

CHAPTER 1

INTRODUCTION

“Agile” is definitely the latest and coolest buzzword in the software development world—everyone wants to be “agile,” but there are many misconceptions of what “agile” means, and many people don’t fully understand the implications of what it takes to develop an effective agile development process.

MEANING OF THE WORD “AGILE”

First, it is essential to define what we mean by the word “agile”. In the software development arena, the word “agile” has become synonymous with specific forms of agile such as Scrum and Extreme Programming (XP):

Scrum is an agile software development model based on multiple small teams working in an intensive and interdependent manner. The term is named for the scrum (or scrummage) formation in rugby, which is used to restart the game after an event that causes play to stop, such as an infringement.

Scrum employs real-time decision-making processes based on actual events and information. This requires well-trained and specialized teams capable of self-management, communication and decision making. The teams in the organization work together while constantly focusing on their common interests.¹ (See Appendix B for more detail.)

Extreme Programming is a discipline of software development based on values of simplicity, communication, feedback, and courage. It works by bringing the whole team together in the presence of simple practices, with enough feedback to enable the team to see where they are and to tune the practices to their unique situation.

In Extreme Programming, every contributor to the project is an integral part of the “Whole Team.” The team forms around a business representative called “the Customer,” who sits with the team and works with them daily.

¹ What is Scrum?, http://searchsoftwarequality.techtarget.com/sDefinition/0,,sid92_gci1230820,00.html

4 Introduction

Extreme Programming teams use a simple form of planning and tracking to decide what should be done next and to predict when the project will be done. Focused on business value, the team produces the software in a series of small fully integrated releases that pass all the tests the Customer has defined.² (See Appendix A for more detail.)

This connotation of the word “agile” has become widely accepted in common usage in software development and creates the impression that the only way to be “agile” is to practice those specific methodologies. In other words, if you’re not doing Scrum and/or Extreme Programming, you’re not agile at all.

The meaning of the word “agile” has become confusing because of these widely-used specific connotations that have become associated with it. These connotations are also in the context of a development perspective and a much broader context is needed. A more general connotation of the word “agility” is defined by Dr. David Rico³ as follows:

- “The ability to create and respond to change in order to profit in a turbulent global business environment
- The ability to quickly reprioritize use of resources when requirements, technology, and knowledge shift
- A very fast response to sudden market changes and emerging threats, by intensive customer interaction
- Use of evolutionary, incremental, and iterative delivery to converge on an optimal customer solution
- Maximizing the business value with right-sized, just enough, and just-in-time processes and documentation”⁴

The word “agility” as defined above has a much broader meaning than the meaning that has typically been associated with the word “agile”—it implies that there are many different levels of “agility” and not just a limited number of specific ways to being “agile”. A key idea that this book will develop is that there is a spectrum of different approaches to “agility”:

- At one end of the spectrum, there are approaches such as the traditional Waterfall methodology that heavily emphasize upfront planning and control and have a very limited amount of agility. These approaches might be classified as “plan-driven” or “predictive” because they use upfront

² What is Extreme Programming?, <http://xprogramming.com/book/whatisxp>

³ Rico, David F., Lean and Agile Systems Engineering, <http://davidfrico.com>

⁴ Rico, David F., “Lean and Agile Systems Engineering, <http://davidfrico.com>

planning heavily to attempt to predict the cost and schedule for a project. They also typically place an emphasis on controlling changes once the project is in progress to manage the scope of the effort to ensure that the original predictions of costs and schedules remain valid.

- At the other extreme, there are approaches such as Scrum and Extreme Programming (XP) that have a much higher level of agility and would be considered much more “adaptive” because they place a much higher level of emphasis on being flexible to user needs and requirements as the project progresses and they have a much lower level of emphasis on upfront planning and control of costs and schedules.

In the middle of these two extremes, there are many ways to blend together different levels of agility and control to fit a given business environment.

This definition of the word “agility” also acknowledges that “agility” includes a higher-level business perspective as well as a development practice perspective and that the two must be integrated. Scrum and Extreme Programming (XP) are excellent methodologies that can have an enormous impact on improving a company’s software development process, but too often, a migration to agile is led from a development strategy perspective driven primarily by implementing a new development methodology such as Scrum and XP, and there are much broader implications that need to be considered such as:

- What role does the product development process play in the company’s business? What are the advantages to be gained by becoming more agile?
- What is the most appropriate balance of agility and control for the company’s business environment and competitive strategy?
- What are the benefits and tradeoffs associated with different levels of agility that might be used to provide that balance?

In this book, to avoid confusion about terminology, the phrase “pure agile” or “extreme agile” will be used to refer to the most adaptive forms of agile methodology such as Scrum and Extreme Programming (XP).

MEANING OF THE WORD “WATERFALL”

The agile movement started out as a revolution against traditional development practices, such as the Waterfall, process that have been perceived as heavily laden with documentation and were perceived as cumbersome, ineffective, and extremely bureaucratic. Figure 1.1 shows a typical Waterfall model.

The Waterfall process is called that because it is a series of sequential phases that have to happen in sequence, and each phase cascades into the next. For example, one of the early phases is a requirements definition phase, where the user requirements are defined and documented and then handed off to the

6 Introduction

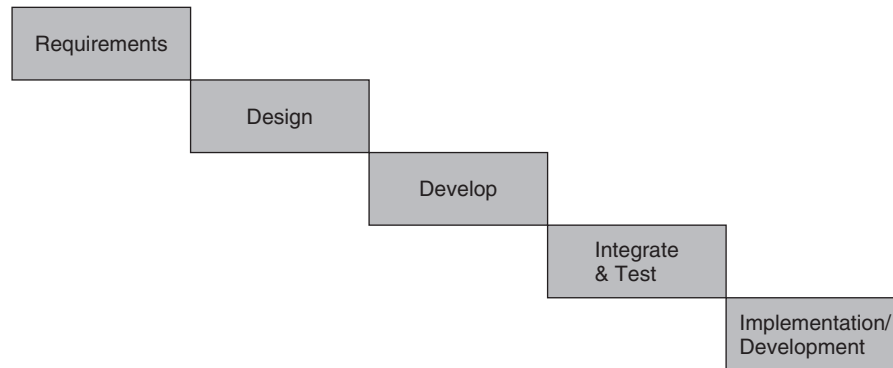


Figure 1.1 Typical Waterfall model

development team in the next phase to develop a solution. There are several significant problems that are inherent in that kind of process—it assumes that:

- The users are capable of defining explicitly detailed requirements for everything that they need without seeing it at all.
- The business environment is also very stable and the requirements aren't going to change much throughout the rest of the project.
- The requirements can be accurately documented in a way that developers are going to easily understand what needs to be done to satisfy those needs and nothing will be lost in translation.

Those may not be very realistic assumptions in many situations today, but the Waterfall approach has appealed to companies who have seen it as a way to get control and predictability over project costs and schedules. The truth is that control over costs and schedules is an illusion if the requirements are uncertain and are likely to change significantly. It may appear that there is a well-defined plan with accurate and predictable estimates of costs and schedules, but one of two things is likely to happen:

1. The rigidity of the requirements and the change control process make it difficult to adapt to user needs as the project progresses. The project may meet its cost and schedule goals but miss the mark in satisfying the business need if changes in user needs are overly controlled.
2. On the other hand, if users are allowed to make changes freely, attempting to impose a rigorous change control process in an environment with very uncertain user requirements can create enormous amounts of unnecessary bureaucracy processing change requests on top of change requests and can make the earlier estimates of costs and schedules meaningless.

Attempting to apply a Waterfall model in that kind of environment is unrealistic, yet many project managers attempt to do it anyway because it provides

an “illusion” of control. It’s much better to just accept the fact that the user requirements are uncertain and choose a model that is designed to be more flexible and adaptive to uncertain user requirements. The cost and schedule estimates may not be as exact and precise as you might like, but it’s unrealistic to believe that the cost and schedule estimates can be any more exact and precise than the requirements are.

POLARIZATION OF AGILE AND TRADITIONAL WATERFALL APPROACHES

There has been a considerable amount of polarization associated with traditional Waterfall and agile approaches:

- The agile movement was essentially a revolution against bureaucratic, Waterfall-style processes, and people in the agile community wanted to distance themselves as far as possible from those practices. For that reason, when the agile movement started out, it moved the pendulum to an extreme point in the other direction away from the traditional Waterfall approach (little or no documentation, process, or methodology, etc.).
- At the other extreme, some companies and practitioners of the traditional Waterfall approach have developed a strong and deeply rooted control orientation with an emphasis on accurately estimating and managing costs and schedules that can be difficult to change. If your goal is to accurately manage costs and schedules, any good project manager knows that it is essential to control and manage changes in requirements and scope. For that reason, it’s very understandable why agile methodologies that emphasize unrestricted flexibility to adapt to change would feel very uncomfortable to those people.

In 2003, Barry Boehm and Richard Turner observed:

“Unfortunately, rather than find ways to support each other, these two approaches to software development have considered each other opposites in a zero-sum game. The agilists rail against the traditionalists and lament the dehumanization of software development by ‘Taylorian’ reductionists who worship process. The establishment has responded with accusations of hacking, poor quality, and having way too much fun in a serious business. True believers on both sides have emerged to proclaim their convictions with near messianic stridency, raising the perplexity level of software developers and managers trying to evolve their success strategies.”⁵

⁵ Boehm, Barry and Turner, Richard, *Balancing Agility and Discipline—A Guide for the Perplexed*, New York: Addison-Wesley, 2003, p. 4

8 Introduction

Since that time, the reality is that both sides of this conflict have evolved considerably, yet there is still a perception in some cases that they are far apart:

- The agile movement has matured rapidly and considerably. Methodologies like Scrum have evolved that are more than just a development process and have a strong base of knowledge and experience behind them, yet many people still retain an image of the agile movement as just an undisciplined development process from its early anarchist, revolutionary days.
- Many people have found innovative ways to make traditional development approaches more agile by making the documentation and process requirements much more sensible and adopting more iterative development approaches that balance control with agility, yet many people still have an image of traditional development approaches as highly dogmatic and bureaucratic.

It's time to find the middle ground and start to build a bridge across this chasm—much of these differences are more rooted in perceptions and misconceptions rather than in reality and, in many cases, the judgment and skills associated with how the methodology is applied has as big an impact as the methodology itself.

There is much to be learned from both of these areas and a broad base of knowledge and a continuum of different methodologies, practices, and principles from both the agile and traditional project management approaches need to be considered in many cases to develop an effective overall strategy. The selection of a project methodology (or methodologies) is a very important strategic decision for all organizations that depend on effective project management. The approach needs to be well aligned with the company's business strategy, culture, and business environment, as well as the risks and complexity of individual projects. For example:

- In very high-risk industries and application areas, depending on the nature of the risk, a considerable amount of upfront planning may be needed to ensure that the risks have been defined and mitigated.
- In highly regulated environments, project requirements, test plans, and test results may have to be well documented to satisfy regulatory requirements.

In these situations and many others, a balance of control and agility is needed to satisfy these requirements, and there are many ways to provide some level of agility without completely sacrificing control. Selecting an overall approach that provides the right balance of control and agility should be done jointly by the business and development sides of the organization based on an objective understanding of the alternatives and all the issues and tradeoffs involved. Attempting to use any standard methodology (either agile or non-agile) by the book that doesn't fit the risks and complexity of the business and project environment is not likely to be very successful. And, in many cases, the optimum solution may

not be a single, standard methodology at all, but a combination of methodologies that can be customized and tailored to the requirements of each project.

THE PROGRAM DU JOUR EFFECT

Any new methodology, when it is really the hot thing to do, has a tendency to become the “program du jour.” (Or program of the day) Agile methodologies have the potential to have an enormous impact; however, like many other new and hot methodologies:

- Consultants tend to swarm all over them and sell them as a cure for almost anything that ails you.
- Many companies and managers want to jump on this bandwagon, and that further builds the hysteria in the market.

The result of this can be:

- Jumping into agile looking for a “quick fix” without fully realizing that it takes a significant commitment to make it successful, resulting in superficial implementations that are likely to fail
- Attempting to implement agile methodologies in a business environment or organizational culture that is inconsistent with an agile approach
- Attempting to use agile methodologies for projects that they are inappropriate for or failing to blend a sufficient level of agility with the level of control that is needed

I’ve seen this phenomenon before. When I did the research for my book on business excellence,⁶ I looked at a number of companies that were implementing Six Sigma, which was a very hot methodology at that time:

- There were some companies where the implementation followed the program du jour pattern—there was a lot of hoopla about the implementation, there were green belts and black belts and many of the other rituals that go along with Six Sigma, but if you looked under the surface, you quickly discovered that it didn’t go very deep.
- On the other hand, there were other companies where it wasn’t even obvious that they were doing Six Sigma because it was so well engrained into their business, and they took the time to understand the principles behind Six Sigma at a deeper level. They didn’t even necessarily call it Six Sigma. These companies had their own process improvement methodology that was really fully integrated into how their business operated, and the methodology behind Six Sigma was only one tool in their toolbox.

⁶ Cobb, Charles G., *From Quality to Business Excellence*, ASQ Quality Press, Milwaukee, WI, 2003

10 Introduction

Another phenomenon I've seen before is the tendency to “throw the baby out with the bath water” whenever a hot new methodology comes along. In many of these situations, there is a lot to be learned from the previous methodology, but the previous methodology is considered obsolete and passé, and lessons learned from the past are forgotten when people move on to a hot new methodology like Six Sigma or agile. There is a tendency to think that traditional development methodologies and everything we've learned from them are now completely obsolete and have been replaced by agile, and that's not the case at all. There's a huge body of knowledge that has been developed over many years that should not be lost or ignored. Many of those principles, such as those in *A Guide to the Project Management Body of Knowledge* (PMBOK® Guide)—Fourth Edition are still valid; they just need to be applied intelligently in a different context.

A key objective of this book is to try to avoid the “program du jour” effect and present an objective and unbiased view of agile—where pure agile methodologies work and where a more traditional or hybrid approach may be a better solution, how agile methodologies fit with other methodologies, and what it takes to understand all of these methodologies at a deeper level and apply them successfully.

IMPACT ON PROJECT MANAGEMENT

The Standish Group recently released a report called “CHAOS Summary 2009” that has some disturbing statistics that show a very low percentage of projects met their desired objectives:

“This year's results show a marked decrease in project success rates, with 32% of all projects succeeding which are delivered on time, on budget, with required features and functions” says Jim Johnson, chairman of The Standish Group, “44% were challenged which are late, over budget, and/or with less than the required features and functions and 24% failed which are cancelled prior to completion or delivered and never used.”⁷

Jamie Capella of the Corporate Executive Board⁸ presented data that showed a similar result. They did a study that indicated that a large number of projects (greater than 50 percent) that had successfully met their cost and schedule goals failed to deliver the business value that was intended. Both of these reports indicate a need to take a hard look at how project management is being done to determine what could be wrong.

The fundamental problem, in many cases, is one of two things:

- An overemphasis on control of costs and schedules can cause losing focus on the successful achievement of business results.

⁷ Standish Group, http://standishgroup.com/newsroom/chaos_2009.php

⁸ Capella, Jamie, Presentation to PMI MassBay Chapter, March 18, 2010

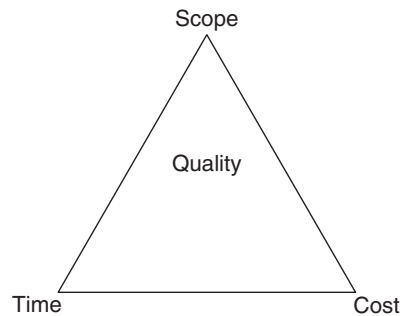


Figure 1.2 Traditional Project Management Triangle

- In a situation where the requirements are very uncertain and difficult to define upfront, attempting to force-fit that kind of project to a rigidly controlled, traditional Waterfall model is probably not the best model.

The traditional Project Management triangle in Figure 1.2 shows a project management approach that has been used for years.

If the project was managed within the constraints of time, cost, and resource availability, and it delivered the items that were within the scope of the project with an acceptable level of quality, it was considered a success. That's the predominant way that projects have been managed for a long time. There is nothing explicit in that triangle about providing business value unless you make the assumption that delivering what is in the scope is going to provide that value, and that's a *very big* assumption.

“Too often project managers (and those above them) focus on the usual constraints of time and cost. There are times when value doesn't seem to matter at all—its schedule, schedule, schedule, as if value will take care of itself. Then there are those that focus on scope and detailed requirements but not the end goal of value . . . The assumption gets made that delivering on scope, schedule, and cost means delivering value.”⁹

In the same PMI[®] Meeting¹⁰ where Jamie Capella spoke, there was a panel discussion with seven Boston-area CIOs that followed. The message they gave was very loud and clear—in today's world, successful project managers need to go well beyond managing costs and schedules and focus on achieving successful business outcomes. This will require rethinking and rebalancing priorities in some cases. For example, the emphasis on management of project costs and schedules can have a negative impact:

⁹ Highsmith, Jim, *Agile Project Management: Creating Innovative Products*, New York: Addison-Wesley, 2010, p. 18

¹⁰ PMI MassBay Chapter Meeting, March 18, 2010

12 Introduction

- In order to control costs and schedules, users are expected to sign off on requirements at the start of a project, and any changes from that point on are well controlled and minimized.
- The level of user involvement in the development effort after the project has entered the design stage may be very limited in order to control changes.

When projects fail to provide the business value that they were intended to provide, there's a usual list of suspects to blame such as:

- The users didn't adequately specify the requirements or the requirements weren't sufficiently defined.
- The project team didn't understand the requirements or the business need changed.

The root cause of the problem may be in the way that projects have been typically managed. Attempting to use an inappropriate methodology on a project may not satisfy *either* the objectives of managing costs and schedules or providing the desired business value:

- Too much emphasis on managing costs and schedules can lead to a somewhat rigid management of project scope that makes it difficult for the users to provide sufficient input and to react to change.
- In an unclear, uncertain, or rapidly changing environment, a very different kind of approach may be needed that has much more emphasis on satisfying business needs as a primary goal and has a lot more flexibility to adapt to change. Attempting to use a methodology that is optimized around managing costs and schedules in that environment is not likely to be successful.

A very different approach is needed to provide the right balance of flexibility and responsiveness to adapt to user needs in an uncertain environment:

“Traditional Project Managers tend to focus on requirements as the definition of scope, and then concentrate on delivering those requirements. Agile Project Leaders focus on delivering value and are constantly asking questions about whether different renditions of scope are worth the value they deliver.”¹¹

Naturally, in this environment, the estimates of costs and schedules cannot be much more accurate than the user requirements are, but that doesn't prevent establishing a ballpark estimate for the project based on high-level requirements that are defined upfront and are further elaborated as the project progresses.

¹¹ Highsmith, Jim, *Agile Project Management-Creating Innovative Products*, New York: Addison-Wesley, 2010, p. 27

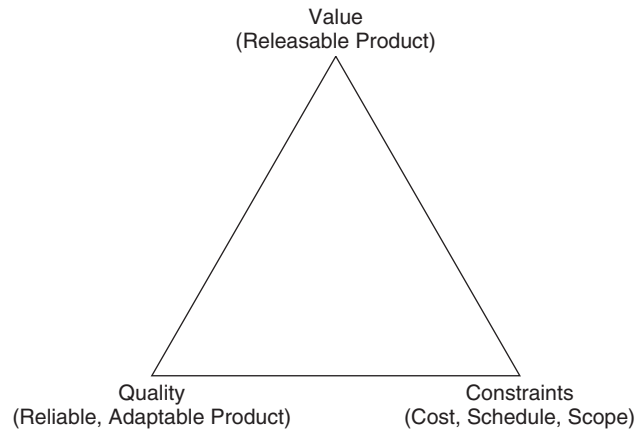


Figure 1.3 Agile Project Management Triangle

Jim Highsmith in his book *Agile Project Management*¹² proposes an agile triangle shown in Figure 1.3.

In an agile project, there is a much stronger focus on producing value for the customer. The customer is directly involved in the development effort, and it is understood and *expected* that the customer is going to introduce changes throughout the project to try to optimize the business value the project produces. In today's world:

- It may be unrealistic to expect that users can totally predict all the business requirements for a project far in advance of when the project will be delivered.
- In many cases, business results and software application solutions are so intimately intertwined, that achieving successful business outcomes requires a much more collaborative and integrated approach.

The trend towards more agile project management raises some important questions for project managers and any organization that depends on effective project management discipline for successful implementation of product development projects. For example:

- How does “agile” impact existing project methodologies? Is there still a need for traditional plan-driven development approaches?
- How do I reconcile all the traditional project management practices, such as those in the Project Management Institute's *A Guide to the Project Management Body of Knowledge—Fourth Edition* (PMI® *PMBOK® Guide—Fourth Edition*), which has been the foundation of project

¹² Highsmith, Jim, *Agile Project Management-Creating Innovative Products*, New York: Addison-Wesley, 2010, p. 21

14 Introduction

management for so long, with many of the new ideas and principles that are the foundation of the agile movement?

- What is the impact of the agile movement on the future of project management? How does it change the project management role? What skills are needed for a project manager to be effective in an agile environment?

The answers to these questions are not immediately obvious, and these are very important and strategic questions for businesses, as well as the project management profession.

COMMON AGILE MISCONCEPTIONS

Our understanding of agile methodologies is rapidly evolving; however, there are a number of misconceptions that seem to persist about agile development. It is essential to clear up some of these misconceptions in order to objectively understand the benefits, limitations, and requirements of implementing an agile strategy.

The Pizza Box Methodology

There are people who think that being agile is nothing more than some business-people sitting down with some developers over a box of pizza and starting to write code with no structure, process, tools, or methodology for how it is done. The truth is that being agile does require a planned approach and a very disciplined process for doing the work, and in a very fast-paced development environment, having appropriate tools to manage the effort can become imperative.

“Agility without discipline is the unencumbered enthusiasm of a startup company before it has to turn a profit. Great companies have both (discipline and agility) in measures appropriate to their goals and environment.”¹³

An effective implementation of an agile approach can require a *much higher* level of skill and self-discipline from everyone on the project team:

“Although it may seem counter intuitive [sic], agile is an extremely disciplined approach to working. Agile does not equal sloppiness. Most people will have a difficult time adjusting to this.”¹⁴

¹³ Boehm, Barry and Turner, Richard, *Balancing Agility and Discipline—A Guide for the Perplexed*, New York: Addison-Wesley, 2003, p. 2

¹⁴ “‘Gotchas’: Common Pitfalls when Moving to Agile,” Sapient Corporate White Paper, www.sapient.com

All-or-Nothing Thinking

There are people in the agile community who are very zealous and aggressive about promoting the benefits of Scrum and Extreme Programming (XP) to the extent that it seems to be a black-and-white proposition—either you fully adopt those specific agile approaches or you’re not agile at all. The truth is that:

- The principles behind agile methodologies, including Scrum, can be customized and tailored to fit different kinds of projects and can be used in combination with other project management approaches as necessary to provide a balance of control and agility.
- There are many alternative approaches between pure forms of agile at one end of the spectrum and traditional Waterfall development approaches at the other end, and there are many ways an organization can become *more agile* through incrementally adopting new development practices and approaches without necessarily going to pure forms of agile, as shown in Figure 1.4.

Traditional Development Approaches Are Dead

There is also a misconception that traditional methods of doing development and all the practices and tools that have been commonly associated with them are either obsolete or irrelevant and are completely replaced by agile methodologies, practices, and tools. The truth is that:

- There are many good reasons why traditional plan-driven development approaches still can make sense for a given project, depending on the risk and complexity of the project and other factors.
- There are a number of ways to apply agile principles to traditional methodologies in order to make them more agile.

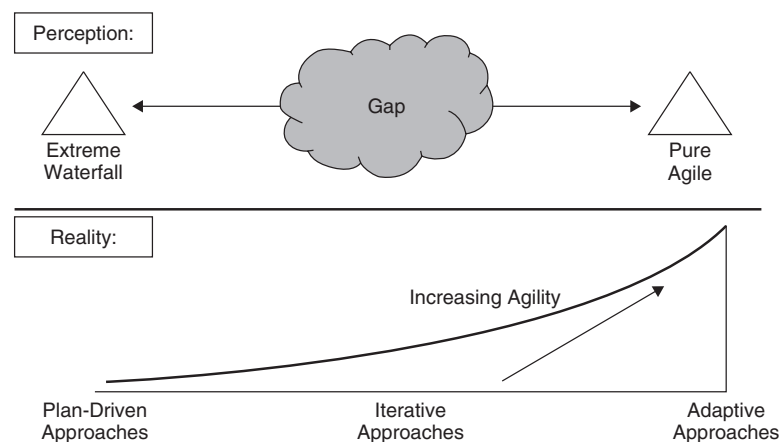


Figure 1.4 All-or-nothing thinking

16 Introduction

- There are lots of ways to apply more traditional practices and tools to agile projects to achieve higher levels of control and predictability if necessary.

Just Do It Faster

Software development projects are notorious for becoming “death march” projects. Edward Yourdon defines a “death march” project as:

“Quite simply, a ‘death march’ project is one whose “project parameters” exceed the norm by at least 50 percent. In most projects, this means one or more of the following constraints have been imposed upon the project:

- The schedule has been compressed to less than half the amount estimated by a rational estimating process; thus, the project that would normally be expected to take 12 calendar months is now required to deliver its results in six months or less . . .
- The staff has been reduced to less than half the number that would normally be assigned to a project of this size and scope . . .
- The budget and associated resources have been cut in half . . .
- The functionality, features, performance requirements, or other technical aspects of the project are twice what they would be under normal circumstances . . . ”¹⁵

Basically a “death march” project amounts to pressuring project teams to just do the same work faster without necessarily changing anything about the process and methodology of how the work is done. It’s a brute force approach to compress the project schedule, which is fraught with many potential problems.

John Wooden is a famous American basketball coach at the University of California at Los Angeles (UCLA). “In Coach Wooden’s last twelve years as coach, UCLA won ten National Collegiate Athletic Association (NCAA) championships. In the 27 years he led the Bruins, they never had a losing season. Their record of 88 consecutive winning games will probably never be surpassed.”¹⁶ He is noted for saying “Be quick, but don’t hurry”.¹⁷ Being quick requires discipline and training, hurrying can be just a brute force way to try to get it done faster and is likely to have a much lower success rate.

Becoming Agile Only Impacts the Development Organization

Many people think that becoming agile only impacts the product development organization. This is a quite common misconception on the business side—if we

¹⁵ Yourdon, Ed, “What is a Death March Project and Why Do They Happen?”, www.informit.com/articles/article.aspx?p=169512

¹⁶ John Wooden, Academy of Achievement, www.achievement.org/autodoc/page/woo0pro-1

¹⁷ “The Wizard’s Wisdom: Woodenisms,” <http://sports.espn.go.com/ncb/news/story?id=5249709>

can just get our development organization to be more agile and speed up the way they do things, it will solve all our problems. The truth is that becoming more agile impacts much more than just the development side of the organization—it requires a broad-based commitment from the business side of the organization to work in a close, collaborative partnership with the development side of the organization, and it may also require some major shifts in organizational culture and thinking to make that work.

Migration toward an agile approach without substantial sponsorship and long-term commitment from the business is probably doomed to failure. In many cases, the business side of the organization should take a major leadership role in driving the change that is needed.

“The business stakeholders involved must learn the process as it is very different for them... if this group is not brought into the fold, there will be major disconnects (in terminology, approaches to situations, etc.) Once educated, they will be able to see the benefits of being able to direct the development as it progresses. The business also needs to clearly understand the expectation that it will also be frustrating for them to see the “dirty laundry” being aired each iteration: unfinished work, team mistakes, and other issues that are often hidden in a methodology with long breaks between business reviews.”¹⁸

Agile Is Just a Development Methodology

A similar misconception is that agile is just a development methodology and not a project management methodology or framework. That perception is probably a carryover from the early days when agile was primarily a development methodology with little or no structure or process. Since that time; however, “agile” has matured significantly and Scrum is a relatively well-defined methodology and process with a significant knowledge base associated with it.

However, there are probably several key reasons why that perception still persists:

1. In an agile project, the line between what is a development methodology and what is a project management methodology is often blurred:
 - The development process may not be a discrete function separate from requirements and other project management aspects of the project—they are typically very well integrated.
 - The project management methodology may not be as formally defined as a separate process from the development process, and in many cases, agile methodologies don’t use the term “project manager” at all.

¹⁸ “‘Gotchas’: Common Pitfalls when Moving to Agile” Sapient Corporate White Paper, www.sapient.com

18 Introduction

2. Agile methodologies are meant to be building blocks and combined and extended as necessary to create a complete project methodology. Agile does not specifically define all of the higher-level planning and project management that might be necessary for larger, more complex projects, but it would be inaccurate to characterize it as just a development methodology.

WHAT AGILE DOESN'T TELL YOU

There are a couple of key reasons why there are misconceptions about agile:

1. Agile methodologies, by design, are not prescriptive—they don't tell you exactly what needs to be done to implement the methodology or how to do it in many cases. In general, agile methodologies define some principles that intentionally require interpretation for a given situation. For example, the Agile Manifesto defines four values and twelve principles. The four values are as follows:

“We are uncovering better ways of developing software by doing it and helping others do it. Through this work we have come to value:

- Individuals and interactions over processes and tools
- Working software over comprehensive documentation
- Customer collaboration over contract negotiation
- Responding to change over following a plan

That is, while there is value in the items on the right, we value the items on the left more.”¹⁹

It's up to the person implementing agile to interpret what those values mean in a given situation and determine how to apply them in the context of his/her business and project environment. In some cases, they've been misinterpreted. For example, some have taken an extreme view to interpret them to mean that:

- There is no documentation, and no processes and tools.
- Agile is not consistent with environments based on customer contracts.
- Change control is not consistent with agile.

It certainly wasn't the intent of the people writing the Agile Manifesto that the contents would be interpreted as absolutes²⁰— they were intended to be relative statements. The key point is that you have to interpret those principles in the context of your business and project environment and

¹⁹ Agile Manifesto, <http://agilemanifesto.org>

²⁰ Highsmith, Jim, *Agile Project Management-Creating Innovative Products*, Addison-Wesley, New York, NY 2010, p. 16

apply them intelligently. Unfortunately, that has not always been the case and agile methodologies have been misapplied or poorly implemented in some situations.

2. Agile methodologies are still maturing, and there's a lot to be learned about what works and what doesn't work. In fact, agile is heavily based on continuous improvement, the whole methodology is meant to evolve, and there is an evolution taking place as the body of knowledge associated with agile grows. Jim Highsmith²¹ breaks down the most important agile tools and techniques fall into the following levels:

- Technical Practices
- Iteration Management
- Project Management
- Portfolio Governance

Agile has its roots in technical practices and that area is the most mature and most established in terms of having a widely accepted body of knowledge associated with it, but as you move up the ladder, the levels of knowledge associated with the other areas are progressively less defined. For example, a few books have touched on the portfolio governance area, but there is much to be learned in that area.

Traditional Project Management Office (PMO) organizations need to change radically in some cases to adapt to agile. In many cases, instead of being only an enforcer of rigidly defined corporate processes and policies about how to manage projects, PMO organizations need to shift their focus to a value-added consulting orientation to help their companies learn to successfully implement more flexible and adaptive processes.

The important thing about agile is that implementing it mechanically “by the book” is not the right approach, you really need to understand the principles underneath it at a deeper level to know how to apply it in the context of your business and project environment. That's why you won't find a prescriptive approach defined for how to implement every aspect of agile.

Understanding any methodology at a deeper level (understanding the principles behind it) is essential for selecting a methodology (agile or non-agile) that is consistent with your business environment and tailoring it as necessary to fit specific projects. That is the approach behind this book.

²¹ Highsmith, Jim, *Agile Project Management-Creating Innovative Products*, Addison-Wesley, New York, NY 2010, p. 78

