I.: Evaluating Pair Programming with Respect to System Complexity and Programmer Expertise. IEEE Transactions on Software Engineering 33/2, pp. 65–86, 2007.

[Ha09] Hannay, J. E.; Dybå, T.; Arisholm, E.; Sjøberg, D. I.: The effectiveness of pair programming: A meta-analysis. Information and Software Technology 51/7, pp. 1110–1122, 2009.

[Pl15] Plonka, L.; Sharp, H.; van der Linden, J.; Dittrich, Y.: Knowledge transfer in pair programming: An in-depth analysis. International Journal of Human-Computer Studies 73/, pp. 66–78, 2015.

[SC90] Strauss, A.; Corbin, J.: Basics of Qualitative Research. Grounded Theory Procedure and Techniques. Sage Publications, 1990, ISBN: 978-0803932500.

[ZP14] Zieris, F.; Prechelt, L.: On Knowledge Transfer Skill in Pair Programming. In: Proceedings of the 8th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement. ESEM '14, ACM, 2014.

[ZP16] Zieris, F.; Prechelt, L.: Observations on Knowledge Transfer of Professional Software Developers During Pair Programming. In: Proceedings of the 38th International Conference on Software Engineering Companion. ICSE '16 (SEIP), ACM, pp. 242–250, 2016.

[ZP20a] Zieris, F.; Prechelt, L.: Explaining Pair Programming Session Dynamics from Knowledge Gaps. In: Proceedings of the ACM/IEEE 42nd International Conference on Software Engineering. ICSE '20, ACM, Seoul, South Korea, pp. 421–432, 2020, ISBN: 9781450371216.

[ZP20b] Zieris, F.; Prechelt, L.: PP-ind: A Repository of Industrial Pair Programming Session Recordings, 2020, arXiv: 2002.03121v3 [cs.SE].