

Managing an agile portfolio

When the right people on the right teams have the right context, they naturally do the right thing.



BY DAN RADIGAN

BROWSE TOPICS

- Agile manifesto
- > Scrum
- > Kanban
- > Agile project management
- > Product Management
- ✓ **Agile at scale**
 - Overview
 - Agile iron triangle
 - Using Improvement Kata to support lean
 - The Large-Scale Scrum (LeSS) framework
 - Beyond the basics whitepaper
- > Software development
- > Design
- > The agile advantage
- DevOps
- > Agile Teams
- > Agile tutorials
- > About the Agile Coach
- All articles

Can agile practices work across a large portfolio of many teams and lots of developers? Absolutely. Netflix coined the phrase "highly aligned, loosely coupled" to describe well-tuned agile development across a large organization. Let's look at the role of context and autonomy, as well as some common characteristics found among high-functioning agile portfolios.

Set the right context

With so much focus on the team, senior management in a large-scale agile organization are often left wondering what their role is. As drivers of strategy and direction, senior leadership can be key champions of agile culture for the whole organization. For agile development across a portfolio of products, strategy and direction manifest as "themes": large areas of focused work, defined over a period of time. Themes help multiple teams working on one business initiative to collaborate effectively and deliver on the organization's goals.



Senior leaders of successful agile portfolios also foster a culture of transparency and openness.

A transparent culture surfaces issues without fear of retaliation, and it minimizes the negative aspects of company politics. As a result, it's easier to find the right solution and keep teams moving forward.

Expand agile practices across the organization

The most successful companies running agile at scale share three common traits. First, the entire [program](#) is iterative. Traditional portfolio management is focused on top-down planning with work laid out over long time periods, but agile portfolio management takes the concept of build-measure-learn cycles used by individual agile teams and applies it on a larger scale. [Long-term agile](#)

RELATED READING

[Lean Portfolio Management](#)

[Learn more →](#)

SUBSCRIBE

[Sign up for more articles](#)

Email

[Subscribe](#)



WHITEPAPER

[Beyond the basics of scaling agile](#)

Learn the new rules to transformational agility.

[Download the whitepaper →](#)

planning is carried out by teams working together, using modular design, and sharing findings on a regular cadence. And they're empowered to make trade-offs in the scope of work, the timing of its execution, and the resources allocated to get the job done—key elements on the [iron triangle of planning](#). This results in tremendous flexibility, which shifts the focus from continuing to execute on an inflexible plan to delivering value and making tangible progress according to business strategy and goals.

Second, successful organizations communicate across the portfolio. They share knowledge and break barriers between organizational silos. Similar to agile ceremonies on the [team](#) level, context needs to be shared constantly throughout the organization so that goals, progress, and stumbling blocks are transparent for everyone. This fosters respect between teammates and coworkers alike, regardless of role within the organization, and encourages interactions that are rooted in empathy and understanding.

Third, the most agile organizations make frequent [releases](#) across the portfolio, even if a release involves the work of multiple programs. Aligned sprint cycles, work invested in strong APIs and technical decoupling, and an efficient automated [testing](#) and deployment pipeline all guarantee constant visibility into who is shipping what and when.

Share one vision, but embrace diversity

As in traditional agile development, work in portfolio agile development is delegated to [teams](#) rather than individuals. Each team understands the organization's larger goals, and each team also develops a vibrant culture that optimizes its own processes and delivery.

For example, story points are a common way teams [estimate](#) work, and use a set of values based on the Fibonacci sequence (0.5, 1, 2, 3, 5, 8, 13, 20, 40, 100). Team A's idea of what an 8 means will probably differ from team B's. For this reason, senior management shouldn't measure teams by numeric velocity alone. (Numeric velocity is the amount of story points a team can complete within a sprint.) They must understand that each team's velocity will be unique because each team calibrates story point values differently.

Likewise, agile teams have different release cultures. Scrum teams usually release software at the end of each sprint, while kanban teams release continually or when the [product owner](#) requests a build be pushed to production. One of the big challenges in agile portfolios is releasing large quantities of code at once—or rather avoiding that. No one wants releases that require

rather, avoiding that. No one wants releases that requires the entire engineering staff on deck, after all. Separating code into independent release streams by using modular, rather than monolithic, design, goes a long way in terms of flexibility and autonomy among business units. Modular design also reduces risk in each release because less code is changing with each release. That makes it easier to diagnose and fix problems afterwards.

Autonomy also extends into workflows for portfolio organizations. Teams that work in different parts of the business may have unique [workflows](#). It's a good bet that a software engineering team will have a different workflow and process than a marketing team, even when both departments follow agile principles like iterative development and regular retrospection. Or two development teams may choose to divide their workflow into different states. And that's ok! This diversity gives large agile portfolios the benefit of shared knowledge. **Trying more things means you're learning more things that can be shared across the organization.**

Expand agility as you grow

An agile framework for a large portfolio means scaling agile principles used at the team level across a whole organization. Agile culture is a force multiplier: it naturally scales upward and outward when its core principles are followed and shared. But the portfolio will only be as successful as its weakest team. To ensure success, senior leadership must work in partnership with all teams to build a healthy agile culture.

Ready to take agile adoption to the Portfolio level? Learn how Enterprise Agile Planning platform, Jira Align, helps improve visibility, strategic alignment, and enterprise adaptability in order to accelerate your digital transformation.

[Learn more](#)

SHARE THIS ARTICLE



DAN RADIGAN

Agile has had a huge impact on me both professionally and personally as I've learned the best experiences are agile, both in code and in life. You'll often find me at the intersection of technology, photography, and motorcycling.



TUTORIAL

Learn how to use epics in Jira Software

Use these step by step instructions to learn how to use and create epics in Jira Software

[Try this tutorial →](#)

ARTICLE

Long-term agile planning - 8 steps to get started

Learn how to build great software using long-term agile planning.

[Read this article →](#)

Agile Topics

Agile project management

Scrum

Kanban

Design

Software development

Product management

Teams

Agile at scale

DevOps

Sign up for more agile articles and tutorials.

Email

[Subscribe](#)



Up Next

[Lean portfolio management →](#)