



Managing the “Big Rock” Projects

By Mimi Hall

Many of the typical principles of program management apply when you’re facing a major, cross-functional effort—the “big rocks” that you must move. Let’s consider a few ideas for how to modify your approach in an Agile organization.

In Agile organizations, the ideal is that teams sprint to deliver new or enhanced functionality. They do this iteratively, getting quick feedback from the market and ultimately enabling a product vision.

Time-bound, scope-constricted, and resource-constrained “projects” and “programs” (or really big projects) are terms of the past for Agile evangelists, and often have the connotation of a four-letter word.

In Agile teams that are long-standing and organized to deliver on specific product value propositions,

we don’t typically see “normal” project management processes. We don’t assemble teams on demand. We don’t attempt to capture all requirements at the start of an effort.

We’re kidding ourselves, though, if we think businesses will stop having discrete needs to be managed by at least some project or program manager type roles.

Typical Agile roles (see sidebar) focus on priority, value, and productivity of the team. Let’s call these focus areas “little rocks.” Coordination roles, such as release

train and value stream engineers at higher levels of the organization, generally focus on getting those “little rocks” through the system, while managing dependencies on other teams or teams of teams.

So, what happens when you have a “big rock” that you want to build? Who pays attention as that goes through the “system”?

Adept program managers can serve this role. They can mind the “big rocks.” A few “big rock” circumstances may require your organization to adopt program management:



Agile Team Roles in Brief

An Agile or scrum team is comprised of:

A scrum master focused on team productivity and optimizing throughput of work through the team (i.e., the “system”).

Product owners representing the voice of the customer on the team, and prioritizing work based on the most value delivered.

Team members doing the work, showcasing it, learning from it, and adjusting to user and market feedback.

- When you require a large, time-bound upgrade across platforms—and it must happen while the regular work continues.
- When changes in the regulatory environment demand compliance across products, with a centralized group across teams responsible for managing the details and providing progress updates.
- When a custom request from an important customer demands a special team to deliver the results. A note about this, however: If it happens often and the exception becomes the norm, you may consider reorganizing teams to deliver on specific customer value statements.

What in Program Management Stays the Same from Waterfall to Agile?

When serving as an Agile program manager, many traditional program management principles still apply, including:

- Aligning program management practices with organizational culture, while balancing program deadlines and objectives.
- Defining success metrics; establishing processes to measure progress toward the desired business outcomes.

- Providing consistent documentation across teams and work streams.

What Changes in an Agile Organization?

While some aspects of traditional program management apply to the “big rocks,” Agile program managers will likely need to make a few adjustments to succeed with those efforts.

Pace of requirements and software development: Gone are the siloed days of the typical software development life cycle. Work with your sponsors to define the success metrics at the highest level of the program. Defer lower-level decision-making to those who have the information about how best to deliver on those desired outcomes.

Frequency of cross-team and stakeholder interaction: While it may not be at the same cadence of development team sprint demos (typically every two to four weeks), challenge yourself and your team to integrate across workstreams. How often can those integrations result in demos to end-users to solicit feedback and update your backlog(s) of work accordingly?

Vocabulary: Work items may be called “stories, features, and epics.” And the change requests of yesterday are now called “enhancements.” Learn the vocabulary of the Agile teams you’re

working with and ensure everyone is speaking the same language. You can serve as a key translator to business partners and executive stakeholders to understand the size and complexity of the work at hand.

Level of documentation (and possibly where it’s stored): Negotiate the “right level of documentation” for your program. Ensure you have the visibility you need into work that is in progress or completed. This may not be in your typical intranet or application life cycle management tool. Instead, it may look like a virtual whiteboard of sticky notes moving across. If you are truly empowered to successfully manage the program at hand, you will have the power, access, and influence you need to ensure consistent data across teams so you can track the “big rock” to completion.

The bottom line: Program management is required when a big effort crosses workstreams. When capable coordination across workstreams, and vertical and horizontal communication are vital for your success, consider Agile program management.

Mimi Hall

mimi.hall@jabian.com