

Agile tailoring with scaling agile frameworks in distributed large-scale settings

Rafael Camara¹, Marcelo Marinho², Hermano Moura¹

¹Informatic Center – Federal University of Pernambuco (UFPE)
Recife – PE – Brazil

²Department of Computer Science – Federal Rural University of Pernambuco (UFRPE)
Recife – PE – Brazil

Abstract. *Agile practices in large-scale distributed organizations are becoming a norm. Current research shows that large-scale projects are betting on agile scaling frameworks to coordinate better the several teams that compose large-scale organizations. However, the literature still needs further investigation on the scaling and tailoring of agile practices that distributed teams use in large-scale settings. Due to it, we aim to conduct a systematic literature review regarding the tailored agile practices used by distributed teams that use scaling agile frameworks in large-scale organizations. Further, we aim to conduct a case study on a large-scale Brazilian fintech company with more than 3000 employees. This company uses the Spotify model to organize its software product development.*

1. Research Problem

Distributed software development can be considered as an established paradigm for software development [Jalali and Wohlin 2012]. Many companies are adopting this model to achieve expense cuts, a 24/7 development cycle, and access high skilled talents [Rizvi et al. 2015]. However, such benefits come with some challenges related to cultural issues, timezone, coordination, linguistic barriers, and skill levels [Hossain et al. 2009].

Besides it, to overcome such challenges, companies are adopting agile, which by definition have been developed to help co-located teams. However, many studies demonstrate that agile methods can be helpful to mitigate DSD problems [Razzak et al. 2017]. Further, the agile presence in DSD is not the only main concern. Many organizations with DSD teams require tailoring agile practices to scale it in large-scale settings [Fitzgerald et al. 2013]. In response to the difficulty of carrying traditional agile methods into large-scale projects [Dybå and Dingsøy 2008], several scaling agile frameworks have emerged. However, similar to agile methods, adopting such scaling agile frameworks also requires the teams to tailor their practices [Salameh and Bass 2019].

Such scaling frameworks have many industrial reports regarding their successful adoptions in the industry, although academic studies evaluating them are still needed, especially in emergent markets like Brazil. Due to it and the emerging growth that those frameworks have in the industry, it becomes clear that it needs academic attention. It is necessary to go deeper into these agile scaling frameworks to understand better how distributed teams in large-scale contexts adopt the agile methods.

Further, the focus of our study will be regarding management and coordination activities in large-scale contexts. Since it gets complicated by the time the project and teams scale.

Based on it, we hope to answer the following research question in this study: *RQ1 How do DSD teams that apply scaling frameworks tailor agile methods in large-scale settings?*

2. Research Context

Since agile methods have become a successful norm in the mainstream of software engineering, large organizations are now starting to scale such methods [Alqudah and Razali 2016] by embracing scaling agile frameworks, such as SAFe, LeSS, DAD, Spotify model, Nexus. Further, globalization led the companies to distribute their activities across their national boundaries, which allowed the advent of DSD (Distributed Software Development) and GSD (Global Software Development). Most of the time, DSD and GSD teams are working on large-scale contexts, which requires a high level of coordination and collaboration that such frameworks promise to deliver.

Based on the research problem presented in section 1. We first conducted an exploratory review to aggregate relative aspects of the research theme. Second, the outcomes of the exploratory review helped develop an SLR (Systematic Literature Review) protocol about the topic, which is being conducted now to answer the RQs initially. Third, we aim to conduct a case study on a Brazilian fintech company with more than 3000 employees and 1000 IT specialists to investigate how the teams tailor agile.

One of the research goals is to present the agile methods in a tailored and scaled view from the literature of large-scale. Also, since there are some case studies from the industry, it still lacks more of those studies conducted by the academy. Further, we expected to help the practitioners with the results from the SLR and the output of a real industry case. Finally, the employees and the company board can be benefited by better understanding what the teams are doing in the day-to-day.

We intend to compare the tailored practices from a real industry case with those presented in the literature during the case study.

3. Background

Based on our research problem, the following themes need to be discussed: agile global software development, scaling agile, scaling agile frameworks.

Agile Global Software Development (AGSD) - The presence of agile practices in GSD was seen since 2002, [Jalali and Wohlin 2012] one year after the launch of the Agile Manifest [Beck et al. 2001]. “Agile Global Software Development” (AGSD) or “distributed agile software development” (DASD) are the terms used to configure the application of agile practices, methodologies, and techniques on globally distributed projects [Paasivaara et al. 2008]. Furthermore, the use of agile practices is also related to the success of GSD projects [Hossain et al. 2009]. Paasivaara *et al.* [Paasivaara et al. 2008] reported that the use of agile practices on large and distributed software development projects with challenges related to technologies and requirements could be benefited from agile practices.

Scaling Agile - In response to the difficulty of carrying traditional agile methods into large-scale projects and organizations [Fitzgerald et al. 2013], several scaling agile frameworks have emerged, such as DAD, SAFe, Spotify model, Nexus, and LeSS. According to Scott Ambler [Ambler 2013], scaling agile could be divided into three interpretations that consider the scaling. First, scaling the agile through the organization means scaling agile across teams, which needs a high level of collaboration. Second, scaling agile for other projects means scaling the size of the project. Third, scaling the agile to the entire value stream of an organization means to scale and maximize the development of business value without focusing only on the agile teams.

Also, since the number of teams is a determining factor in the project's scale, this study follows the taxonomy presented by [Dingsøy et al. 2014], which defines the scale of agile software development projects based on the number of teams involved. Small-scale projects usually have only one team. Large-scale projects can have 2 to 9 teams. Finally, very large-scale projects are composed of 10 or more teams.

Scaling Agile Frameworks - In the road to ease the insertion of agile in globally distributed teams, many companies choose frameworks to scale agile practices. In the following lines, we are going to provide a brief description of five of the most common scaling agile frameworks [Alqudah and Razali 2016].

- *DAD* - Disciplined Agile Delivery [Ambler and Lines 2012] is a framework that extends Scrum practices and combines them with other agile methods;
- *LeSS* - Large Scale Scrum [Larman and Vodde 2016] is a framework that has two approaches: (i) The LeSS framework is designed for medium companies, and (ii) LeSS huge designed for large companies;
- *SAFe* - Scaled Agile Framework focuses on scale agile on large enterprises crossing all of its sectors [Leffingwell, Dean 2020];
- *Spotify* - The Spotify model [Henrik and Anders 2012] focuses on organizing the teams into squads, which each squad uses its preferable agile practices;
- *Nexus* - The Nexus framework provides a structure that consists of a set of roles, practices, and events that serve for large DSD teams [Schwaber 2015].

4. Research Method

4.1. Systematic Literature Review

We intend to aim to answer both research questions through an SLR following the Guidelines of Kitchenham [Kitchenham and Charters 2007]. The research protocol is currently designed (See in <https://bit.ly/315QRHy>), contemplating the research questions, exclusion and inclusion criteria for studies selection, the research string used, the digital libraries chosen, quality evaluation guide, data extraction and analysis process.

The search produced 1520 references from 2006 to 2021. The following Table 1 shows the number of studies extracted from each engine through the search string and how many were accepted in each phase of the extractions.

Each selected publication was examined to extract information regarding: (i) the study aim or research question; (ii) other results relevant to the study; (iii) potential themes emerging from the study's conclusions; and (iv) practices, model/framework adaptations, organizational context, and business domain.

Engine	Selection	Phase 1	Phase 2
IEEE	31	26	16
ACM	836	69	24
Springer	191	61	27
Wiley	404	9	2
Scopus	58	82	22
Total	1520	247	91

Table 1. Papers by engine.

4.2. Case Study

We aim to conduct a case study in a large-scale fintech company from Brazil that has grown at least 100% in each year since 2018. The organization holds a software product used by over 14 million users. Besides it, it has diverse solutions in the financial sector, expanding its market share for other countries. The company comprises more than 3000 employees, and more than 1000 IT specialists. Also, the organization is using the Spotify model to coordinate more than 30 squads with several teams spread across Brazil.

The case study is still in the planning phase. However, since the case study method allowed the researchers to go deeper into the studied scenario, we aim to consider the whole company as the case and the unit of analysis as the teams. Since teams are related to squads and tribes, we hope to present how the members are using the Spotify model to tailor agile in such a dynamic environment. Also, since the SLR output would be a setlist of practices and models tailored from agile, we aim to compare the practices used by the teams in the case in contrast with the status of the art.

5. Research Status

Currently, we are conducting the SLR. Half of the 91 papers selected for the extraction phase were thoroughly read and had their data extracted. We aim to finish such a phase by the beginning of October. After the extraction, we aim to synthesize such data.

The Figure 1 describes the research' steps. The completed steps were painted green, the current step is yellow, and the future steps are white.

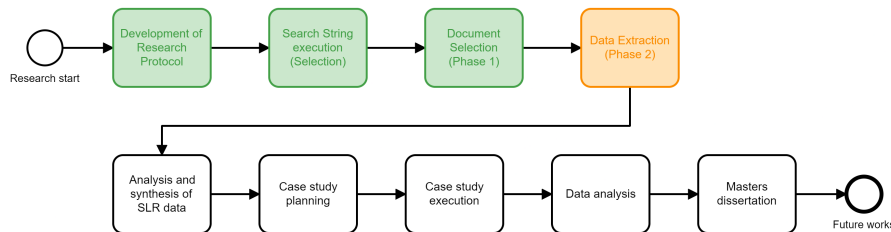


Figure 1. Research Process

6. Related Works

Edison *et al.* [Edison et al. 2021] conducted a systematic literature review to compare the main scaled agile frameworks and methods. The authors compared the scaled agile methods by showing their principles, practices, tools, and metrics. It also shows the original method specifications and the extensions and modifications identified in the empirical

papers selected. Finally, challenges and success factors faced by the teams while scaling agile on a large scale are presented in an organized view divided by each scaled agile method.

Bass and Salameh [Salameh and Bass 2019] conducted an embedded case study to evaluate the applicability of the Spotify Model and also the agile tailoring on a large-scale B2B (Business-to-Business) project of a financial company. The paper presented 44 tailored practices. From those, the authors classified six as modified from the Spotify model to fit the organization's needs, 28 as newly introduced by the teams and ten practices used by the teams that initially came from the Model. Despite the results presented by the authors, we aim to present the tailored practices of the Spotify model in a financial company that matches with the B2S (Business-to-Society) model from Spotify.

Paasivaara and Lassenius [Paasivaara and Lassenius 2016] conducted a case study at Nokia to evaluate how 20 teams with 170 members distributed across four countries scaled Scrum through the adoption of LeSS. The paper's main contribution is summarized by describing the five central practices adopted by the teams. Further, the authors summarized the differences among the practices chosen by the project teams and the practices from LeSS Framework. Besides it, we also aim to compare the tailored practices from our case study with the Spotify model and evaluates whether the changes made harmed the teams.

7. Expected Results

The topic of tailoring agile while scaling in large-scale settings is being constantly researched. However, the research literature requires real industry cases that present relevant results regarding the tailoring of agile practices in different large-scale contexts. Most of the studies in this context were conducted in Europe by evaluating the adoption of an agile scaling framework. Also, many industrial reports regarding their successful adoptions in the industry are presented on the Framework homepages, although academic studies evaluating such adoptions are still needed.

From the SLR, we aim to build a setlist of tailoring practices and models from agile in a distributed, large-scale context. Also, we aim to summarize the adaptations based on the cases identified and the scaling frameworks used by those cases.

Further, we aim to compare the content present in the setlist with the tailored practices used by the teams in the case study. Since the Spotify model does not specify what practices the teams should use, we hope to present how the teams tailor the practices with the Spotify model in a large-scale distributed company case.

We aim to present state-of-the-art for the community regarding agile tailoring literature for large-scale. Also, we aim to show how teams from a large-scale company in Brazil are genuinely tailoring agile practices in an emergent market using the Spotify model with B2S products in the financial sector.

References

- Alqudah, M. and Razali, R. (2016). A review of scaling agile methods in large software development. *IJASEIT*, 6:828.
- Ambler, S. (2013). Going beyond scrum disciplined agile delivery by.

- Ambler, S. W. and Lines, M. (2012). *Disciplined Agile Delivery: A Practitioner's Guide to Agile Software Delivery in the Enterprise*. IBM Press, 1st edition.
- Beck, K., Beedle, M., van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., Grenning, J., Highsmith, J., Hunt, A., Jeffries, R., Kern, J., Marick, B., Martin, R. C., Mellor, S., Schwaber, K., Sutherland, J., and Thomas, D. (2001). Manifesto for agile software development.
- Dingsøy, T., Fægri, T. E., and Itkonen, J. (2014). What is large in large-scale? a taxonomy of scale for agile software development. In *PROFES*, pages 273–276. Springer.
- Dybå, T. and Dingsøy, T. (2008). Empirical studies of agile software development: A systematic review. *Information and software technology*, 50(9-10):833–859.
- Edison, H., Wang, X., and Conboy, K. (2021). Comparing methods for large-scale agile software development: A systematic literature review. *IEEE TSE*, pages 1–1.
- Fitzgerald, B., Stol, K.-J., O'Sullivan, R., and O'Brien, D. (2013). Scaling agile methods to regulated environments: An industry case study. In *2013 35th ICSE*, pages 863–872.
- Henrik, K. and Anders, I. (2012). Scaling agile @ spotify with tribes, squads, chapters & guilds.
- Hossain, E., Babar, M. A., and Paik, H. (2009). Using scrum in global software development: A systematic literature review. In *2009 4th IEEE ICGSE*, pages 175–184.
- Jalali, S. and Wohlin, C. (2012). Global software engineering and agile practices: A systematic review. *Journal of Software: Evolution and Process*, 24.
- Kitchenham, B. and Charters, S. (2007). Guidelines for performing systematic literature reviews in software engineering. Technical report, EBSE Technical Report.
- Larman, C. and Vodde, B. (2016). *Large-Scale Scrum: More with LeSS*. Addison-Wesley Professional, 1st edition.
- Leffingwell, Dean (2020). Scaled Agile Framework 5.0. <https://www.scaledagileframework.com/whats-new-in-safe-5-0-2/>. [Online; accessed 16-May-2021].
- Paasivaara, M., Durasiewicz, S., and Lassenius, C. (2008). Distributed agile development: Using scrum in a large project. In *2008 IEEE ICGSE*, pages 87–95.
- Paasivaara, M. and Lassenius, C. (2016). Scaling scrum in a large globally distributed organization: A case study. In *2016 IEEE 11th ICGSE*, pages 74–83.
- Razzak, M., Noll, J., Richardson, I., Canna, C. N., and Beecham, S. (2017). Transition from plan driven to safe@: Periodic team self-assessment. *ArXiv*, abs/1711.00959.
- Rizvi, B., Bagheri, E., and Gasevic, D. (2015). A systematic review of distributed agile software engineering. *J. Softw. Evol. Process*, 27(10):723–762.
- Salameh, A. and Bass, J. (2019). Spotify tailoring for b2b product development. In *2019 45th Euromicro Conference SEAA*, pages 61–65.
- Schwaber, K. (2015). Nexus guide. the definitive guide to nexus: the exoskeleton of scaled scrum development.