



Global Knowledge®

Expert Reference Series of White Papers

A New Trend in
Agile –
Incorporating
Program
Management

A New Trend in Agile – Incorporating Program Management

Brian D. Egan, MSc, MBA, PMP

Introduction

How can an organization reap the benefits of both traditional (waterfall) and more flexible (agile) project management techniques? The answer is through program management designed to deliver enhanced agility.

Agile project management practices are being widely adopted in the IT project management world. But classical agile methods were created for software development and are therefore not suitable for every situation and industry.

If classical Agile techniques do not work for every situation, how can organizations in diverse industries become more agile? The answer is by replicating the features of Agile software development techniques (the ones that contribute most to agility) and combining them with conventional project management practices.

Agile program management is structured to coordinate and direct projects with short planning windows, lighter documentation, and higher degrees of customer involvement. The result is a hybrid system of project delivery that is more agile than a traditional (waterfall) approach.

Avoiding Destination Fixation

A common failing of long, waterfall-style projects is that these projects often develop “destination fixation.” They emphasize the work that has been planned rather than building what is best for the organization. The waterfall approach encourages the team to follow the original plan (through destination fixation) rather than adapting to what the customer actually needs.

In waterfall, changes to the project are integrated into the project’s plan through formal change controls. However, change controls are generally unsuccessful at managing substantive or continuous change because of the documentation involved and the effort required to obtain approvals.

The inertia in the waterfall approach is to “stay on plan,” which prevents effective adaptation to changing realities. It is easier to follow the original, formally approved plan than it is to alter it. Enhanced agility breaks the cycle of focusing on long-term plans and allows the organization to focus on delivering benefits.

Separating Agile Methodologies from Agility

Implementing a full suite of classical Agile project management methodologies would require radical changes in the way projects are approved and managed in most organizations. As a result, many of the organizations that are attempting to “go Agile” are simply picking and choosing a few Agile-related techniques (such as daily scrums and the use of user stories) rather than converting to a full Agile methodology. The result is a blend of old and new techniques but not necessarily a more agile project management organization.

To be more agile is to be able to change direction, to react to reality, and to be responsive to discoveries. An agile project management organization is one that can pivot as needed in response to changes in technology and market forces.

Agility is the ability to rapidly change execution strategy. Trains running on railway tracks are not agile. Rabbits are. A project management organization is agile if the systems employed to authorize and manage projects are responsive to change.

An organization becomes more agile if the systems that support projects allow for more frequent re-evaluation and alteration of a project's execution strategy. Increased agility is not simply a consequence of adopting a few Agile project management tools. Agility is determined by how responsive or adaptive to changing conditions the overall system of project management is irrespective of the techniques employed.

To be more agile is within reach of any organization with or without the use of classical Agile software development techniques. Agility comes from having a responsive, flexible system of project management.

When considering whether to become more agile, the question an organization must answer is not whether they should adapt classical Agile software development methodologies but rather, what is the right level of agility to aspire to and how to achieve it? What is the best way to become suitably agile?

Increasing Organizational Agility

The keys to greater organizational agility are:

- A. Shorter planning windows (more frequent, shorter projects linked together into a cohesive program),
- B. Lighter documentation (made possible by shorter planning windows and hands-on strategic coordination provided by program management), and
- C. Increased customer involvement (supported by short projects and program oversight).

Shorter projects require less knowledge of the distant future. They are individually easier to plan and to execute successfully than projects with long planning windows.

Lighter documentation both speeds up the rate that projects can be planned and reduces the commitment to long-term planning that often prevents effective adaptation to changing situations.

Increased customer involvement helps to ensure that final outcomes deliver the maximum benefits to the intended audience. It reduces "destination fixation" in which a project builds whatever was originally conceived rather than what was the best solution at the time of completion.

Collectively, short projects, light documentation, and enhanced customer involvement can be combined to deliver a more agile project management environment than would be possible with the standard, long-term waterfall approach.

Program Management to Bind Them

A project management environment that employs shorter projects, light documentation, and customer involvement requires strong leadership. This leadership is delivered through program management. The program manager becomes the connection between senior management and the project teams. They serve as the "voice of the customer," ensuring that project efforts remain focused on the best possible outcomes given current conditions.

In Agile program management, projects are planned and managed using familiar waterfall practices (plan the work, work the plan). The difference is that these waterfall-style projects are short in duration (two to six weeks), and between each short project, there is the opportunity to reassess the overall program's strategy. These points of reassessment are called "strategic inflection points." It is through these strategic inflection points that an

organization's project management system becomes more agile.

Agile Software Development Life Cycle

Figure 1 below illustrates an example of the classic Agile project management life cycle. The salient points are:

- A. Preplanning at a strategic level, followed by
- B. Detailed planning at the beginning of each iteration.
- C. Between each iteration, there is a customer review of the product to date. This is followed by analysis of lessons learned by the team with the objective of improving performance in the next iteration.
- D. Revelations from the customer reviews and from sources inside or outside the project lead to changes in what was planned for subsequent iterations. Adaptation to change is a routine part of the planning of the next iteration. (This adaptation to change is the fundamental basis of the system's agility. These strategic inflection points are what deliver agility in the Agile methodologies.)
- E. Big discoveries (or change) force a re-think of the overall strategy, which implies the overall plan for iterations and releases may be revised.

And the cycle repeats itself.

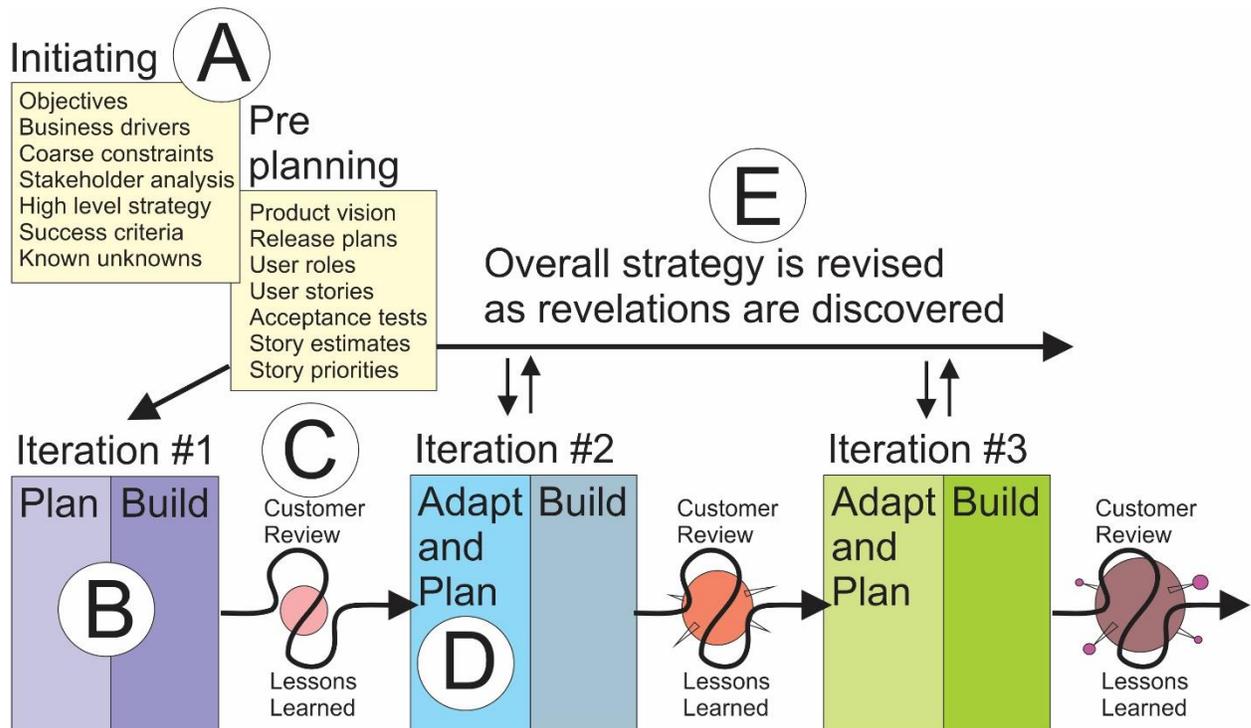


Figure 1. Agile Software Development – Project Management Life Cycle

Important Characteristics of the Agile Life Cycle

The characteristics of Agile project management methodologies that contribute most to an organization's agility are:

Short iterations. Iterations, in the agile world, are just short projects by another name. They are actually waterfall-style projects in the sense that they begin with detailed planning and rigid expectations of scope (stories that will be developed), time (iteration length) and cost (resources available during the iteration). But, of course, they are very short, so the problems of rigid planning do not undermine the effective long-term flexibility of the

overarching project.

The shorter the iterations, the more frequently the adaptation to change can take place. An agile project with iterations of five months each would be far less "agile" than a project with iterations of only two weeks. This point needs to be emphasized. The strategic inflection points between iterations (where teams can adapt the strategy to changes and re-plan) are what deliver agility.

The tools of agile software development (user roles, stories, rough estimates, etc.) are designed to support quick revisions of the project plan. A system of project controls is only agile if there are regular opportunities to revise the strategy before moving forward. Adaptation will only occur if the planning system lends itself to change.

The management of individual iterations is actually quite rigid. External stakeholders are not permitted to impose change on the project during an iteration. This is only allowed between iterations.

Change is restricted during iterations because change is disruptive to prepared plans (even for short projects). Changes discovered during an iteration are integrated into the overall plan through adaptation at the next inflection point (between iterations).

Light documentation. If you are only planning in detail for one iteration at a time (two to six weeks), then it is not necessary to have a deep base of documentation.

Not having detailed plans long into the future supports adaptation and change. Change controls are brutal in conventional waterfall projects because of the complexity of acquiring approvals and of updating a myriad of documents. Reduce the length of your forecasts (plans) and it becomes easier for projects to move quickly and change strategy as necessary. Light documentation saves time at the outset by making it faster to get work started.

"Just in time" planning describes the Agile software development approach to project documentation. Long-term plans are high level with broadly defined objectives. They are strategic and focused on long-term outcomes. Short-term plans (iteration plans) are detailed; they are tactical and focused on near-term (two to six week) outcomes.

With just in time planning, scope is developed progressively. It is developed in detail as needed to support the next iteration and no more. Ambiguity in terms of what will happen in the distant future is allowed until the details are both necessary (one can't proceed without it) and available (stuff has been learned that allows for scope to be defined). Planning for an iteration (i.e., the next mini project) proceeds when enough is known to be sure that the iteration will add value to the overall project.

Customer focus. Agile project management methodologies depend on continuous customer involvement. This is achieved in three ways:

First, the customer (or their representative) participates in the planning of each iteration by helping set priorities and answering questions.

Second, the customer (or their representative) is expected to be available to the development team while work is underway.

Third, the project's product-to-date is reviewed with the customer at the end of each iteration. These reviews are structured to ensure that the team understands what went right and what went wrong. They serve to both initiate corrective behavior on the part of the team and to encourage customer involvement.

By keeping the customer continuously involved, agile teams demonstrate that they are working for the customer. Thus, the agile approach encourages a focus on benefits delivery. Continuous customer involvement is a key success factor for agile projects. It both reduces the need for advanced requirements documentation and it ensures that the customer's current perceptions are used to drive design.

Addressing Problems of Complexity

In general, IT projects move quickly and suffer from a lot of complexity. Complexity is derived from such things as having numerous stakeholders, involving new technologies, dealing with overlapping systems, and requiring diverse skill set. Conflicts between development work and ongoing operations also contribute to complexity. Making matters worse is the fast pace of technical innovation and ever-evolving stakeholder expectations.

High complexity makes it difficult to define a project's scope of work in advance and, therefore, to estimate the cost and effort required to complete that work; subsequently, this is a big problem because these are necessary ingredients of the traditional waterfall approach to project management.

By contrast, Agile project management techniques allow requirements to be refined progressively rather than up-front. The execution strategy (project plan) is permitted to evolve as knowledge about the scope evolves. This progressive evolution of scope allows work to begin quickly when just enough is known to get started and without knowing everything. It supports "just in time" planning.

The Downside of Agile Methodologies

However, there is a reason why Agile methodologies have not been universally embraced. Agile projects do not begin with a detailed plan; they do not begin with a promise of what will be accomplished in the end like a waterfall system of project management does, and that is a huge hurdle for the people approving projects.

The waterfall approach to project planning creates a detailed forecast. This gives the people paying for projects a greater sense of confidence in what will be accomplished and of the total cost to which they are committing. There is the perception that a waterfall-style approach to project planning reduces uncertainty and, thus, reduces the risk of failure. Unfortunately, history tells us that it is not true. Waterfall planning does not reduce the risk of failure in a rapidly changing, complex environment.

Waterfall's Weaknesses

Waterfall project management is based on heavy up-front analysis, which leads to detailed "plans." A project's plan is a forecast of future outcomes in terms of the work that must be completed, the cost involved, and the schedule for all that work. The waterfall approach is to plan (or attempt to predict) exactly what will be done and then execute the project according to the plan. Detailed predictions offer a conceptual logic that provides comfort (i.e., a perception of risk reduction) to senior managers who are funding the projects.

Unfortunately, the waterfall approach is based on the presumption that scope can be unambiguously defined at the beginning of a project. The waterfall planning system depends on being able to define all of the work (or scope) in advance.

And that is the problem.

IT projects are rarely given the time necessary for sufficient analysis to produce accurate forecasts. Even if time were available, changing technology and operational complexity inherent in many IT projects makes detailed scope definition, in advance of starting the work, virtually impossible. By the time an exhaustive study of requirements is complete, the requirements are already out of date because of technical and market changes. Painstaking requirements analysis has not proven to be a guarantee of IT project success – just delays, frustration,

and misdirected efforts.

Waterfall project management requires settings that are stable enough to support a methodical approach to planning. Suitable stability is not the norm in the rapidly changing IT sector. Therefore, what is needed is an updated waterfall model.

Enter Agile program management.

Agility in a Waterfall World

Can we have the familiarity of a waterfall approach and some of the agility promised by classical Agile methodologies? Yes, by using numerous short projects (rather than big long ones) that are strategically coordinated as part of an overarching agile program.

Figure 2 below illustrates an example of an agile program management life cycle.

- A. Programs are initiated in a similar way to traditional projects with an emphasis on strategic objectives.
- B. Program planning revolves around benefits definition and delivery.
- C. Short, conventionally planned projects (or iterations) produce interim deliverables.
- D. Project deliverables are reviewed by the program team for completeness and correctness (outcome review).
- E. Project teams assess their performance and make recommendations for improvements on future projects (lessons learned).
- F. The combined product, from all projects-to-date, is reviewed with the customer to ensure correctness and completeness. Additional lessons learned are captured.
- G. Plans for the next round of projects begin by adapting to what has been learned and revealed. (These inflection points are what make the system agile.)
- H. The project's product continues to be developed by one or more additional projects.

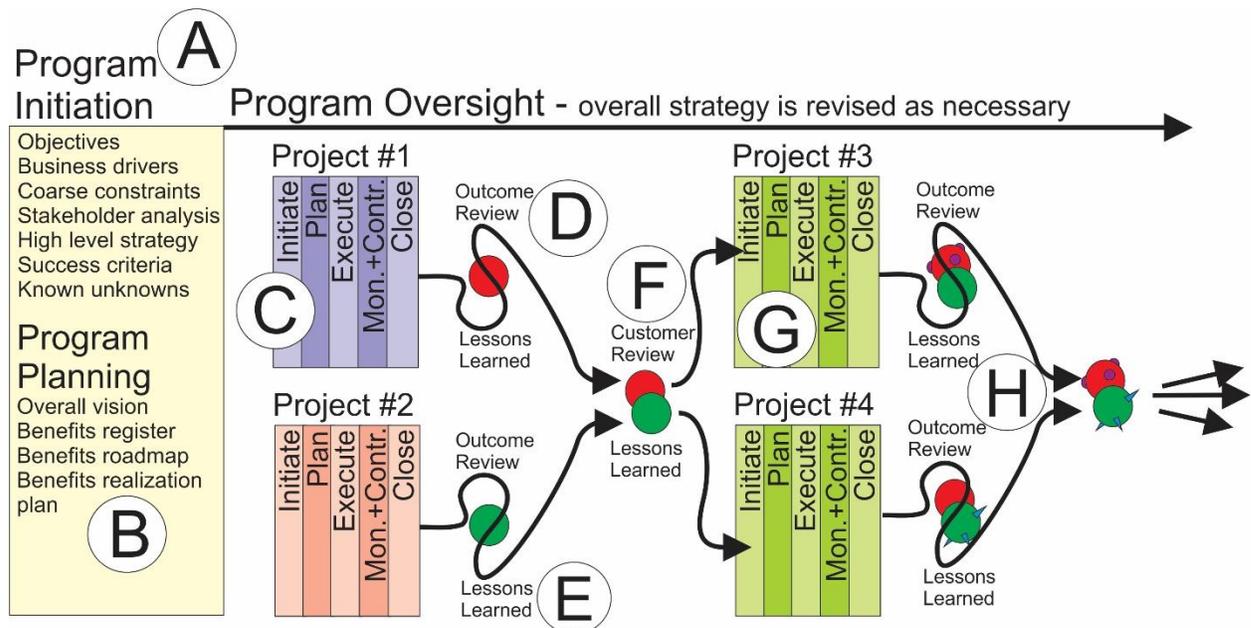


Figure 2. Agile Program Management Life Cycle

Individual projects are short in duration but are planned in the conventional waterfall way. Between projects, there is the opportunity to adapt to revelations both from individual projects and based on the combined outcomes of multiple projects.

Light documentation is made possible by shorter planning windows and careful stewardship by the program managers.

Customer involvement through the program manager as the 'voice of the customer' ensures that the team never loses sight of the overall objectives.

Agile program management delivers the crucial elements that underlie the agility of Agile software development methods without having to adopt all of the agile tools and techniques.

The Role of the Program Manager

To guide and direct. Careful oversight is key to the success of agile programs. Project teams are able to focus on their small world (only their projects) because the bigger picture, and stakeholders are generally managed by the program team.

The program manager serves as both mentor and customer. They become the key stakeholder of the short projects, act as the "voice of the customer," and keep the project's focus on short-term objectives while simultaneously keeping the program's emphasis on long-term benefits.

To monitor and control. The program manager monitors project efforts and outcomes to make sure each small project contributes incremental benefits. They control re-planning and strategy development between projects to ensure delivery of program benefits.

Keys to Success

Success with the Agile program management approach hinges on:

1. **Program managers who understand the business value driving the investment.** They are the captains of the ship, altering course as necessary to ensure that maximum value is delivered despite changes in weather and competitive forces. Program managers must be well informed, highly involved, and senior enough to approve corrective actions. Their role is to avoid destination fixation while optimizing outcomes.
2. **Adaptation to reality.** After each iteration, the execution strategy must be permitted to evolve. The next round of project plans must reflect what has been learned and must respond to changes in the program's competitive environment. Each round of planning must be more informed than the last.
3. **Light project documentation.** Light documentation is made possible by careful and continuous oversight by the program team. Long-term plans are written in pencil. They serve as guidelines rather than rigid strategies. Documentation must be flexible enough not to discourage innovation of the strategy at the inflection points between projects.
4. **Stakeholders who understand why Rome was not planned in advance.** If the stakeholders expect a detailed plan for all the work to be done before any of the work begins, you cannot have an agile program. Stakeholders must accept that the program team is working for them to optimize outcomes and need to appreciate that complexity and compressed timelines prevent accurate long-term forecasting.

Real Risk Reduction

In the waterfall world, risks are managed by having a detailed plan and “fixed priced” relationships between the project team and sponsoring organization. A waterfall project plan binds the team to a specific set of outcomes (scope, time, and cost). The stakeholders, therefore, think the risk of failure has been reduced. But history tells us otherwise. You cannot plan for what you do not know, and you cannot find out without plenty of time and resources.

Faced with operational complexity and rapidly changing environments, organizations need to manage risk differently. How successful would army generals be if they planned every troop movement in advance of a battle, and if, during battle, any changes to the battle plan had to be approved through a laborious military hierarchy? Not very.

Success in a fluid environment comes from planning only as far into the future as there is certainty. Beyond that, you need to be flexible, and that flexibility comes at the cost of not being able to demand long term, rigid plans from project teams.

Greater organizational agility reduces risk, but it requires a substantive change in attitudes towards risk management, which also requires a new approach to how projects and programs are funded and controlled.

Conclusion

Agility means being able to adapt to changing conditions and being responsive to evolving needs. In a rapidly changing, complex environment, greater agility contributes to lower project costs and more effective project outcomes. Greater agility delivers this by making project management systems more responsive to discoveries, revelations, and changes that arise during execution.

Every organization can be more agile but only to the extent that they are willing to plan tactically in the short run and strategically in the long run. The agile approach requires near-term specifics and long-term generalizations.

What holds organizations back from being more agile is funding models that require waterfall style project planning complete with heavy up-front documentation that produces rigid plans for outcomes long into the future. These detailed execution strategies create momentum for investing in solutions that quickly go out of date but cannot be derailed. This results in “destination fixation” and projects that build the wrong things.

The “magic bullet” is Agile program management. Agile program management enables the delivery of benefits through a series of shorter projects that are strategically tied together. These short projects allow the overall execution strategy to be regularly revised. Documentation is kept light to support flexible planning. Detailed documentation (such as plans and expectations) is only developed for near-term deliverables, and general documentation is only created for long-term outcomes. The combination is a more agile project delivery system that better serves ever-evolving customer needs emphasizing the short-term without losing sight of the long-term.

The agile program mantra:

Near term specifics, long term objectives.

Learn More

Learn more about how you can improve productivity, enhance efficiency, and sharpen your competitive edge through training.

[Agile Project Management](#)

[PMI Agile Certified Practitioner \(PMI-ACP\) Boot Camp](#)

[Program Management](#)

Visit www.globalknowledge.com or call **1-800-COURSES (1-800-268-7737)** to speak with a Global Knowledge training advisor.

About the Author

Brian has been a contract instructor for Global Knowledge since 1999. He divides his time between management consulting, project management, technical writing, and professional development training.

Brian is a “serial entrepreneur.” He has started companies in such diverse fields as fish farming, woodwork, gift manufacturing, and catering. He is the author of numerous training courses relating to professional skills, project management, and decision making.