Eighth International Workshop on Large-Scale Agile Development

Agile at Scale: A Summary of the 8th International Workshop on Large-Scale Agile Development

Julian M. Bass and Abdallah Salameh

University of Salford, 43 Crescent, Salford M5 4WT, UK
j.bass@salford.ac.uk, a.salameh@edu.salford.ac.uk

Abstract. The Large-Scale Agile Development workshop explored the main research challenges in large-scale software development. We considered multi-site organisations with large-scale projects that include a large number of teams adopting agile methods. Such topics include inter-team coordination, knowledge sharing, large project organisation, agile transformation, agile teamwork quality, project models that facilitate several self-organising teams, and practices for scaling agile methods. We accepted five full research papers, which are included in this volume. The accepted papers report empirical research studies using surveys, observations and case studies. Also, an interactive online discussion session was conducted to compare the two approaches, SAFe and Spotify. The workshop participants, which were around a hundred people, joined this discussion to compare the two approaches and suggest some future research questions about the hybridisation of SAFe and Spotify. This workshop summary contributes as a current snapshot of research along with some results from an interactive discussion about SAFe and Spotify.

Keywords: Large-scale agile software development · Software engineering · SAFe · Spotify · Inter-team coordination · Agile transformation · Agile teamwork quality · Organisational change · Autonomous teams

1 Introduction

The goal of Large-Scale Agile Development workshop was to explore the main research challenges in conducting large-scale software development programmes using agile methods. How to apply agile methods to large-scale projects was identified as the "top burning research question" by practitioners at XP2010 and has since then attracted increasing interest among agile practitioners and researchers. The first of this workshop series was organised at XP2013. The workshop was planned to be conducted during the XP conference in Copenhagen in June 2020. However, the workshop was conducted online because of the Coronavirus disease (i.e., COVID-19) outbreak. Despite the pandemic, around a hundred attendees joined the workshop, which is more than double the attendance last year.

Agile methods are conventionally applied in small colocated software development teams. Since many organisations with small collocated teams have realised successful

implementation of software projects, agile methods became increasingly attractive for researchers and practitioners to apply agile software development to large-scale projects [5].

Large-scale projects are challenging because several teams need to work closely together to release a single software project [1, 4]. This workshop addressed research challenges in large-scale agile development and identified topics such as inter-team coordination, knowledge sharing, large project organisation, agile transformation, agile teamwork quality, project models that facilitate several self-organising teams, and practices for scaling agile methods.

2 Workshop Contributions

The workshop comprised speakers selected following submission of short papers, which were peer-reviewed by members of the program committee, and an interactive online discussion session about the differences between SAFe and Spotify.

2.1 Research Papers

For the 2020 workshop we had seven submissions, of which five were accepted as full research paper presentations. The first paper, "Transitioning from a First Generation to Second Generation Large-Scale Agile Development Method: Towards understanding Implications for Coordination" [2] reported preliminary insights on the coordination impact when an organisation moves from first (combined agile methods with traditional project management frameworks) to a second generation (using large-scale agile frameworks). The authors used four theories of coordination from different fields to analyse the findings and explain changes in coordination. They found that two of the theories are well suited to characterising the phases of the transition, providing answer to how coordination was done. While two other theories provide answers to why the coordination changes occurred and could help explaining the success of such transition.

The second paper, "Exploring the Product Owner Role within SAFe Implementation in a Multinational Enterprise" [10] compares previously identified activities of Product Owners outside the context of SAFe with activities of Product Owners in an examined SAFe implementation to improve the understanding of the Product Owner role within the context of SAFe. The authors found that the Product Owners role in the SAFe deviates from the previous understanding of the role outside the context of SAFe as the range of Product Owner activities are narrowed. They attribute the narrowed activities of Product Owners at SAFe to the introduction of a new form of management-driven top-down approach with the fragmentation of the roles.

The third paper, "A systematic approach to agile development in highly regulated environments" [8] describes an approach, called Levels of Done-Product Quality Risk (LoD-PQR), to align agile teams and ensure that teams meet regulatory requirements and product specific quality while retaining as much autonomy as possible. The authors claim that this approach enabled the autonomous teams, in the case study organisation, to realise efficiency by design and to share techniques on how to implement compliance

requirements. This in turn has streamlined the development processes in the case study organisation and led to a positive impact on process performance.

The fourth paper, "Evaluation of Agile Team Work Quality" [9] presents an approach to measure "agile Team Work Quality (aTWQ)", which enables teams in improving their agile mindset and practices without external assessments. This approach includes measurement indicators, which are based on extending the teamwork quality construct that are developed by previous research. The paper presents also how the case study organisation has made use of findings on teamwork to create a usable "Toolbox" for internal process improvement.

Finally, the fifth paper, "Operationalizing Agile Methods: Examining Coherence in Large-Scale Agile Transformations" [3] explores coherence in operationalising large-scale agile methods by presenting the results of a comparison between a successful and a failed large-scale agile transformation. Also, the paper describes challenges in understanding the rationale, differences, values, and roles associated with the methods to support successful large-scale agile transformation. In addition, the authors highlight factors that contribute to failed large-scale agile transformations.

2.2 SAFe vs Spotify - A Short Discussion

Both the SAFe framework and the Spotify model, which was initially introduced by Kniberg and Ivarsson [6, 7], are increasingly attracting agile practitioners in organisations of different context [4, 10–12]. The workshop participants were asked to provide similarities and differences between SAFe and Spotify by using an online Metro Retro board. Also, the participants were asked to provide possible future research directions for the hybrid of SAFe and Spotify.

The discussion did rise some aspects of importance when comparing SAFe to Spotify. These aspects are highlighted in Table 1.

The participants highlighted very few similarities between SAFe and Spotify compared to what differences they have provided. The mentioned similarities include the utilisation of communities of practice, falling back to the agile mindset to pinpoint what is needed, and trying to copy the agile approach. However, the participants highlighted many differences between SAFe and Spotify. For example, SAFe is a knowledge base (i.e., toolbox) of integrated principles, practices, and competencies for Lean, Agile, and DevOps that range from scrum teams to portfolio. According to some

Table 1. The rised aspects from the discussion – comparing SAFe to Spotify

	SAFe	Spotify
Process/culture	Knowledge base (toolbox)	Culture & interactions
Inter-team dependencies	High	Low
Innovation	Inhibitor	Enabler
Completeness	Detailed and complete	Abstract
Adoption	Easy	Complicated
Tailoring & improving	Hard	Easy

participants, SAFe implementation is complicated, includes unnecessary process, plan focused, bureaucratic, and dis-empowers team autonomy. Hence, SAFe is characterised as anti-agile. On the contrary, the authors of the Spotify model do not want to develop a big toolbox but rather to emphasise the need to create interactions between the teams through an Agile culture. This Agile culture focuses on enabling teams' autonomy by aligning the teams to each other to common product goals and objectives.

Participants considered SAFe suitable for projects and environments that have many dependencies among teams. Such dependencies, in turn, result in spending considerable resources to plan and coordinate work. SAFe is appropriate when the need for innovation among the developers is not a high priority because SAFe favours command and control, and teams are not highly empowered compared to the Spotify teams. The teams in the Spotify model have high autonomy to increase their creativity and innovation.

SAFe offers a complete course for its implementation and certification path to creating coaches of the framework. Whereas, the Spotify model is considered abstract and provides high-level details. Consequently, everything needed to implement SAFe is almost ready. Yet, it will require following its strict recommendations for the implementation, which are difficult to set up but not difficult to implement. On the other hand, adopting the Spotify model is perceived as an adventure where there are plenty of rooms for agile process tailoring, which in turn demands to have senior agile coaches to implement it. Such experienced agile coaches need to help autonomous squads to tailor their Agile processes, align all squads together and to project objectives, set up portfolio or program part, and define the Spotify communities (i.e., Squads, Chapters, Tribes, and Guilds [6]) and their content. Unlike the Spotify model, everything is already defined for SAFe, which makes it complicated to improve and adapt.

The participants were asked to provide possible future research directions for the hybrid of SAFe and Spotify. Interestingly, three participants in our discussion revealed encountering such hybrids in the industry nowadays. Also, the workshop participants provided few research directions for such hybrids, as follows:

- Why should we have a hybrid Agile development approach from SAFe and Spotify?
- How SAFe and the Spotify can be hybridised in the industry?
- How about a comparison of organically evolved approaches with prescriptive frameworks and models (SAFe vs Spotify)?
- What are prerequisites for inter-team coordination through practices such as "big room planning" in SAFe?

3 Programme Committee

Many thanks to the members of the programme committee many of whom have also contributed to previous workshops. The members' name are ordered alphabetically by last name, as follows:

- Finn Olav Bjørnson, Norwegian University of Science and Technology, Norway.
- Torgeir Dingsøyr, Norwegian University of Science and Technology, Norway.

- Denniz Donmez, Enabling Structures, Switzerland.
- Jutta Eckstein, IT communication, Germany.
- Peggy Gregory, UCLAN, UK.
- Tomas Gustavsson, Karlstad university, Sweden.
- Andrew Haxby, Competa IT BV, Netherlands.
- Aymeric Hemon, University of Nantes, France.
- Helena Holmström Olsson, University of Malmo, Sweden.
- Eric Knauss, Chalmers University, Sweden.
- Philippe Kruchten, University of British Columbia, Canada.
- Maarit Laanti, Nitor Delta, Finland.
- Carl Marnewick, University of Johannesburg, South Africa.
- Nils B. Moe, Sintef, Norway.
- Parastoo Mohagheghi, NAV, Norway.
- John Noll, University of Hertfordshire, UK.
- Maria Paasivaara, IT University of Copenhagen & Aalto University, Denmark & Finland.
- Yvan Petit, ESG UQAM, Canada.
- Jan Pries-Heje, Roskilde University, Denmark.
- Scarlet Rahy, University of Salford, UK.
- Knut H. Rolland, University of Oslo, Norway.
- Darja Smite, Blekinge Institute of Technology, Sweden.
- Christoph Stettina, Leiden University, Netherlands.
- Klaas-Jan Stol, Lero, UK.
- Viktoria Stray, University of Oslo, Norway.
- Ömer Uludag, Technical University of Munich, Germany.

Without the valuable support of these programme committee members the workshop would not have been possible. Thanks to Hubert Baumeister and Mansooreh Zahedi, the workshop co-chairs for XP 2020. Thanks also to Maria Paasivaara, the conference chair for XP 2020.

4 Conclusions

The Large-Scale Agile Development workshop successfully created an opportunity for researchers and practitioners to consider the latest trends in large-scale agile software development. The accepted papers in this proceeding and the interactive discussion session contribute as a snapshot of the start-of-the-art in the field of large-scale agile software development. The authors presented evidence of approaches being used to enable agile development in large-scale contexts. Yet, an incomplete adoption of some presented approaches was provided since the authors share preliminary findings of their conducted research. The workshop participants joined an interactive discussion to compare SAFe and Spotify and suggest future research questions about their hybridisation.

References

1. Bass, J.M.: Future trends in agile at scale: a summary of the 7th international workshop on large-scale agile development. In: Hoda, R. (eds) Agile Processes in Software Engineering and Extreme Programming – Workshops. XP 2019. LNBIP, vol. 364, pp. 75–80. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-30126-2_9
2. Bjørnson, F.O., Dingsøyr, T.: Transitioning from a first generation to second generation large-scale agile development method: towards understanding implications for coordination. In: Kruchten, P. (ed.) Agile Processes in Software Engineering and Extreme Programming – Workshops. Springer International Publishing (2020)
3. Carroll, N., Bjørnson, F.O., Dingsøyr, T., Rolland, K., Conboy, K.: Operationalizing agile methods: examining coherence in large-scale agile transformations. In: Kruchten, P. (ed.) Agile Processes in Software Engineering and Extreme Programming – Workshops. Springer International Publishing (2020)
4. Conboy, K., Carroll, N.: Implementing large-scale agile frameworks: challenges and recommendations. IEEE Softw. **36**(2), 44–50 (2019)
5. Dikert, K., Paasivaara, M., Lassenius, C.: Challenges and success factors for large-scale agile transformations: a systematic literature review. J. Syst. Softw. **119**, 87–108 (2016)
6. Kniberg, H.: Spotify squad framework, March, 2017. https://medium.com/project-management-learnings/
7. Kniberg, H., Ivarsson, A.: Scaling agile spotify with tribes, squads, chapters & guilds, October, 2012. https://blog.crisp.se/wp-content/uploads/2012/11/SpotifyScaling.pdf
8. Poth, A., Jacobsen, J., Riel, A.: Systematic agile development in regulated environments. In: Yilmaz, M., Niemann, J., Clarke, P., Messnarz, R. (eds.) EuroSPI 2020. CCIS, vol. 1251, pp. 191–202. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-56441-4_14
9. Poth, A., Kottke, M., Riel, A.: Evaluation of agile team work quality. In: Kruchten, P. (ed.) Agile Processes in Software Engineering and Extreme Programming – Workshops. Springer International Publishing (2020)
10. Remta, D., Doležel, M., Buchalcevová, A.: Exploring the product owner role within safe implementation in a multinational enterprise. In: Kruchten, P. (ed.) Agile Processes in Software Engineering and Extreme Programming – Workshops. Springer International Publishing (2020)
11. Salameh, A., Bass, J.M.: Spotify tailoring for promoting effectiveness in cross-functional autonomous squads. In: Hoda, R. (eds.) Agile Processes in Software Engineering and Extreme Programming – Workshops. XP 2019. LNBIP, vol. 364, pp. 20–28. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-30126-2_3
12. Salameh, A., Bass, J.M.: Heterogeneous tailoring approach using the spotify model. In: Proceedings of the Evaluation and Assessment in Software Engineering, EASE'20, pp. 293–298. Association for Computing Machinery, New York (2020)