Scaling Agile: Fractals of Innovation

An excerpt from “Agile Business” by Bob Gower and CA Technologies

Scaling Agile: Fractals of Innovation

By Ronica Roth

"Sure, agile works for small organizations, but we've got to organize hundreds (or thousands) of people on four continents and agile just won't work for us." Sound familiar? It does to CA Agile Central coaches. It's something we hear all the time, and yet very large companies—all of whom acknowledge that something is wrong with the traditional way of doing things—continue to come to us for help.

The challenge we face is not how to create success with their development teams, but how to scale agile across all teams. This is particularly difficult because we must factor in the many different things done in the organization, including product roadmapping, funding, system architecture, and release planning.

Sound daunting? The good news is you needn't reinvent it all for your organization. An excellent starting point has been developed in the Scaled Agile Framework®, released by Dean Leffingwell. SAFe®, as it's called, has been tested in many large enterprises, including some of CA Agile Central's biggest customers.

As with existing agile frameworks like Scrum and XP, you'll want to start with the core prescriptive elements and adjust for your environment as you learn and change.

Scaling by Fractals

Scaling agile means that we apply its principles to large, even very large, groups of people. When we do this, we allow those people to be more connected to their work and its impact, despite being part of a huge system.

This process is effective because we scale agile by fractals, meaning we create similarly shaped structures at different levels of scale throughout the organization. And rather than build bigger teams, we add more small, cross-functional teams and stitch them together into a larger whole. This way, the team remains the core unit and owns its working agreements, information radiators and policies.

As you might expect, it's not quite enough to simply add teams. Those teams will need leadership to guide them, a structure to align them, and information to enable agile's continuous improvement.

Cadence and Synchronization

In order to effectively coordinate work with many teams, you must look three to six iterations down the line. And to do this effectively, your teams need to have worked together for a while. These mid-range plans, sometimes called release plans, need not be perfect of course, but by creating a certain level of predictability, you remove the risk from coordination and delivery.

Once we have a team with a cadence of iteration and release planning, we add several agile teams, each between five and 10 people. The number of teams depends on who is needed in order to build and deliver a system of value. This helps determine how many teams make up the program.

While it's usually obvious which individuals make up a program, some organizations struggle with creating smaller, cross-functional teams. If you find yourself in this situation, provide your employees with the boundary conditions and let them divvy up the teams together.

Those teams must now coordinate their work in two key ways: by ensuring their delivered work adds up to valuable features, and by managing dependencies between them. Teams achieve this by synchronizing their cadences, and planning and retrospecting together.

With every team running tested software by the end of each iteration, we have a clean synchronization point. This system isn't perfect, but it's far easier than systems used in traditional phased development.

As for ensuring all the work leads to value, the key is to host a joint release-planning event that includes all the members of each team. This event is designed as a way to check on each team's plans and make important adjustments.

SAFe describes the Scrum release timebox as a potentially shippable increment, or PSI. Even if your group will never release software so frequently, this midrange planning horizon is critical to quality, coordination and success.

Support Structures

Synchronized teams still need supporting roles and structures beyond themselves in order to coordinate effectively and deliver value. SAFe

defines several of these, which are outlined below. But, you'll need to define the right structures and roles for your own organization.

Building the right thing: How can all those team-level Product Owners (POs) make sure their backlogs of stories add up to value? Product Managers (PMs), who own the backlog of Features at the Program level, meet regularly with Product Owners to ensure shared vision and understanding. This way, POs can make adjustments in their own backlogs at a later date, confident they continue to support the shared vision.

Building the thing right: The System Team (ST) owns care and feeding of the whole system, while the Release Management Team (RMT) ensures the success of each release. Architecture is a first-class citizen in SAFe, and a program roadmap of features also includes architecture deliverables.

Building the whole thing: Among other structures and roles, SAFe adds the so-called Release Train Engineer (RTE)—an "uberScrumMaster" who helps teams coordinate deliveries, communicate, escalate problems, manage risk and continuously improve at the program level. The RTE helps keep the train on the tracks. In your organization, these people might be called project managers, program managers or development managers.

The Role of Leadership

Strong leadership—not management—is essential to helping agile work at this scale.

To lead, on one level, is to provide a clear vision around what the program is trying to achieve. This vision gets created, and can be trusted, in part by injecting agile discipline and visibility at the portfolio and program levels. That means, for example, funding fewer initiatives and limiting work in process rather than filling development queues with wishful thinking.

We apply similar rigor to steering decisions. Agile portfolio management emphasizes steering over planning. Having funded an initiative, and seeing the early results of execution, the portfolio council decides—on a cadence—whether to continue, to adjust, or to kill each initiative. Those decisions tell the teams that their learnings are appreciated and heard.

The whole system, from top (portfolio) to bottom (team) and from left (business) to right (DevOps and support) is ultimately about learning and creating feedback loops.

To facilitate innovation and performance, leaders from the very top of the organization must build a culture of learning.

At the team and program levels, organizational leaders are responsible for coaching their people on the tools of continuous improvement and encouraging them to relentlessly improve. For it's only when employees see the vision for their work and have all the necessary information that they'll feel truly empowered to act.

Connect with CA Technologies at ca.com

CA Technologies (NASDAQ: CA) creates software that fuels transformation for companies and enables them to seize the opportunities of the application economy. Software is at the heart of every business, in every industry. From planning to development to management and security, CA is working with companies worldwide to change the way we live, transact and communicate—across mobile, private and public cloud, distributed and mainframe environments. Learn more at ca.com.

For more information, please visit **ca.com/ca-agile-central**