Scaling Method Comparison

Large-Scale Scrum (LeSS)

LeSS was created by Bas Vodde and Craig Larman in 2005. The architects of the LeSS framework believed that Scrum was developed for smaller teams or organizations (Vodde & Larman, About LeSS, 2005). However, they needed something scalable for a bigger corporation, keeping/extending the scrums existing framework.

LeSS is a set of principles similar to Scrum but just scaled up for a large organization. There are two types of LeSS frameworks. (i) smaller LeSS and (ii) LeSS Huge (Vodde & Larman, LeSS Framework, 2005)

Roles

1. Teams

LeSS can have multiple teams under the same product owner. However, LeSS recommends having eight teams under the same product owner (Vodde & Larman, LeSS Framework, 2005). If there are more than eight teams, increase the number of product owners.

2. Product Owner

LeSS excludes the product owner from participating at a detail level and gives the teams more responsibility to figure out a solution (Lecture Notes)

3. Area Product Owner

When there are more than eight teams, maintaining those eight teams falls under the area product owner, and the product owner watches a team of area product owners. (Vodde & Larman, LeSS Huge, 2005)

Processes

Most of the processes remain the same. Sprint Planning, Sprint Review, and Sprint Retrospective remain the same across all teams. There are some additions to the traditional scrum framework. (Vodde & Larman, LeSS Framework, 2005). Here is the process for the LeSS framework.

1. Sprint Planning 1

The product owner defines the backlog for up to 8 teams. Instead of the regular sprint planning stage in the agile framework, LeSS asks for two sprint planning meetings. In the 1st Sprint Planning, the teams meet, self-manage, and divide the product backlog items. Teams also find places where there is an opportunity to collaborate. (Vodde & Larman, LeSS Framework, 2005)

2. Sprint Planning 2

Sprint Planning 2 follows the sprint planning 1 meeting. This stage is independently by each team (and usually in parallel) to discuss the backlog on a team level. If teams colloboarte on delivering the same product, they might be in the same room. (Vodde & Larman, LeSS Framework, 2005)

3. Daily Scrum

These meetings are held daily and independently by each team. Although a member can observe other teams daily Scrum to increase information sharing (Vodde & Larman, LeSS Framework, 2005)

4. Overall Product Backlog Refinement

There may be an optional Product Backlog Refinement meeting that includes product owners from all the teams. (Vodde & Larman, LeSS Framework, 2005)

5. Sprint Review

After a sprint, a general review meeting under the product owner helps to showcase the product. It can include all the relevant members and stakeholders. LeSS recommends a “bazaar” or a “science fair” environment in a large room where teams can show and demonstrate their product in their iterative process. (Vodde & Larman, LeSS Framework, 2005)

6. Team and Overall Retrospective

Sprint Retrospective is one meeting where all the teams come together and explore ways to improve the overall system and or process. The maximum duration of this meeting is 45 minutes. This meeting should include the Product Owner, Scrum Master, and rotating members from each team. After the team retrospective, a team retrospective is held to do the same but on a team level. (Vodde & Larman, LeSS Framework, 2005)

Artifacts

1. Daily Sprint

Daily sprint stays the same as dictated by the agile and Scrum framework

2. Sprint Planning

Sprint Planning is divided into Sprint Planning 1 and Sprint Planning 2. Sprint planning 1 deals with the product backlog on a production scale. All teams get to self-organize and claim the work that the team can handle. Teams also find collaboration opportunities among them to work the product together.

Sprint Planning 2 is independently hosted by each team to prioritize the stories. If a team collaborates towards a product, they can have representation in this meeting.

3. Sprint Review

Sprint Review includes the Product Owner, Stakeholders, and end-users. Sprint Review is where teams talk and demonstrate the work done in that sprint. Product Owner and Stakeholders can refine the backlog at this stage.

4. Sprint Retrospective

Teams representatives use this forum to discuss the opportunity and shortcomings on the sprint to design and work towards improvement in the next sprint. Teams also have an independent Sprint Retrospective to improve and provide feedback for the next sprint.

NEXUS

Nexus framework was created by Ken Schwaber, the co-creator of the Scrum framework, in 2015. The Nexus framework is based on the scrum framework and uses their iterative and incremental approach and scaled to work in large organizations (Schwaber & Bourk, 2015)

As mentioned earlier, Nexus extends the scrum framework to multiple scrum teams working towards a single product backlog. Nexus recommends three to nine teams. One product owner maintains the backlog, and each team may have a shared or dedicated, part-time or full-time scrum masters to drive the product backlog. The ideal size of this framework is less than 80 members (Lecture Notes)

Roles

1. Product Owner

There has to be one product owner maintaining the product backlog.

2. Scrum Master

Scrum Masters can be full-time members or half-time members of the scrum process. The scrum master can handle multiple teams if they don't have a dedicated scrum master.

3. Nexus Integration Team

Nexus Integration Team is a mediator among all the teams. They coordinate all the integration issues that may exist, blocking the delivery of the end product.

Processes

1. Product Owner

The product owner is a dedicated owner of the product backlog, as in Scrum, to maintain the product delivery. In addition, they define the view and scope of the product.

2. Teams and Scrum Master

Teams and Scrum Master work towards the backlog, with the scrum master being the servant leader as in Scrum or Agile framework.

3. Nexus Integration Team

Nexus Integration Teams consists of the Product Owner, a Scrum Master, and Nexus Integration Team. This team is responsible for maintaining and handling, communicating, and working towards the integration issues.

Artifacts

1. Nexus Team Integration

Nexus Team Integration is a team of Product Owner, Scrum Master, and Nexus Integration Team Members. This team works towards eliminating and communicating any integration issue that the team may or might face in delivering the product in the current sprint. (Lecture Notes)

2. Nexus Daily Scrum

Like the Daily Stand-Up in Scrum, Nexus Daily Scrum meets daily to highlight any concerns and discuss the work being done for the current sprint. When the Nexus Integration Team highlights an issue, individual teams work towards solving that issue and being a new one, if any, to the next Integration Team meeting. (Lecture Notes)

3. Nexus Sprint Review

Nexus Sprint Review is similar to the sprint review in Scrum framework. At the end of the sprint, teams provided feedback to improve the process. The product is demonstrated in the Nexus Sprint Review for the end-users, stakeholders, and product owner. The Product Owner can refine the Product Backlog at this stage. (Lecture Notes)

4. Nexus Sprint Retrospective and Refinement

Representatives from each team meet and share the challenges faced in this sprint and provide bottom-up intelligence in dealing with these challenges. In addition, teams are responsible for conducting their independent retrospective and refinement meet. (Lecture Notes)

DISCIPLINED AGILE DELIVERY (DAD)

Disciplined Agile (DA) is a toolkit built on top of the Scrum framework, and this toolkit enables Scrum to be used in a large organization. DAD is a learning-oriented hybrid agile approach for IT solution delivery (Project Management Institute, n.d.). The Project Management Institute acquired DA in 2019 (Pradeep, 2019)

Roles

1. Product Owner

Each DAD Team has a single product owner who looks at the project backlog and defines the view and direction of the end product. In addition, the product owner represents the opinions of the stakeholders and end-users to develop stories. (Project Management Institute, n.d.)

2. Architecture Owner

The architecture owner is the source of project risk responsible for mitigating this risk. Teams may or may not have a minor role in architecture risk. Typically this role goes out to the senior developer in the team. This role is not a hierarchical role to which the team reports. This person must be technically capable of mitigating those risks. (Project Management Institute, n.d.)

3. Team Lead

Team lead is similar to the role of a Scrum Master in the traditional Scrum framework. They empower the team to be self-organizing and self-optimizing. The team lead should be the one to remove all the impediments in the process. (Project Management Institute, n.d.)

Processes

1. Product Backlog

Management defines the vision, funding, roadmap, and guidelines for the product to be delivered. The product owner then determines the backlog and prioritizes stories to work. (Project Management Institute, n.d.)

2. Release Planning

The product owner, team lead, and team members ask for requirements and design the release plan for the stories in the backlog. The architect owner defines the risk for the release plan for the team. (Project Management Institute, n.d.)

3. Sprint Iteration

The team picks up the high-priority items on the backlog and works towards a potentially shippable product at the end of each sprint. (Project Management Institute, n.d.)

4. Daily Huddle

The teams meet daily to discuss the work and highlight any impediments that might affect their work. (Project Management Institute, n.d.)

5. Release

The product is packaged and released to the solutions frame at the end of each sprint, and support teams are notified about the change. (Project Management Institute, n.d.)

Artifacts

DA provides various options for the team to choose from. These options cover the whole organization in architecture, design, testing, programming documentation, and more.

This toolkit is based on the Scrum framework and therefore derives the most from it.

Scaled Agile Framework for Enterprises (SAFe)

Dean Leffingwell created the SAFe framework to work for a large organization. In addition, this framework was designed to be scalable. The core of this framework is the Scrum framework and Lean framework (Leffingwell, 2021).

Roles

In addition to regular roles of the Scrum framework

1. Scrum Master

The Scrum Master's role is the same as in the Scrum framework. In addition, Scrum Master serves as the servant master in the product delivery life-cycle.

2. Product Owner

The Product Owner is in charge of maintaining the backlog on the team level. They define the view and guidelines for the product to be released.

3. Release Train Engineer

The release train engineer is responsible for program-level deliveries. In addition, they also act as a servant leader at the program level.

4. Program Manager

The Program Manager role is similar to the Product Owner role but on a program level.

5. Value Stream Engineer

The Value Stream Engineer is in charge of the Value Stream level deliveries. In addition, they act as the servant leader at the Value Stream level.

6. Solutions Manager

A Solutions Manager acts as the product owner at the Value Stream level.

Portfolio Management

The Portfolio Management team helps dictate the steams to be worked on. Portfolio Management uses dictates the teams use strategic themes and allocate the budget.

Processes

There are four levels in the SAFe framework, starting from the bottom:

1. Team

The team level is where the work gets done. Teams can use the Scrum framework in Kanban to organize the work. There is a scrum master to remove any impediments. The Product Owner defines the vision and product backlog. Teams work in an iterative process to deliver the solutions. (Lecture Video)

2. Program

Multiple teams align their deliveries to the vision and backlog dictated by the Program Manager. Program level works on a similar iterative process called Agile Release Train (ART). The ART is five iterations long. The 5^{th} iteration is where the teams discuss future opportunities and process improvements. The ART works in continuous exploration, integration, and development cycles. (Lecture Video)

3. Value Stream

Value Stream can define multiple AR trains to deliver, working on a more extensive product release than program level. A Value Steam Engineer handles each Value Stream, and the Solution Management defines the backlog. (Lecture Video)

4. Portfolio Management

Each portfolio management team handles various value streams. The portfolio management team defines strategic themes and allots the budget for the final product. Each solution becomes an epic in the story. (Lecture Video)

Artifacts

1. Program Iteration

At the program level, there are Program Iterations (PI). These work similarly to the sprint iteration in the traditional Scrum framework. (Lecture Video)

2. Agile Release Train (ART)

Five PIs become an ART. The 5^{th} PI is added for the exploration phase. Some common examples are hackathons, code sprints, etc. (Lecture Video)

References

Leffingwell, D. (2021, February 10). *Welcome to Scaled Agile Framework 5!* Retrieved from SAFe: https://www.scaledagileframework.com/about/

Pradeep, D. (2019, August 9). *Project Management Institute Announces Acquisition of Disciplined Agile*. Retrieved from Project Management Institute: https://www.pmi.org/about/press-media/press-releases/project-management-institute-announces-acquisition-of-disciplined-agile

Project Management Institute. (n.d.). *Disciplined Agile*. Retrieved from PMI: Disciplined Agile: https://www.pmi.org/disciplined-agile

Project Management Institute. (n.d.). *Full Delivery Life Cycle*. Retrieved from PMI: Disciplined Agile: https://www.pmi.org/disciplined-agile/process/introduction-to-dad/full-delivery-lifecycles-introduction

Project Management Institute. (n.d.). *People First: Roles in DAD*. Retrieved from PMI: Disciplined Agile: https://www.pmi.org/disciplined-agile/process/introduction-to-dad/people-first-roles-in-dad-introduction

Schwaber, K., & Bourk, S. (2015). *An Introduction to the Nexus Framework*. Retrieved from Scrum.org: The Home of Scrum: https://www.scrum.org/resources/introduction-nexus-framework

Vodde, B., & Larman, C. (2005). *About LeSS*. Retrieved from Less: More with LeSS: https://less.works/resources/about

Vodde, B., & Larman, C. (2005). *LeSS Framework*. Retrieved from LeSS: More with LeSS: https://less.works/less/framework?setlang=true

Vodde, B., & Larman, C. (2005). *LeSS Huge*. Retrieved from Less: More with LeSS: Vodde, Bas; Larman, Craig