

RG Perspective

Program Management for Agile

Transitioning Toward a New Organizational Paradigm



11 Canal Center Plaza
Alexandria, VA 22314
HQ 703-548-7006
Fax 703-684-5189

teamRG.com
© 2017 Robbins-Gioia, LLC

For more than a decade, Agile methods such as Scrum, XP, and Kanban have proven very effective for software development projects.

In spite of their immense popularity and applicability, however, many organizations—particularly those with traditional management structures—still have difficulty implementing Agile. As is true for many difficult implementations or complex projects, program management (PgM) is the best way to systematically and successfully achieve desired outcomes. Program management for Agile projects—including non-IT projects—combines two powerful approaches for optimum results.

At first, PgM might seem antithetical to Agile. In fact, the framework of PgM—with some nuanced modifications—ultimately will ensure a more successful Agile implementation in your organization. The fundamental challenges to using Agile on larger-scale efforts mirror those of any large program and include planning, contracting, staffing, reporting, and organizational readiness.

- **Planning:** Establishing a vision, creating the value proposition, defining requirements, and involving the customer are, if anything, more important with Agile. Change is needed in the expectations, management, and working environment to enable collaboration using a lighter but more continuous and engaged approach.
- **Contracting:** The federal government presents some unique obstacles to the application of Agile management, given the need for adherence to outmoded contracting regulations and product development standards.
- **Staffing:** Agile teams require very specific staffing. The model depends on small matrixed teams made up of highly motivated, poly-skilled, collaborative professionals accomplished in their fields and communications techniques.
- **Reporting:** Agile methods use different metrics to report progress that must be learned and incorporated for sufficient management visibility. On Agile projects, since the product is being delivered rapidly in small batches, the best way to measure progress and maintain

intelligent control is through the use of metrics that are focused on business outcomes, not process or product outputs.¹

- **Organizational readiness:** Making the switch to Agile is a big change for many organizations. And it must be said that not every organization—or project—is suited to Agile.

Before discussing how program management can help you meet these challenges, it is recommended that you choose a pilot project to keep the challenges to a manageable size and gather lessons learned. While all projects and functions can benefit from *elements* of Agile—increased communication, accountability, closer collaboration, and the like—Agile methods may not be a fit for every project or organization. Table 1 provides guidance on when and when not to use Agile.

PgM for Agile can also be accomplished for pieces of a larger project as well as an individual one. At a project level, in general, good candidates for an Agile “piece” of a larger project exhibit these characteristics:

- Can be completed in six weeks or fewer or decomposed into modules that offer a discrete, usable product
- Has well-defined and -understood interfaces
- Lends itself to incremental testing or inspection to ensure quality
- Can be addressed by existing, available, dependable organizational resources.

¹ Augustine, Sanjiv, and Arlen Bankston, “Transitioning to Agile Project Management; A Roadmap for the Perplexed,” *Cutter Consortium Agile Product and Project Management Executive Report* 9 (2008): 1.

Program Management for Agile: Transitioning Toward a New Organizational Paradigm

Table 1. When and when not to use Agile

Use Agile When...	DO NOT Use Agile When...
<ul style="list-style-type: none"> • Qualified, well-seasoned, well-disciplined team members will be staffed. • Interfaces/interfittings are well-defined. • Deliverables can be reasonably distributed in work packets achievable in 2-4 week periods. • Responsiveness to customer requests defines success. • The customer or customer representative is available for close collaboration throughout the project. • Scope can be adjusted to fit schedule. • The customer is able to reprioritize and add requirements as they progress. • Work is ground-breaking with steps defined by progress resulting in estimates that are not expected to be reliable. • Incremental results have significant value. • Process is by nature iterative, allowing for cumulative results though sprints, and for overlapping planning for the next sprint while work takes place in the current sprint. 	<ul style="list-style-type: none"> • Work is commoditized and can be delivered by lower-skilled staff. • Inexperienced individuals are principal candidates. • Deliverables cannot be reasonably distributed in work packets achievable in 6 week periods. • Success is defined as absolute adherence to fully scoped customer requirements identified completely in advance of development. • The customer is <i>not</i> available for close collaboration throughout the project. • Critical (or many) steps involve long lead times or lots of specialized resources. • Incremental results have little or no significant value for anyone. • Process is best implemented as linear (waterfall model in software) or spiral. • Contract requirements or regulations mandate the use of a specific life-cycle process and attendant documentation that does not follow Agile tenets, and under which no ready allowance can be made for an Agile approach.² • No contract requirements or regulations preventing the use of Agile or making its employment impractical apply.

² Kevin Thompson, "When to Use Scrum for software projects," *Cprime*, <http://www.cprime.com/community/articles/whentousescrum.html>

Planning

Traditionally, the project manager attempts to have the project scope and detailed requirements defined and approved early in the life cycle and then builds a detailed project plan to support the scope, schedule, and budget constraints. Indeed, in many instances the project manager is provided the scope and high-level requirements without his or her participation. In any case, the approved scope and high-level requirements result in a fixed set of requirements; the consequent project plan results in a delivery date promised to the customer—or the project sponsor and/or owner in an Agile project.

When considering an Agile approach, a project manager must take into account the ability of the customer to define requirements and adapt them as needed. For knowledge-based projects, requirements can change rapidly based on shifting project and organizational contexts. The project manager needs the ability and authority to realign requirements to ensure deliveries continue to meet strategic objectives. The Agile mindset fixes the schedule and to some extent resources, leaving requirements to be determined by a collaborative team including the customer.

In practice, trying to fully define requirements early results in a significant waste of time, energy, and resources. A Standish Group study indicates that as much as 60 percent of the functionality developed in software-intensive projects is never or rarely used. The Agile approach, with its emphasis on continual planning synchronized with customer-driven priorities, often results in achieving most of the value at a fraction of the cost by eliminating functionality that customers might include in case it might be needed in the future. The shift from requirements-driven to time-driven is one of the fundamental planning reorientations for traditional project managers.

Contracting

Before an organization attempts to roll out Agile methods, it is wise to see first if this approach can meet contracting requirements. Administratively, because it requires adherence to contracting regulations and product development standards, the federal government is a special case. Most

software to be developed for military use, for example, requires different formal documentation and processes than those associated with an Agile project. At a minimum, to engage in Agile a third-party product developer would need to:

- Obtain from the Contracting Officer's Technical Representative (COTR) an agreement stating that Agile is an acceptable approach to be used on the project.
- Receive a waiver for each waterfall document required by the applicable standard for product development, but not delivered.
- Submit a document detailing the product development cycle to be employed and identifying the controlling documentation to be produced.

For its part, the government should consider:

- Indefinite-delivery indefinite quality (IDIQ) contracts, which support a modular, Agile-style product development.
- Multiple-award task order contract frameworks to promote contracting for Agile development modules, coupled with competition among a small group of contract holders for each module. (This would require architecture/integration support either through the government or a contractor.)
- Allowing task orders to be priced on a time-and-materials basis.
- Using past performance extensively rather than upfront performance or other requirements in making future task order awards.
- Eliminating requirements in the task orders for Agile projects, using objectives instead.
- Allowing a simple, Agile-style change order process.³
- Using rolling contracts that support short development iterations.⁴

³ Steve Kelman, "Unraveling the agile development knot," *FCW*, Sept. 20, 2012, <http://fcw.com/blogs/lectern/2012/09/agile-development.aspx>

⁴ Allan Kelly, "Agile Contracts," *InfoQ*, Feb. 8, 2011, <http://www.infoq.com/articles/agile-contracts>

- Alternatives to statements of work such as employing performance-based contracts and statements of objectives; the latter focuses on outcome and how to get there as opposed to building a system.⁵

Staffing

Having the right people is the most critical requirement for success. An Agile project or Scrum team requires five or six individuals who are mature, motivated, and skilled in their professions and interpersonal relations. (Scrum is a term used when a rugby game is restarted after a minor infraction; in product development, it refers to the daily 15-minute stand-up meetings the team holds each morning of its 2-to-6 week sprint, or work package. Scrum itself is likened to rugby in which the ball is moved more or less continuously up and down the field.) This team is led by the Scrum master, who either acts as or interfaces with the program manager. The team also includes the product owner, who represents the client. This person must know what needs to be built and in what sequence—and have the ability and authority to include, exclude, and implement requirements. Team roles are shown in table 2.

It is advisable to have a program manager versed in Agile—from either inside or outside the organization—to run the initial project. Failing that, the organization can look for a program manager who has led previous projects with elements of Agile and capitalize on the knowledge gained to apply to the new project. For example, one organization “unknowingly” practiced Agile in the development of a website.

The work was accomplished iteratively in short periods of time, with a workable version of the website as a result of each work segment. Beginning with an empty concrete shell, another organization stood up a highly complex command, control, and communications center in three months. In both

cases, frequent communication, collaboration, motivated and skilled professionals, and a solid PMO were key to successful delivery of these projects.

Thus, it is important that managers do not discount valuable skills and experience they might have that can transfer to Agile for program management. In particular, it is important that they understand that their leadership skills are of tremendous value on Agile projects.⁶

But it is also important to note the differences between the traditional and Agile PMO. Table 3 below contrasts their activities. Again, the best scenario for implementing PM for Agile in your organization is to have one or more individuals skilled in both.

Reporting

Managers can help their teams develop business outcome-based metrics by focusing on the customer’s definition of value, which in Agile involves quality and progress.

Quality

While Agile methods promote quality through frequent interaction and communication with the customer, ensuring that what is desired is what is built, there is a need for objective, recordable, reportable measures of quality. One metric that lends itself to mathematical objectivity is the number of test cases measured against number of open defects discovered⁷ (both in-sprint and following release.) This may be expressed as:

$$\text{Quality} = (\text{No. Open Defects}) / (\text{No. Test Cases})$$

This measure reports on the quantity of corrective work to be done, as well as the progress of the corrections. In terms of actions, this metric indicates where test cases may require reengineering.

⁵ Jared Serbu, “CIOs cite struggles with agile development,” *Federal News Radio*, May 27, 2011, <http://www.federalnewsradio.com/697/2399940/CIOs-cite-struggles-with-agile-development>

⁶ Augustine and Bankston, p. 2.

⁷ Jochen Krebs, *Agile Portfolio Management*, (Redmond, Wash.: Microsoft Press, 2008).

Program Management for Agile: Transitioning Toward a New Organizational Paradigm

Table 2. New roles for stakeholders in Agile for PgM

Scrum Master (Program Manager)	Product Owner (Customer Representative)	Scrum Team (Product Developers)
<ul style="list-style-type: none"> • Protects the team by making sure they don't overcommit. • Ensures the team lives by values and practices of project-oriented Scrum. • Facilitates the daily Scrum. • Is responsible for working to remove obstacles that come out of the daily Scrum. • On non-software projects, maintains the deliverables project schedule, ensuring that what is to be delivered is worked on in order. 	<ul style="list-style-type: none"> • Represents the client organization or acts as the principal interface to the client or, if the work is being done internally, someone in a key management role. • Provides support in removing obstacles. • Commits to not throwing new requirements or project directions at the team during a sprint.⁸ 	<ul style="list-style-type: none"> • Plans and manages each step of the project. • Works in an iterative and incremental style in close collaboration with a representative of the business/customer to understand the detail of the next step and create and validate an evolving solution. • Assumes the responsibility for determining how to best achieve the product goals (as established by the product owner). • Collaboratively decides which person should work on which tasks, which technical practices are necessary to achieve stated quality goals, etc.⁹

⁸ Mark Levinson, "Can Product Owner and Scrum Master be Combined?" Dec. 11, 2008, <http://www.infoq.com/news/2008/12/scrum-master-product-owner>

⁹ Mountain Goat Software, "The Product Owner," *Topics in Scrum*, <http://www.mountaingoatsoftware.com/product-owner>

Another important measure of quality that does not readily lend itself to a formula is the impediments to development, testing, and rework.¹⁰ These are most often identified in the Scrum meetings.

Progress

Progress is understood as converting requirements into a working product, the latter being the actual value realized. One example is cycle time, which is the period elapsed between the time a customer specifies a product feature and the time that feature is delivered back to that customer in production.

This critical process metric allows many organizations to compete on the basis of speed. Cycle time can be measured by using a value stream map that details value-added and non-value-added activities along the entire project delivery chain.¹¹

Organizational Readiness

Whenever an organization embarks on a bold new initiative, it is always a good idea to get as many people as possible on board with the idea and excited about the possibilities. This is particularly true of an initial Agile pilot project—a best practice approach that mitigates risk and allows initial experimentation and learning with Agile. But it is important to make *everyone* feel part of the Agile team. A companywide explanation of the approach and goals of the initial Agile project—as well as an orientation on why Agile is important to each employee and his or her function and job—will help to bring about awareness and the beginning of buy-in. It is important to show how various functions (HR, finance, etc.) can benefit from an Agile approach—and then help the teams to implement Agile and enjoy its process and results.

As with any new initiative, management commitment is key. With Agile, however, it means more than money and resources. It also requires patience to wait for the training to take effect, new behaviors learned, missteps to be corrected, and results achieved.

In addition to requiring the training and orientation mentioned above, there are other important steps to take to get Agile to stick in an organizational culture. They are:

- A transparent report on what went right in the pilot, what went wrong, and what could have been done better
- Specific actions to make the process better in the next iteration
- Pilots rolled out in other areas of the organization until each element has conducted and evaluated at least one Agile pilot project

The end result will be fewer bureaucratic layers (decisions are made at the lowest level possible), less paper, and a more collaborative and effective organization.

Of particular value is applying Agile practices to organizational functions. One way to do this is to start weekly Scrum meetings in functions that may not have formal projects (HR, finance, contracts, etc.) around topics such as process improvement, work bottlenecks, or documentation. There is no set prescription for making administrative documentation Agile; the best approach is to apply the tenets of Agile in a pilot for each area of the organization:

- Be willing to risk failure. If failure occurs, learn from the mistakes.
- Streamline. Every document created, maintained, or stored must have a purpose. Eliminate redundant documents.
- Be creative. For example, instead of creating a lengthy business plan, draw a narrative illustration on a whiteboard, create an animation of the product (or service) on a computer, or build a prototype out of cardboard. Once the idea is approved for further exploration, you can get the idea down on paper later.

In all cases, celebrate victories, announcing successful project completions and emphasizing where (and how) the Agile approach has helped. You have nothing to lose but paper; and the time and enthusiasm you gain from streamlining processes can be put to creative use.

¹⁰ Ibid.

¹¹ Augustine and Bankston, p. 2.

Program Management for Agile: Transitioning Toward a New Organizational Paradigm

Table 3. Contrast between Traditional and Agile PMOs

Methods and Rules		
<i>Management Practice</i>	<i>Agile Approach</i>	<i>Traditional Design/Build</i>
Justification—Business Case and Charter	Project value proposition within a governance framework.	Project value proposition within a governance framework.
Scope	High-level feature hierarchy fully established early in project with WBS and allowance for discovery of operational needs and incremental prioritization.	Detailed WBS established early in project and put under configuration control to manage scope creep; customers asked to commit to needed features and functions early in the project life cycle.
Requirements	Vision and macro requirements established at initiation, while detail emerges later as functional and iterative increments. Modules are prioritized with customers in a program roadmap. Detailed requirements discovered and prioritized during incremental development. Documentation only to the level needed to maintain the increment.	High-level vision and robust detail established early in the life cycle before building begins. Strict configuration control and management of requirement changes at all levels. Reliance on written documents to communicate with operations for approval.
Scheduling	Master schedule developed with customer based on a functional release schedule. Details within the increments managed within the increment team and not necessarily migrated to the master schedule.	Rolling wave—activities broken down to fewer than two weeks within the wave. Detailed activities become part of the baseline integrated schedule.
Integrated Planning	Real-time integration planning develops over time, often in weekly sessions with team leaders. Major integration points established during planning focused on “what” and “why” with “how” during incremental module development.	Master schedule with dependencies established early. Budget and schedule tied to the integrated plan. Team members work to the plan and customer notified when the plan must change to meet objectives.
Procurement/Contracting	Collaborative decision making among the parties can relieve pressure on liability, warranty, and similar issues. The contract must support relationships based on collaboration, transparency, and trust in the operation of the Agile process and incremental delivery.	A performance work statement or statement of objectives is a best practice typically with a fixed priced contract. The contract process compartmentalizes risk to the parties. Focus on project deliverables.

Program Management for Agile: Transitioning Toward a New Organizational Paradigm

Skills and Training		
Collaboration	Daily team integration with weekly management engagement; work across organization's business silos. Share understanding and discoveries immediately.	Reliance on a good project plan that enables individual, team, and organizational contributions. Status, issues, and risks are managed through project reporting.
Lean Six Sigma	A systems approach to collaboration with a focus on eliminating non-value added activity during development.	Focus on streamlining processes prior to development and during sustainment.
Business Analysis	Business process modeling and requirements management through user stories or similar mechanisms.	Business process modeling precedes detailed requirements definition and project planning; intensifying again during implementation.
Staffing	Semi-autonomous small teams work collaboratively as self-organizing, self-regulating, collocated groups of individuals. Product owner representative included in the development team. Teams consist of senior staff working as peers.	Large integrated teams work on assigned, predetermined activities. May be collocated or dispersed. Use of a wide mix of skills and experience often relying on a few senior staff to guide junior staff.
Organizational		
Prioritization	Portfolio priorities established collaboratively across the organization. Project priorities can be established just before incremental development with customer.	Portfolio and project priorities established during (typically) annual budget process and reviewed quarterly.
Customer Involvement	Throughout the project.	Focused on early definition and planning and final acceptance.
Value (alignment to strategy)	By working in small increments with the ability to prioritize within the increment, there is a nearly continuous alignment of project work to value.	A tendency (though not a best practice) to measure performance strictly against the project plan delivery dates. Value realization depends upon full implementation.
Manage Constraints	Portfolio management used to identify and reallocate resources incrementally based on continuous identification of priority project components. Mid- and long-term resource management based on strategic alignment versus aggregate project plans.	Constraints and mitigation/buffers are built into the project plan.
Transition to Sustainment	Enterprise view by release. Customer acceptance of increments; collaboration continues throughout the life cycle. Prioritize and organize across organizational silos.	System readiness review with formal handoff to operational business units.

Conclusion

You may be more Agile than you think. With a shift in orientation, most current PgM practices apply in an Agile environment. The biggest transition is not in reengineering processes but in management. A management approach for Agile involves different expectations, working relationships, incentives, metrics, and reviews. The next challenge is to grow the Agile approach across the enterprise.

As we've seen, the Agile approach and mindset can be applied to activities beyond IT or even product development—in fact, to essentially any endeavor involving a knowledge worker environment. A number of program management practices can be fine-tuned to support Agile programs to even greater effect. For example:

- Stakeholders must increase collaboration on priorities and requirements for the project using an enterprise perspective instead of a narrower business unit or functional one.
- The PMO must remove project and enterprise constraints for Agile teams to be successful. Examples include providing access to resources, and aligning Agile management philosophy/priorities with a clear rewards system.
- There must be a focus on project value and throughput and the realization that cost and schedule, although important, are means to an end.

The intersection of good PgM and an Agile approach means faster project delivery, higher quality, and improved predictability from frequent communication between the team and customer. When program management is applied to Agile, the benefits of both approaches—collaboration, prioritization, and project value—make it powerful, results-focused model to implement.