The American Journal of Engineering and Technology
ISSN 2689-0984 | Open Access

TYPE Original Research
PAGE NO. 135-143
DOI 10.37547/tajet/Volume07Issue06-15

Check for updates

Scalable Agile Frameworks: Comparing Safe, Less, And Nexus for Enterprise Adoption

Dhanasekar Elumalai
Fidelity Investments

OPEN ACCESS

SUBMITTED 24 April 2025
ACCEPTED 17 May 2025
PUBLISHED 19 June 2025
VOLUME Vol.07 Issue 06 2025

CITATION

Dhanasekar Elumalai. (2025). Scalable Agile Frameworks: Comparing Safe, Less, And Nexus for Enterprise Adoption. The American Journal of Engineering and Technology, 7(06), 135–143. https://doi.org/10.37547/tajet/Volume07Issue06-15

Abstract: This study aims to identify and systematically compare the main large-scale agile frameworks that companies can adopt to manage the work of large scale and distributed teams. The companies can consciously perform better decision on the choice of the framework that fits the practices and challenges of their organizations. The work employs a qualitative approach supporting an exploratory analysis identifying the processes of migration to large scale agile. First the assessment criteria for scaling agile are discussed. Second these criteria used to perform a comparative analysis fo 3 large scale agile frameworks i.e. SAFe, LeSS and Nexus. The findings reveal there isn't a dominant large scale agile framework in all dimensions. However, framework like Nexus offer low technical complexity accommodating the changes easily while other frameworks like SAFe offer high level of scalability more demanding and deep efforts changing work processes in organization.

Keywords: SAFe, LeSS, Nexus, scalable, frameworks.

Introduction: Since the creation of the Agile Manifesto in 2001, agile software development methods have become mainstream in software projects that are continuously affected by external drivers, such as changing customer demands, shifting regulatory requirements, and new technological advances. Since then, academics and practitioners have devoted a great deal of attention to agile methods. Agile methodologies, such as Extreme Programming and Scrum, were initially created for small, co-located, and self-organizing teams developing software in close collaboration with customers applying short iterations. Given the successful adoption of agile methods in small organizations and projects, many software practitioners

are increasingly engaged in using these methods in large projects and organizations. Some existing agile methods and software practitioners, who have worked with companies to scale agile approaches to their settings, have promoted various agile scaling frameworks, such as Large-Scale Scrum (LeSS)1 and Scaled Agile Framework (SAFe) to address issues related to the adoption of agile methods in large companies and projects. As large organizations face increasing pressure to become more agile, the adoption of agile scaling frameworks in the industry has grown, to provide off-the-shelf solutions for scaling agile methods.

Even though there is a growing body of knowledge on agile scaling frameworks, the existing literature neglects to investigate why companies select a specific framework, what benefits they expect, and how satisfied they are with selected framework. We understand this gap in the literature by presenting our findings on the reasons, expected benefits, and satisfaction of organizations adopting agile scaling frameworks of industry participants worldwide.

Agile methodologies play a vital role in software development in teamwork from team flexibility collaboration and have been delivering high quality products, the need of scaling agile practices is crucial for organizational growth, maintaining efficiency and alignment across multiple teams. This. Paper aiming to compare 3 popular scaling frameworks: SAFe (Scaled Agile framework), Nexus, and LeSS (Large scale scrum) focusing for suitability of enterprise adoption. Let's examine their principle, benefits, challenges and real-world examples providing comprehensive analysis to guide the most appropriate framework for the needs.

1. SAFe (Scaled Agile Framework):

SAFe is framework designed to scale agile practices for large organizations integrating Lean, Agile and Devops principle structured for aligning teams, programs and portfolios. The principles of SAFe: (Fig 1. Refer for the big picture of the SAFe framework)

- Lean- Agile mindset
- Agile Release Train (ART)
- Continuous delivery pipeline

2.1 Lean- Agile mindset is combination of lean thinking and agile principles aiming a culture of continuous improvement, collaboration and customer focused in an organization emphasizing the importance of delivering value effectively and efficiently to the changing circumstances. Lean thinking focus maximizing value and minimizing waste encouraging to process streamline, eliminate inefficiencies and optimize work flow. Key principles include **Value, Value stream, flow, pull and perfection.**

2.1.1 Value defined from the customers perspective which is derived by engaging with customer to understand their needs and deliver value.

2.1.2 Value stream will identify, and map steps required to deliver value, this is adopted to changing requirement and market conditions

2.1.3 Flow of work throughout the value stream smoothly and continuous via iterative development cycles.

2.1.4 Pull will produce the work based on demand rather then forecasts in collaboration with diverse team members.

2.1.5 Perfection by continuously improving the process reflecting the practices.

The lean agile mindset integrates agile principles with lean thinking creating an approach to deliver value by encouraging Embrace change an opportunity for improvement rather disruption, focus on value delivering for customers and stakeholders, Optimize flow ensuring efficient delivery of work, empower team by making decisions and ownership of their work, continuous learning culture and experimentation promoted. This benefits by reducing waste and optimizing process to deliver value faster, enhance collaboration between cross functional teams working efficiently together, Flexibility adapting to changing circumstances and customer needs, Higher quality of products and services delivered, Increased customer satisfaction by consistent value delivered.

2.2 Agile Release Train (ART) is one objective of delivered value by agile teams working in synchronous, having a fixed schedule and following a cadence of program increments (PIs) lasting 8-12 weeks. The ART designed aligning teams, stakeholders and business

objectives all working towards a common goal. Key components of ART include

2.2.1 Program increment (PI) which is a timeboxed ART delivery incremented in a 8-12 week durations comprising of multiple iterations.

2.2.2 Agile teams with cross functional knowledge practicing Scrum or Kanban to collaborate and deliver features and capabilities.

2.2.3 PI Planning a two day even kickoff each program increment along with teams and stakeholders together planning the upcoming PI work. While setting objectives related dependencies and risks identified and mitigated.

2.2.4 System demo to demonstrate the integrated work during the PI providing visibility of progress and gathering feedback from stakeholders.

2.2.5 Inspect and Adapt an workshop held end of each PI to review the outcome, identify improvements and plan for next PI comprising on continuous improvement and problem solving.

The following are the benefits of ART ensuring teams working towards common objective and business goals, provide predictability schedule of delivered value reducing uncertainty, Collaboration among teams, stakeholders and business leaders, continuous delivery of value through regular increments, enhances transparency in progress, dependencies and risks. There are challenges for ART in terms of coordination required among multiple teams and stakeholders, managing dependencies and integration can be complex, training teams and coaching to adopt ART practices effectively, change management facing resistance in organization implementing it. (Figure 1. Refer for the big picture of the SAFe framework)

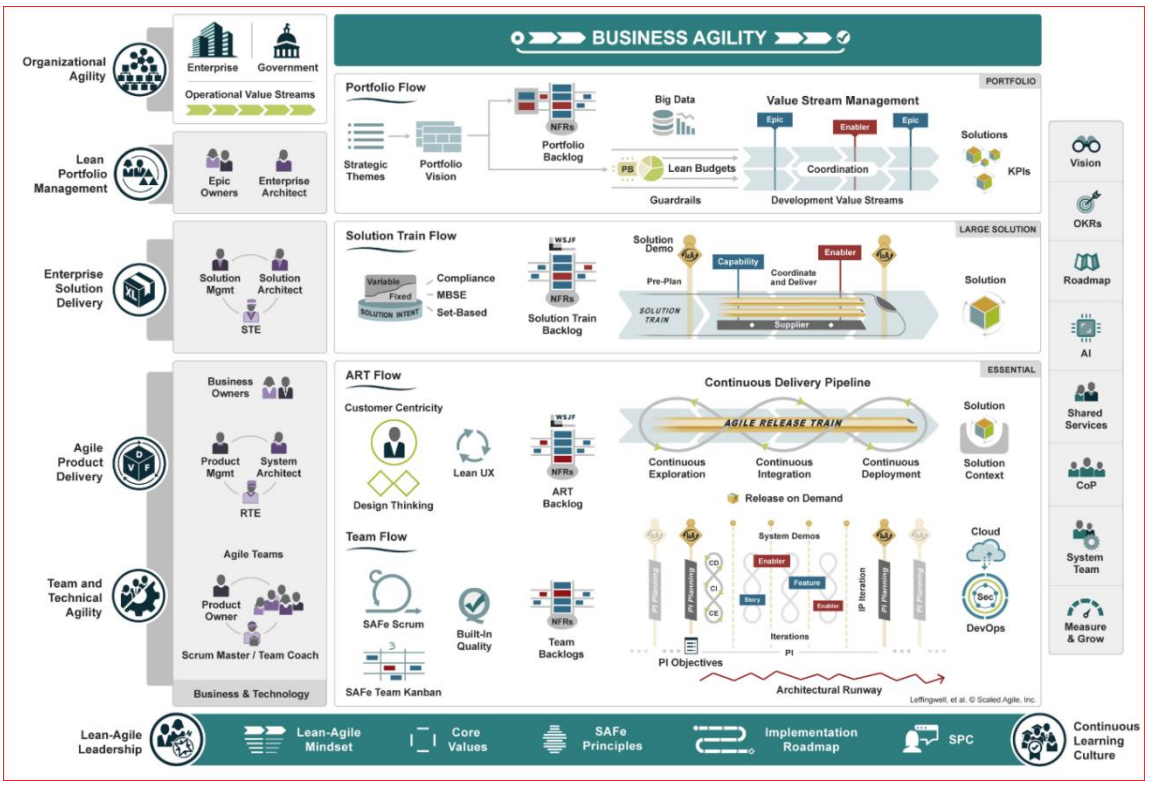

Figure 1. SAFE Big picture depicting roles and framework

The Agile release train being powerful mechanism scaling agile practices across large organizations providing structured approach delivering value, helping teams stay aligned, collaborate effectively and improve process continuously.

2.3 Continuous Delivery pipeline represents workflow, activities and automation to deliver new features and capabilities encompassing the entire lifecycle of a product from ideation to deployment that value is delivered continuously and efficiently. Key components of the continuous delivery pipeline:

2.3.1 Continuous exploration: Keep focusing on identifying and refining new ideas and opportunities, involve in market research, customer feedback and

competitive analysis, ensure the organization working on most valuable and relevant features.

2.3.2 Continuous integration: Integrating frequent code changes and testing to ensure quality, automated testing, build and integration process used to detect issue early promoting collaboration among teams reducing integration risks.

2.3.3 Continuous Deployment: Deployment of code changes to production environments with new features delivered to users quickly and reliably by involving automated testing, release management and monitoring.

2.3.4 Release on Demand: Allowing organizations to release new features and capabilities as neeed and aligning business priorities and customer needs by involving feature toggles, canary releases and other techniques to manage releases.

Benefits of the continuous delivery pipeline are Faster time to market of new features and capabilities reducing lead time, Better Quality using automation for testing and integration process identifying issues early and resolving it, Enhanced collaboration among development testing and operations, Greater flexibility responding to changing market conditions and customer needs and finally increased customer satisfaction with value delivered continuous keeping customers happy.

Challenges of continuous delivery pipeline are Automation, Coordination, Change management and monitoring. Though it's the key to adhere for the SAFe approach has to be tackled in order for achieving faster delivery of features, improved quality and enhanced collaboration among teams. (Figure 1. Refer for the big picture of the SAFe framework)

SAFe can be done at portfolio, program and team levels, lets know at high level from each perspective in brief. Portfolio level it is important focusing on aligning strategy with execution ensuring investments are directed towards most valuable initiatives involving strategic themes, setting investment guardrails managing the portfolio backlog.

Strategic themes are high level business objectives guiding the portfolio direction helping all initiatives align with organizations strategic goals.

Investment guardrails help allocate funding different value streams and initiatives, ensuring the investments are balanced and aligned strategic priorities.

Portfolio backlog are list of prioritized epics and initiatives intending for portfolio delivery by continuously refine and prioritized on strategic themes business value.

Agile portfolio operations involve coordinating, supporting execution of value streams across portfolio managing dependencies, facilitating collaboration and ensuring teams having resources needing success. Lean portfolio management (LPM) oversees portfolio operations, ensuring investments aligned to strategic goals on value streams delivering effectively. Portfolio sync on regular cadence to review progress with stakeholders, address dependencies to keep portfolio on track. Governance for portfolio operation defining guardrails complying with organizational policies and standards.

Program level PI planning is critical event where all ART teams come together planning the work for upcoming program increment ensuring alignment, collaboration and shared understanding of objectives. Setting PI objectives with measurable objectives aiming to achieve the PI aligned overall goals of the ART.

Team level framework used to manage work iterative cycle emphasizing collaboration, transparency and continuous improvement. By implementing the key components at portfolio, program and team levels organizations can effectively scale agile practices delivering efficiently.

3. **Nexus :** Lets talk about this framework in terms of scaling on large product development initiatives involving multiple scrum teams, this approach was created by Ken Schwaber who was one of the co-creaters of Scrum principles. The framework aims to enhance collaboration, integration and transparency within the teams working on a single product ensuring value delivered efficiently and effectively.

The key components of Nexus:

3.1 Scrum based scaling involves applying the principles and practices to large scale product development

initiatives. This framework enables a structured approach managing multiple scrum teams working on single product. The principes of scrum-based scaling involves

* Transparency having visibility into each other's work and progress, promoting an open collaboration and communication.

* Inspection reviewing and accessing work identifying issues/ opportunities for improvement by involving in sprint review and retrospective

* Adaption by continuously improving processes and practices on feedback and insights encouraging to adapt on changing circumstances and customer needs.

* Collaboration among teams ensuring working together effectively and involving on regular ceremonies and communication channels.

The benefits of Scrum based scaling helps enhanced collaboration, improved transparency, greater flexibility and continuous improvement. There are challenges on the approach as well like coordination of managing dependencies and integration with multiple teams can be complex. Effective communication requires effort and discipline, change management could be challenging of resistance on new practices and framework. Training needs to coach and practices effectively for adoption. On overcoming and adopting to the principles will achieve better results to deliver efficient and effective product development.

3.2 **Nexus Integration Team (NIT)** is specialized team with the framework for ensuring multiple scrum teams operate effectively aligned and integrated. This plays a crucial role managing dependencies, facilitate communication and ensure integrated product increment meeting quality standards. The team comprises of members having deep understanding of product, technical expertise and strong facilitation skills, they include 1) product owner who represents stakeholders with well-defined prioritized product backlog. 2) Scrum master who facilitates nexus events, removes impediments and adhering to scrum principles 3) Integration specialists who is expert in integrating multiple teams work by resolving technical dependencies and cohesive to the product increment. 4) Representatives from scrum teams who provide insights about teams progress, challenges and dependencies.

The following are some of the events facilitated by the NIT team

3.2.1 Nexus Sprint

3.2.2 Nexus Daily scrum

3.2.3 Nexus sprint planning

3.2.4 Nexus sprint review

3.2.5 Nexus sprint retrospective

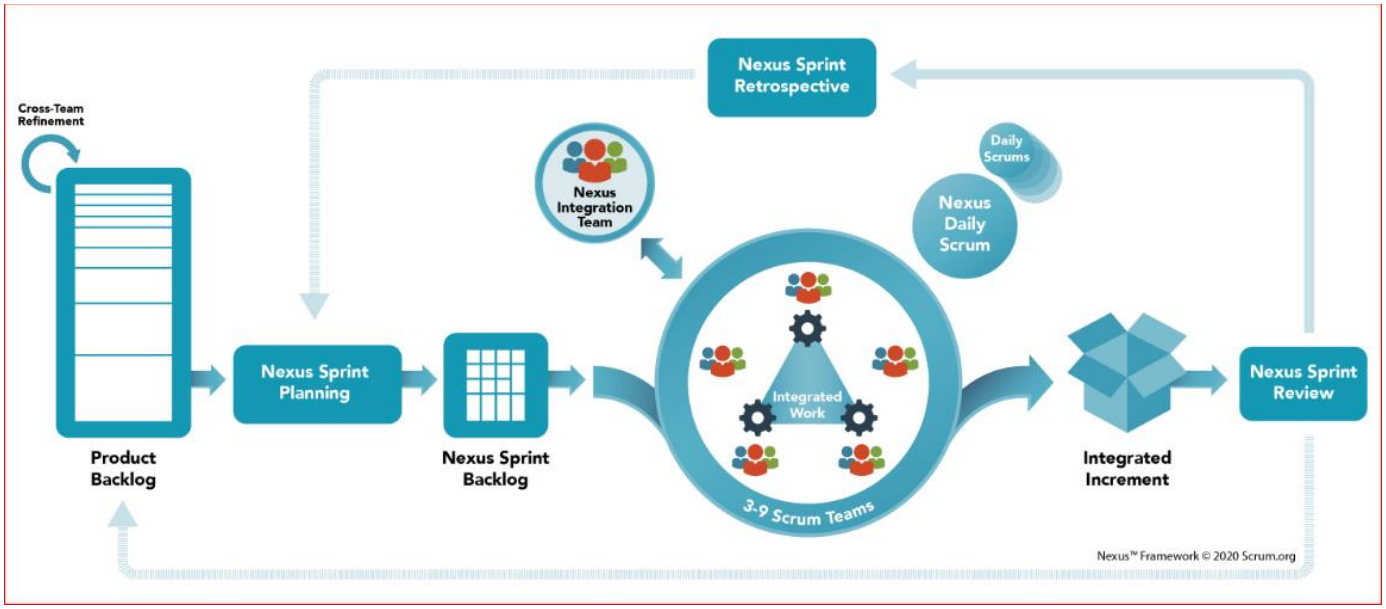

Fig 2. Nexus overall framework

3.2.1 Nexus Integration team is responsible in coordinating and integrating the work done by multiple scrum teams ensuring the dependencies are managed and aligning the team. They facilitate all communication and collaboration among multiple teams.

3.2.2 Nexus sprint is synchronized across all scrum teams working together delivering an integrated incremental product, involving regular ceremonies of sprint planning, daily scrum, sprint review and retrospective.

3.2.3 Nexus Daily scrum is a sync from representatives of each scrum team discussing progress, identify blockers and coordinating the work. It ensures teams are aligned and integration issues are promptly addressed.

3.2.4 Nexus sprint planning where all teams unite to plan the work for the upcoming sprint, identifying dependencies, set objectives and create a shared understanding of the work being done.

3.2.5 Nexus sprint review a forum the teams demonstrate the integrated work with all stakeholders obtaining feedback, it also provides visibility on progress and fosters collaboration.

3.2.6 Nexus sprint retrospective reflecting teams on their processes and uncover improvements, focused on continuous improvement and problem solving.

3.3 Nexus sprint synchronize all scrum teams with Nexus to ensure all teams work together delivering integrated increment of product. The Nexus sprint have the same principle and practice as scrum sprint but has additional layer of coordination and integration to manage complexities of scaling scrum.

Key elements of Nexus sprint is 1) **Sprint planning** a collaborative event of all scrum teams to plan upcoming sprint facilitated by NIT aligning the work and coordination. 2) **Daily scrum** with representative of each scrum team to discuss and coordinate work followed by 3) **Sprint review** and 4**) Sprint retrospective** to review the progress and continuous feedback loop to have enhanced collaboration, improved transparency, greater flexibility and continuous improvement.

The Nexus Sprint is powerful mechanism for scaling and ensuring multiple teams work effectively together to deliver high quality product increment. Through synchronized sprints, ceremonies and continuous improvement Nexus promotes a collaboration and adaptability enabling organization to respond changing market conditions and customer needs.

3. **LeSS (Large Scale Scrum)** applies principles at scale, emphasizing simplicity and effective communication for large organizations. The principle for LeSS:

3.**1 Scrum principles applied at scale**

3.2 **LeSS Framework**

3.3 **LeSS Huge**

Lets discuss in detail on the principle emphasizes the importance of scrum principles such as transparency, inspection and adaptation applying them at scale to ensure that large organizations can achieve the same benefits as smaller scrum teams.

3.1.1 Transparency is crucial to ensure that teams have clear visibility on the work and progress by sharing artifacts, common sprint reviews and open communication channels. Shared artifacts is a single product backlog visible to all teams ensuring everyone is aligned on priority to understanding overall goals. The common sprint reviews help demonstrate their work to stakeholders and obtain feedback promoting transparency that everyone is aware of the progress and challenges.

3.1.2 Inspection is essentially needed on a regular basis to identify the issues and opportunities to improve via synchronized sprint events and continuous feedback loops. Sprint planning conducted together ensuring they are aligned on the work identifying dependencies and risks. Daily scrum representing each team discussing the progress across cross team. Sprint retrospective is held to reflect their process and identify improvements addressing systemic issues and collaboration across teams.

3.1.3 Adaption is core principle enabling for process improvement continuously to changing requirements. This Is achieved through iterative development, feedback

and continuous focus on improvement, LeSS emphasizes importance of continuous improvement both on team and organization level.

The basic LeSS designed for eight teams working towards single product having single product backlog shared sprint planning and common sprint review. LeSS Huge is extension of basic LeSS designed for very large organizations with more than eight teams including additional structures managing complexity of large scale development. Product gets divided into requirement areas by separate product owner manage complexity to make team focus on specific area of product. Area prioritized product backlog is owned by area product owner aligned to overall product goals.

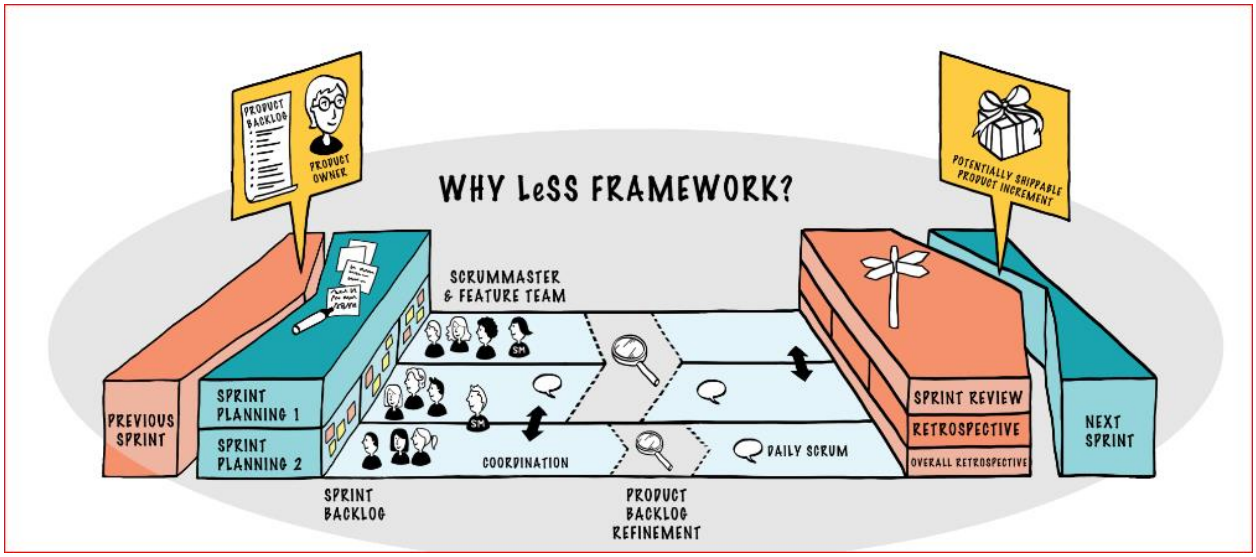

Fig 3. LeSS Framework

Though the LeSS basic or the LeSS Huge framework follows the scrum principles the only difference is the number of teams working towards the product objective changes. LeSS basic covers eight teams with one product and one goal objective to obtain for the product, whereas the LeSS Huge divides the areas of the product separately owned by area product owner ultimately with one product goal. In spite of all practices the benefits of using the framework such as enhanced collaboration, improved transparency, greater flexibility and continuous improvement.

The challenges in coordination among multiple teams can be complex and challenging, effective communication will require lot of effort and discipline, adopting new practice and framework may get resistance. Lot of training needs adopting Huge LeSS practices effectively After carefully reviewing each of the framework let's talk about the statistics on adoption rate for the different frameworks.

Table 1. Comparison of different frameworks

Aspect	SAFe (Scaled Agile Framework)	Nexus	LeSS (Large Scale Scrum)
Purpose	To scale Agile across large enterprises, integrating Lean and Agile practices	To scale Scrum in multiple teams working on single product	To apply Scrum principles in Large scale enterprise contexts
Structure	Multi level team, program, portfolio, large solution	Nexus integration team, Nexus sprint planning, Nexus daily scrum, Nexus	LeSS- 8 teams LeSS Huge – upto 1000 people

		sprint review, Nexus sprint retrospective	
Roles	Release train engineer, Product management, System Architect	Nexus integration team to coordinate multiple scrum teams	Single product owner, multiple cross functional teams
Events	PI planning, System demo, Inspect & Adapt, Scrum of Scrums	Nexus sprint planning, Nexus daily scrum, Nexus sprint review, Nexus sprint retrospective	Sprint planning, daily scrum, sprint review, sprint retrospective
Artifacts	Program backlog, solution backlog, portfolio backlog, PI objectives	Integrated increment Nexus sprint backlog, Nexus goal	Single product backlog, Definition of Done, potentially shippable product increment.
Adoption complexity	High, requires significant organizational change and training	Moderate builds on existing scrum practices with minimal extensions	Moderate to high requires deep understand of scrum and organizational change
Best Suited for	Large enterprises with complex structures and multiple product lines	Organization with multiple scrum teams working on single product	Enterprise looking for simple, scalable scrum framework

Conclusion: Each framework has strengths and challenges in terms of scalability, flexibility and implementation. SAFe is highly scalable providing a structured approach requiring significant effort customizing and implementing. Nexus has simplicity of scrum enhancing collaboration and integration having moderately scalable and easy implementing. LeSS has simplicity and effective communication with flexibility of implementation maintaining scalability of large organization. According to 15th state of Agile Report, SAFe is the most popular scaling framework with 37% respondents using SAFe with its structured approach and comprehensive nature making it suitable for complex product development needs. Nexus is used 9% as per the Agile report based on its simplicity and enhancing scrum appealing for moderate scaling needs. LeSS is used 7% of respondents as per the Agile report emphasizing simplicity and effective communication suitable for large product development needs. According to State of SAFe report ’25 72% of organizations found Scaled agile is extremely useful in their adoption. Based on the above-mentioned facts and suiting the enterprise needs the respective framework can be incorporated for better results. (Table 1 for reference)

REFERENCES

Almeida, Fernando, and Eduardo Espinheira. "Large-scale agile frameworks: a comparative review." *Journal of Applied Sciences, Management and Engineering Technology* 2, no. 1 (2021): 16-29.

Putta, Abheeshta. "Adopting Agile Methods in Large-scale Organizations using Scaling Frameworks." (2022).

Putta, Abheeshta, Ömer Uludağ, Shun-Long Hong, Maria Paasivaara, and Casper Lassenius. "Why do organizations adopt agile scaling frameworks? a survey

of practitioners." In *Proceedings of the 15th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM)*, pp. 1-12. 2021.

Turetken, Oktay, Igor Stojanov, and Jos JM Trienekens. "Assessing the adoption level of scaled agile development: a maturity model for Scaled Agile Framework." *Journal of Software: Evolution and process* 29, no. 6 (2017): e1796.

Christopher, Lauren A., and Marne De Vries. "Selecting a scaled agile approach for a fin-tech company." *South African Journal of Industrial Engineering* 31, no. 3 (2020): 196-208.

Wińska, Ewelina, and Włodzimierz Dąbrowski. "Software development artifacts in large agile organizations: a comparison of scaling agile methods." *Data-Centric Business and Applications: Towards Software Development (Volume 4)* (2020): 101-116.