IJARSCT

ISSN (Online) 2581-9429

International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)

International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal

Impact Factor: 7.67

Volume 5, Issue 4, March 2025

A Study on Investigating the Role of Architects in Scaling Agile Frameworks

Kiran R. Borade, Gayatri R. Jagtap, Chandrabhan R. Ghuge
Guru Gobind Singh Polytechnic Nashik, Maharashtra, India

Abstract*: This article describes the role of architects in evaluating agile processes with the help of data analysis. The Scalable Agile Framework is a knowledge-based integration and workflow model designed to guide companies in scaling their agile workflows. Our goal is to provide an initial analysis of a scalable agile framework. Three popular scalable agile frameworks are described: Scalable Agile Framework, Big Scrum Scrum, and Disciplined Agile 2.0. After explaining the basic concepts of the scalable agile framework, the roles of enterprise, software, solution, and data architects in the four scalable agile frameworks described are defined. Finally, we discuss the broader role of architects in scaling agile processes.*

Keywords: Agile development, Scaling agile, Scaling frameworks, Scaling practices, Framework comparison, Subway Map

I. INTRODUCTION

Enterprises struggle to deal with unpredictable competitive environments due to rapidly changing customer demands, regulatory changes, and technological advancements that can lead to the enterprises success. Software development projects in such environments face changes either directly or indirectly[1]. To resolve such challenges agile leads to development of agile manifesto and many agile software development methods, including extreme programming (XP), and scrum and with this methods team work collaborately. Large enterprises are interested in extending agile methods to include larger teams.

The main objective of this study is to investigate the role of architects in scaling agile frameworks. Based on this objective, three research questions (RQ) were formulated

RQ1: What are the types of scaling agile frameworks?

RQ2: What are the roles of architects in scaling agile frameworks?

RQ3: What are the generalize findings about the role of architects in scaling agile frameworks?

II. CONTRIBUTION

In the first phase, we defined the scope of the review and identified suitable research questions about the role of architects in scaling agile frameworks. In the second phase, key concepts were identified by concept mapping, which also provided the opportunity to identify additional relevant search terms. These were applied to the subsequent literature search in the third phase. In the fourth phase, we created a concept matrix just supposing the different architect roles with the identified scaling agile frameworks to investigate their roles within the scaling agile frameworks. In the last phase, the comparative analysis of architect roles resulted in generalize findings on the role of architects in scaling agile frameworks.

III. SCALED AGILE FRAMEWORKS

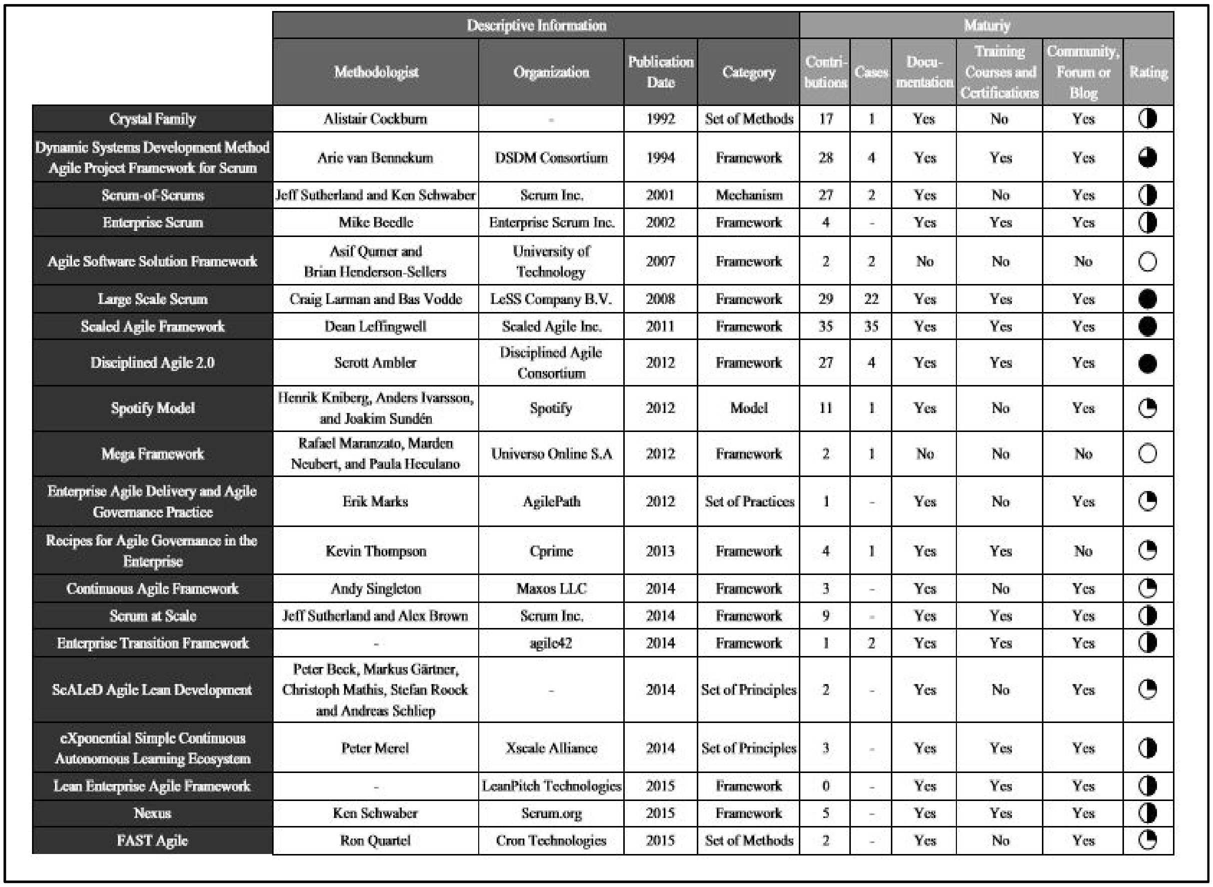
The main purpose of this framework is to manage large Agile teams of 50+ developers spread across multiple domains within the Agile framework. Traditional agile methods like Scrum cannot handle multiple developers. Definition of Scaling Agile Development:

Scaling agile methods introduces new challenges such as collaboration and division of labor among teams without formal rules or regulations [2].

DOI: 10.48175/IJARSCT-24068

Definition: The term scaled agile development refers to agile development from large teams to large multi-team projects that want to leverage agile development at a portfolio or enterprise level.

Introduction to the most popular methods: The literature review of the descriptive process revealed 20 scalable agile methods listed in Figure3.1 [3]. Large multi-team projects. LeSS, SAFe and DA 2.0 frameworks are particularly mature because they are frequently mentioned in the literature, so we evaluate it.



	Descriptive Information				Maturiy					
	Methodologist	Organization	Publication Date	Category	Contri-butions	Cases	Docu-mentation	Training Courses and Certifications	Community, Forum or Blog	Rating
Crystal Family	Alistair Cockburn	-	1992	Set of Methods	17	1	Yes	No	Yes	◑
Dynamic Systems Development Method Agile Project Framework for Scrum	Arie van Bennekum	DSDM Consortium	1994	Framework	28	4	Yes	Yes	Yes	◕
Scrum-of-Scrums	Jeff Sutherland and Ken Schwaber	Scrum Inc.	2001	Mechanism	27	2	Yes	No	Yes	◑
Enterprise Scrum	Mike Beedle	Enterprise Scrum Inc.	2002	Framework	4	-	Yes	Yes	Yes	◑
Agile Software Solution Framework	Asif Qumer and Brian Henderson-Sellers	University of Technology	2007	Framework	2	2	No	No	No	○
Large Scale Scrum	Craig Larman and Bas Vodde	LeSS Company B.V.	2008	Framework	29	22	Yes	Yes	Yes	●
Scaled Agile Framework	Dean Leffingwell	Scaled Agile Inc.	2011	Framework	35	35	Yes	Yes	Yes	●
Disciplined Agile 2.0	Scrott Ambler	Disciplined Agile Consortium	2012	Framework	27	4	Yes	Yes	Yes	●
Spotify Model	Henrik Kniberg, Anders Ivarsson, and Joakim Sundén	Spotify	2012	Model	11	1	Yes	No	Yes	◔
Mega Framework	Rafael Maranzato, Marden Neubert, and Paula Heculano	Universo Online S.A	2012	Framework	2	1	No	No	No	○
Enterprise Agile Delivery and Agile Governance Practice	Erik Marks	AgilePath	2012	Set of Practices	1	-	Yes	No	Yes	◔
Recipes for Agile Governance in the Enterprise	Kevin Thompson	Cprime	2013	Framework	4	1	Yes	Yes	No	◔
Continuous Agile Framework	Andy Singleton	Maxos LLC	2014	Framework	3	-	Yes	No	Yes	◔
Scrum at Scale	Jeff Sutherland and Alex Brown	Scrum Inc.	2014	Framework	9	-	Yes	Yes	Yes	◑
Enterprise Transition Framework	-	agile42	2014	Framework	1	2	Yes	Yes	Yes	◑
ScALeD Agile Lean Development	Peter Beck, Markus Gärtner, Christoph Mathis, Stefan Roock and Andreas Schliep	-	2014	Set of Principles	2	-	Yes	No	Yes	◔
eXponential Simple Continuous Autonomous Learning Ecosystem	Peter Merel	Xscale Alliance	2014	Set of Principles	3	-	Yes	Yes	Yes	◑
Lean Enterprise Agile Framework	-	LeanPitch Technologies	2015	Framework	0	-	Yes	Yes	Yes	◑
Nexus	Ken Schwaber	Scrum.org	2015	Framework	5	-	Yes	Yes	Yes	◑
FAST Agile	Ron Quartel	Cron Technologies	2015	Set of Methods	2	-	Yes	No	Yes	◔

Figure 3.1 primary analyses of Scaling agile frameworks[3]

LeSS framework, named LeSS Huge. It introduces additional scaling factors, which can be required to manage loads of builders in massive businesses. less massive introduces a new concept, specifically, requirement areas (RAs).

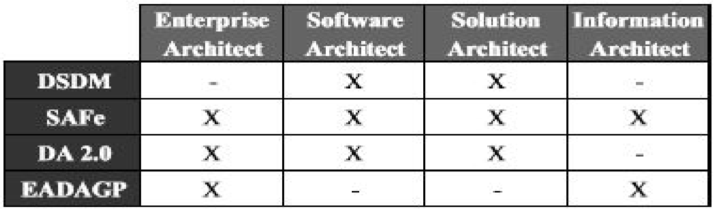
All secure SAFe (version four.zero.) groups are a part of one agile release educate (art), a crew of agile teams that offers a continuous flow of incremental releases of price. on the program stage, safe extends Scrum the use of the equal thoughts however on a better degree. safe highlights four ranges of organization: group, software, value stream, and portfolio.

The DA 2.0 framework: In comparison with secure, DA 2.zero pursuits to address areas that are not very well covered in smaller scaling agile frameworks and recommends three stages: inception, production, and transition.

IV. THE ROLE OF ARCHITECTS IN SCALING AGILE FRAMEWORKS

Since traditional agile methods, such as XP or Scrum, do not include the role of architects, we recognize that this is no longer valid for scaling agile frameworks.[5] We have selected a set of predominant architect roles relevant to the realm of enterprise architecture (EA) from refs. and to describe their function in scaling agile frameworks. These roles comprise the enterprise architect, software architect, solution architect, and information architect.

DOI: 10.48175/IJARSCT-24068



	Enterprise Architect	Software Architect	Solution Architect	Information Architect
DSDM	-	X	X	-
SAFe	X	X	X	X
DA 2.0	X	X	X	-
EADAGP	X	-	-	X

Figure 4.1 Identified Roles in Scaling agile frameworks [5].

We have searched each relevant source for these architect roles and related them to the different scaling agile frameworks. The results are summarized in Figure 4.1.[5] Only 4 out of 20 scaling agile frameworks include architect roles, namely Dynamic Systems Development Method Agile Project Framework for Scrum (DSDM), SAFe, DA 2.0, and Enterprise Agile Delivery and Agile Governance Practice (EADAGP).

The following sections present our results about the role of architects in various scaling agile frameworks.

The role of enterprise architect:

The key responsibilities of the enterprise architect in SAFe are:

- Maintaining a high-level and holistic vision of enterprise solutions and development initiatives.
- Understanding and communicating strategic themes and other key business drivers for architecture to system architects and nontechnical stakeholders.
- Working with business stakeholders and software and solution architects to drive holistic technology implementation across value streams.
- Working closely with software and solution architects to ensure that individual program and product strategies align with enterprise objectives
- Participating in the strategy for building and maintaining the enterprise architectural runway.

The key responsibilities of the enterprise architect in DA are:

- Supporting and collaborating closely with stakeholders on a regular basis to understand their needs and to develop the organization’s roadmap.
- Supporting and collaborating closely with agile teams on a regular basis to guide them through the business and technical roadmaps and help them to identify potentially reusable assets and technical debts.
- Negotiating technical dependencies between solutions.
- Exploring architectural views and
- Adopting and tailoring architectural frameworks.

The key responsibilities of the enterprise architect in EADAGP are:

- Escalating governance issues to IT governance.
- Supporting agile teams by making appropriate governance decisions and providing guidance.
- Communicating governance decisions to enterprise IT governance for their agreement;
- Supporting the governance backlog grooming process.
- Harmonizing governance requirements across sprints and agile teams.
- Reporting technology and architecture requirements/issues to EA oversight for alignment and issue resolution.
- Identifying security requirements and challenges that may not have been pre-determined and
- Raising potential compliance and risk requirements that have to be reviewed and signed off by governance, risk and compliance bodies, and EA sign offs.

DOI: 10.48175/IJARSCT-24068

The role of software architect:
The responsibilities of the software architect in DSDM are:
• Agreeing and controlling the technical architecture;
• Identifying and owning architectural and other technically based risks;
• Working with business analysts to evaluate the technical options and decide on the best way to turn the highlevel business requirements into a technical solution;
• Promoting standards for technical best practice;

The responsibilities of the software architect in SAFe are:
• Defining NFRs, major system elements, subsystems, and interfaces.
• Preparing the architecture vision briefing within the program increment (PI) planning event.
• Presenting the architecture vision, which may include descriptions of new architectural epics for common infrastructure, any large-scale refactors under consideration, and system-level NFRs and
• Supporting the PO by refining the team backlog.

The responsibilities of the software architect in DA are:
• Guiding the creation and evolution of the solution architecture that the team is working on.
• Mentoring and coaching other team members with best practices of architecture.
• Understanding the architectural direction and standards.

The role of solution architect:
The exclusive responsibilities of the solution architect for SAFe and DSDM are:
• Supporting the solution management by managing the value stream kanban.
• Discussing upcoming enabler capabilities and epics with the solution management.
• Defining the overarching architecture that connects the solution across ARTs.
• Working with the system architect to guide the architecture developed by the ARTs.
• Ensuring technical alignment with the solution context, including interfaces and constraints.
• Attending to the value stream and ART PI planning events and
• Updating progress toward milestones, program PI objectives, and dependencies among the ARTs.

The role of information architect:
The role of information architect is only supported by SAFe and EADAGP.
The role of information architect in SAFe is:
• Participating in PI planning as well as in pre and post PI planning.
• Driving requirements and taking ownership of dependent backlog items.
• Collaborating with agile teams to fulfill dependencies during PI executions and
• Participating in system and solution demos.

The role of information architect in EADAGP is:
• Escalating and communicating governance issues and decisions to IT governance.
• Supporting agile teams by making appropriate governance decisions and providing guidance.
• Supporting the governance backlog grooming process
• Harmonizing governance requirements across sprints and agile teams.
• Surfacing technology and architecture requirements.
• Identifying security requirements and challenges that may not have been pre-determined and
• Raising potential compliance and risk issues.

DOI: 10.48175/IJARSCT-24068

V. KEY POINTS IN DISCUSSION

After studying different papers following key points were identified:

Increasing development speed by balancing emergent and intentional architecture design.

Finding the right balance between centralized and decentralized architectural decision making.

Sparing agile development from traditional IT governance.

Ensuring the reuse of enterprise assets.

Architects are aware of existing enterprise assets, e.g., patterns and standards, which are available for reuse, and ensure that agile teams utilize them where applicable. This accelerates the development process and reduces time to market.

VI. CONCLUSION

This study investigates the critical role that architects play in scaling Agile frameworks within organizations, shedding light on their unique contributions, challenges, and strategies for success. As Agile methodologies evolve and are scaled from individual teams to enterprise-wide frameworks, the role of architects becomes increasingly significant in ensuring that the overall system architecture remains cohesive, flexible, and scalable.

Architects, while traditionally associated with the design and implementation of technical architecture, are pivotal in scaling Agile because they bridge the gap between the high-level strategic vision and the operational details of Agile development. Their responsibilities extend beyond merely maintaining technical integrity; they play a central role in fostering collaboration, aligning cross-functional teams, managing complex dependencies, and ensuring that architectural decisions are in line with both business goals and Agile principles.

Key findings from this study highlight that successful scaling of Agile requires architects to embrace Agile values such as flexibility, iterative delivery, and collaboration. However, architects also face significant challenges, including resistance to change, maintaining long-term vision while accommodating short-term iterations, and managing the complexity of systems across multiple Agile teams. The ability to balance these demands requires both technical expertise and strong leadership and communication skills.

Furthermore, the research suggests that a key factor in the success of scaling Agile frameworks lies in the continuous learning and adaptation of architects. Architects must work closely with Agile coaches, Scrum Masters, and Product Owners to ensure that the architecture evolves iteratively and adapts to changing requirements and emerging technologies. Their role is not static but requires ongoing reflection, feedback, and alignment with Agile principles.

In conclusion, architects are integral to the success of scaling Agile frameworks. Their expertise in system design, coupled with a deep understanding of Agile practices, positions them as crucial enablers of scalable, sustainable, and successful Agile transformations. Organizations aiming to scale Agile frameworks should place a strategic emphasis on developing the architectural capabilities of their teams, providing the necessary support and training for architects, and ensuring their involvement throughout the Agile adoption process.

In this study, we have motivated the need for scaling existing agile methods to large-scale agile development due to their deficiencies in inter-team coordination and communication. We have then presented a primary analysis of the identified scaling agile frameworks. Based on our maturity assessment, we have identified LeSS, SAFe, and DA 2.0 as the most mature frameworks. Finally, we have extensively characterized the different architect roles that have been identified in scaling agile frameworks. Our findings indicate that architects both actively and passively support agile teams by driving architectural initiatives, participating in architectural runways, harmonizing governance requirements, and ensuring technical alignment in solution contexts.

Future research may analyze the challenges faced by architects in scaling agile environments by conducting case studies in organizations that can provide practical experience of adopting scaling agile frameworks.

REFERENCES

[1]. Ambler, S. W., & Lines, M. (2012). *Disciplined Agile Delivery: A Practitioner's Guide to Agile Software Delivery in the Enterprise*. IBM Press.

[2]. Cohn, M. (2010). *Succeeding with Agile: Software Development Using Scrum*. Addison-Wesley Professional.

[3]. Highsmith, J. (2009). *Agile Project Management: Creating Innovative Products* (2nd ed.). Addison-Wesley.

[4]. Leffingwell, D. (2017). *SAFe 4.0 Reference Guide: Scaled Agile Framework for Lean Software and Systems Engineering*. Addison-Wesley.

[5]. Larman, C., & Vodde, B. (2016). *Scaling Lean & Agile Development: Thinking and Organizational Tools for Large-Scale Scrum*. Addison-Wesley.

[6]. Mathis, D., & Hanney, S. (2013). *Architecting the Agile Enterprise*. IBM Redbooks.

[7]. Poppendieck, M., & Poppendieck, T. (2003). *Lean Software Development: An Agile Toolkit*. Addison-Wesley.

[8]. Schwaber, K., & Beedle, M. (2002). *Agile Software Development with Scrum*. Prentice Hall.

[9]. Sutherland, J., & Schwaber, K. (2017). *The Scrum Guide: The Definitive Guide to Scrum: The Rules of the Game*. Scrum.org.

[10]. West, D., & Grant, T. (2010). *Agile Governance and Audit: Steering Teams and Projects in a Complex Agile Environment*. Wiley.