REAL

Practical Guidance for Agile Practitioners

W RLD

AGILITY

DANIEL JAMES GULLO

Foreword by **STEPHEN DENNING,**
author of *The Leader's Guide to Radical Management*

REAL WORLD AGILITY

This page intentionally left blank

REAL WORLD AGILITY

Practical Guidance for Agile Practitioners

Daniel James Gullo

Addison-Wesley

Boston • Columbus • Indianapolis • New York • San Francisco • Amsterdam • Cape Town
Dubai • London • Madrid • Milan • Munich • Paris • Montreal • Toronto • Delhi • Mexico City
São Paulo • Sydney • Hong Kong • Seoul • Singapore • Taipei • Tokyo

The author and publisher have taken care in the preparation of this book, but make no expressed or implied warranty of any kind and assume no responsibility for errors or omissions. No liability is assumed for incidental or consequential damages in connection with or arising out of the use of the information or programs contained herein.

For information about buying this title in bulk quantities, or for special sales opportunities (which may include electronic versions; custom cover designs; and content particular to your business, training goals, marketing focus, or branding interests), please contact our corporate sales department at corpsales@pearsoned.com or (800) 382-3419.

For government sales inquiries, please contact governmentsales@pearsoned.com.

For questions about sales outside the U.S., please contact intlcs@pearson.com.

Visit us on the Web: informit.com/aw

Library of Congress Control Number: 2016939876

ISBN-13: 978-0-13-419170-6
ISBN-10: 0-13-419170-6
Text printed in the United States on recycled paper at RR Donnelley in Crawfordsville, Indiana.
First printing, July 2016

To Katie, Fifi, Auggie, Vinnie, Eben, and Biggie—
I love you more
than you will ever know . . .

To MnP, thanks

To Agilists everywhere,
don't let the bastards grind you down

This page intentionally left blank

Contents

Foreword

Real World Agility provides us with a clear and simple Baedeker to the sometimes strange and often baffling world of Agile.

It will be particularly useful to anyone coming into the world of Agile, Scrum, Kanban, and Lean who is finding it a disconcerting and even baffling experience. There is a lot of weird terminology that is not taught in business schools, at least not yet. There are in-jokes about things that you are supposed to know but somehow don't. You are finding some things being done in ways that are the opposite of what you have seen in other organizations. You may have heard that Agile is not for the faint of heart, and yes, you are finding that some aspects of Agile are indeed difficult. You are sensing that some parts of Agile don't fit your own organization. Can you be sure that dedication and persistence pay off? How do you find your way?

None of this would matter if you had a wise and reliable mentor who was at your side at every moment and always ready to answer every possibly dumb question with patience and insight. But you don't have that mentor. But what, you might ask, if that mentor existed and answered your questions in the form of a book? What if the book was written in an informal, lively, down-to-earth idiom—as one real world practitioner to another?

Meet *Real World Agility*, a book that gives you the answers that a wise mentor would have given you, if you had one. Drawing on his long and deep experience in the real world of Agile, Daniel Gullo shares with us his insights on the principal questions that everyone coming to the world of Agile will inevitably encounter.

There are answers to informational questions about the nature of the Agile landscape. What exactly is Agile really about? How do you make sense of all the different flavors of Agile? What's the meaning of certification? Is it useful?

It tackles big controversial questions. Can Agile scale? What are we to make of SAFe? What do you do about an organizational culture that is unsupportive of Agile?

It makes sense of bothersome little issues that every newcomer will come across. Why, for instance, all the in-jokes about pigs and chickens?

It tackles some of the trickier operational matters. How do you break down a backlog into sprints? Why must each increment of code be valuable to the user? How long should sprints be? What are story points all about? What's the fuss about the meaning of "done"?

It offers suggestions on broader management issues. How do you mesh Agile teams with the existing management hierarchy? How does the ScrumMaster relate to the product owner or project manager? Can a ScrumMaster be a member of the team? Why are there so many meetings in Scrum?

It tells the story of notable Agile journeys by a wide variety of individuals. In this catalog of narratives, readers are sure to find analogies to their own situation and its issues and, more important, discover how those issues were resolved.

It contains a handy glossary of the most common Agile terms and a reading list of books for those who want to dig deeper into the world of Agile.

The book covers a huge terrain. It offers insights on controversial issues where other authors might hesitate to tread. Not every expert will agree with every opinion that the book offers, but everyone will agree that the opinions offered here are a useful anchor point for meaningful conversation.

The book gives us frank advice on what's important and what's not. It tells us what's crucial to pay attention to, not just nice to have—but in fact the whole ball game.

There is no longer any need to go on struggling your way through Agile alone. The subject can be understood. It is the future. This book will help show you why.

—Stephen Denning
Author of *The Leader's Guide to Radical Management*

PREFACE

> *"What is the one burning question that you have on your mind about Agile/Scrum right now?"*

For more than eight years I have been coaching and training individuals, teams, and organizations in Agile product development, and I have noticed that in spite of the vast number of books on various topics in Agile, people still have questions that seem to be unanswered. I became intrigued with the idea of capturing these questions to see if there was anything that I could do with them. Are there any patterns? Any usable data that could be crunched and used?

I started using the game of "35" to elicit these questions from class attendees, workshop attendees, user group meetings, etc., and then adding them to my question bank in Excel. The result from the last two years has been a list of over 2,000 questions from entry to mid-level practitioners that cover a broad spectrum of Agile.

Although the nature of the questions varies widely, two things were clear to me: 1) many of the same questions continue to appear time and again, and 2) people are looking for practical, directive advice on what they see as practices from a vast body of largely theoretical writing.

I resolved to address these questions in a book of my own in a style that is practical, based on my own real-life experience over the last 10 years with a somewhat humorous, matter-of-fact, and down-to-earth bent. The driving vision in my mind has been: "What book would *I* want to read if I were starting off or had mid-level experience with Agile?"

Throughout the book, I have attempted to give some of the background as to WHY we do things a certain way in Agile, while also offering some examples from my own experience. When I started my efforts on this book, I found myself citing this client or that, this employer or that, in a way that quickly became tedious and really created a narrow focus for certain topics. I wouldn't want someone reading this book to say, "Oh, well, we're nothing like XYZ company so I won't pay this passage any heed."

Instead, I share my list of organizations with whom I have worked at the end of this preface and assure the reader that all of the writing in this book derives from my direct experiences with these organizations. Also, because of nondisclosure agreements (NDAs) and such, I am technically not allowed to talk specifically about things at some of these organizations. Who I have worked for is a matter of public record, and as long as I don't mention specific practices being used at a specific organization, there is a reasonable level of abstraction from a legal standpoint.

When I am coaching individuals and organizations, I also emphasize that much of Agile involves working things out on your own and making sense of things for your own organization. Because Agile is also largely about freeing the minds of the organization and establishing a culture dedicated to learning, I am reminded of that scene in Monty Python's *Life of Brian* when Graham Chapman as Brian is addressing the massive throng who has congregated outside his home:

> **Brian:** "Listen, you don't need to follow me, you don't need to follow ANYBODY. You can all think for yourselves!!"
>
> **The Crowd:** "Yes! Yes! Tell us more!!"
>
> **Brian:** "No! NO! That's the point, you are ALL individuals!"
>
> **One voice in the crowd:** "I'm not."
>
> **The Crowd:** "Shhhh!!!"

At some point along the line in your Agile journey, you have to recognize that there are no absolute answers. There are only values, principles, practices, patterns, and ideas that serve as hints to the REAL point of all this—to break free of group-think and of standing on convention for the sake of standing on convention and to take a bold stab at being creative and innovative by taking risks and running the experiment . . .

Blessing and peace.

Daniel Gullo

Real Experience

My Agile journey began back in 2006. I was managing the QA practice for a firm in Wilmington, Delaware, and one day, a valued client of mine handed me Ken Schwaber's book *Agile Software Development with Scrum*. He asked if I had read it and began to rattle off a bunch of what sounded like gibberish to me at the time.

I was a hardcore command-and-control PMP with my eyes set on clawing my way up the rigid corporate hierarchy in my organization, or, even better, within one of my high-profile financial services clients once they saw just how "Gordon Gekko" I was . . .

I remember just standing there smiling, nodding, and thinking to myself "Scrum? What the hell is that?? Sounds dirty. Like, what, SCUM? What does that have to do with software development?" I had no knowledge or interest in rugby and was more interested in what Cornelius Fichtner, Rex Black, Capers Jones, and Ivar Jacobson had to say than . . . Ken who? I took the book and made a promise to let the client know what I thought of it within a week or so.

It was love at first sight.

What I read not only made perfect sense to me, but I saw it as the way of the future—the language of innovation but not innovation itself. "Scrum is a catalyst, an enabler . . ." I thought. I also experienced an epiphany as I realized that I hadn't been comfortable for the last 17 years of work because I was forcibly molding myself into something that wasn't really in line with my soul.

I was hungry.

I devoured book after book on Agile. I went to CSM training and insisted on taking my course with Ken Schwaber himself. Tom Mellor was assisting Ken with the class. I remember about 70 people playing "manager and employee" trying to take 60 steps around the conference room. I remember the Major League Baseball ticketing system simulation. I was like a kid in a candy store. I had never had certification training that was actually fun AND practical AND informative.

In addition to managing all of the QA engagements for the company, I began going on billable assignments myself teaching people about Scrum. I acted as the interim ScrumMaster, and "coached" the organizations on how to use the feedback loops of Scrum to improve and explained why it was a good thing to have cost and time be fixed but scope be flexible in order to accommodate change.

Since 2007, I have trained, mentored, coached, and advised many organizations as both a full-time employee and consultant (see Table P-1). This is the experience I have drawn upon in the writing of this book. While I can't divulge specifics of my work with many of these organizations, I feel it's my obligation to share where my experience comes from in aggregate to lend credibility to the advice that I have provided here.

You see, I live in the real world, too. Over the years, I have had students who, because of their inability to see what's possible or their lack of vision or courage, have taken to patronizing me with the claim that they "live in the real world." It often helps when I tell them stories of my real world experience implementing REAL WORLD AGILITY . . .

TABLE P-1 Organizations I Have Worked with (or for) since 2006 Doing Agile Training/Coaching (in alphabetical order)

ABSOLUTE SOFTWARE	PAYPAL
ACCESS GROUP	PETROLEUM HELICOPTERS INTERNATIONAL, INC.
ADP	PFIZER (FORMERLY WYETH PHARMACEUTICAL)
ARKIEVA	PHI HELICOPTER
CAPITALONE (FORMERLY ING DIRECT)	PICTOMETRY
COMCAST	PROJECT MANAGEMENT INSTITUTE (PMI)
CREDIT ACCEPTANCE CORPORATION	ROHM AND HAAS
CSL BEHRING	SIEMENS
FEDERAL RESERVE BANK	SNAP-ON
FIDELITY INVESTMENTS	T. ROWE PRICE
GENERAL ELECTRIC (GE)	TAXANALYSTS
GENERAL MOTORS (GM)	UNITED STATES POSTAL SERVICE (USPS)
INTERNAL REVENUE SERVICE (IRS)	U.S. TREASURY DEPARTMENT
INVISTA	VWR INTERNATIONAL
MICROSOFT	WILMINGTON TRUST
NAVTEQ	

Register your copy of *Real World Agility* for convenient access to downloads, updates, and corrections as they become available. To start the registration process, go to informit.com/register and log in or create an account. Enter the product ISBN (9780134191706) and click Submit. Once the process is complete, you will find any available bonus content under "Registered Products."

ACKNOWLEDGMENTS

This book has been a monumental undertaking for me—one of the most difficult things I have ever faced in my career.

One would think that simply writing down what is in the mind would be an easy thing to do. And a braver soul than mine might have been able to do so with reckless abandon. However, for me it was necessary to balance raw emotion, passion, humor, and a flair for hyperbole with so-called professionalism, decorum, and, moreover, practical guidance.

And so I am compelled to thank the short but incomplete list of those who have inspired, motivated, and supported me through its creation.

Beginning with my soul mate, best friend, wife, mother of my children, and chief muse, Katie, who has never doubted me in spite of all the times that I have.

To my good friend and commiserater, Dave Prior, who has always been the voice of sanity in an insane world.

To my mentor and role model, Mike Cohn, who has regularly helped me to focus on the brass ring in spite of what the haters say (who aren't even on the same carousel).

To my partner in crime, Stephen Forte, leave the gun, take the cannoli. . .

This page intentionally left blank

About the Author

Daniel James Gullo is a prolific contributor to a vast array of online forums, blogs, and other social media, including a network of 8,000 connections on LinkedIn and 25,000 followers on Twitter.

Daniel has given much of his free time to mentoring and coaching candidates for CST and CEC and has received accolades for his efforts in these areas from the people whom he has mentored and his colleagues in the training and coaching community.

Daniel is also a trusted advisor to staff and management of the Scrum Alliance on matters related to policy and strategy, including but not limited to the Events program, CST program, CEC program, and community outreach efforts.

Daniel serves on the Trainer Acceptance Committee (TAC) for the Certified Scrum Trainer (CST) certification program and as a reviewer of Certified Enterprise Coach (CEC) applications. Daniel is a founding member of the Scrum Coaching Retreat Planning Committee; host of Coaches Clinic events; and facilitator of Open Space Events, including the Scrum Alliance's largest Open Space ever at the Scrum Gathering in Berlin (with over 500 people) and the Scrum Gathering in Shanghai.

Daniel was conference chair for the 2015 Scrum Gathering in Phoenix. He was also the conference chair for the 2013 Scrum Gathering in Las Vegas, which makes him the only individual in the community to serve twice in this capacity. He has also coached other conference chairs for both Scrum Gatherings and Agile Alliance events and is one of the administrators of the submission review system for the Scrum Gatherings.

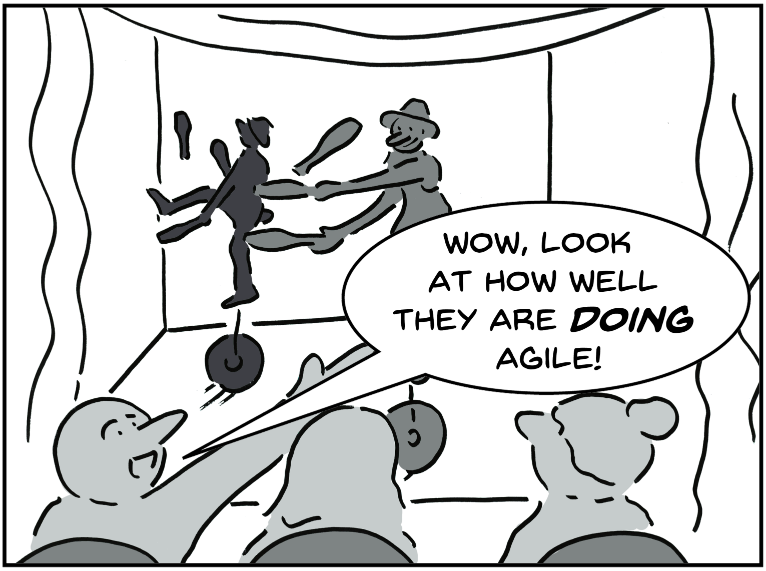
He is the founder of and chief advisor to Agile Delaware and a frequent reviewer, volunteer, and speaker for the Scrum Alliance, Agile Alliance, PMI, and other organizations, including delivering keynote addresses for conferences such as the Scrum Gathering–India, Scrum Gathering–Rio, and Scrum Gathering–China.

As founder and principal of Apple Brook Consulting, he has firsthand experience with what it takes to make business work. A lifelong entrepreneur, Daniel's portfolio of clients is long and distinguished: ING Direct (CapitalOne), NAVTEQ, IRS, PayPal, ADP, U.S. Postal Service, GM, U.S. Treasury Department, T. Rowe Price, GE, and many other high-profile organizations.

Daniel is well known and highly revered as the "Hardest Working Man in Agile," a title bestowed upon him by several of his colleagues in the training and coaching community. His tireless dedication and effort has earned him the distinction of the Most Valuable Agile Professional Award for 2015.

Chapter 1

Agile in General



When people begin to learn about Agile, they struggle with the values, principles, and practices. This is largely due to the fact that our ideal of what an organization looks like is far from what we cherish as part of our human nature. Instinctively and intuitively, human beings are inquisitive, experiential, exploratory, learning driven, and so on. The corporate world tells us that we must follow the rules and procedures—we must behave and conform.

When we adopt an Agile mind-set, we are really returning to our innovative nature as human beings by looking for ways to continuously improve in spite of process and procedures.

In this chapter, we explore concepts that touch on various different areas and concepts of Agility in general. Some of these topics are Scrum related. Others address Kanban, Lean, and other approaches broadly classified under the Agile umbrella. The prompt questions and answers represent Real World examples and statements from REAL people; that is, the statements are presented exactly how they were conveyed to me by my students, clients, and others.

Waterfall Versus Agile

Waterfall was first described in 1970 by Winston Royce in his seminal work "Managing the Development of Large Software Systems" where he mentions that it is something that is " . . . risky and invites failure." Most often, people ignore or forget the key point of Royce's whitepaper, which was intended to emphasize how iterative development is more effective than waterfall.

Waterfall

> *In general, the term* ***waterfall*** *describes the nature of stage-oriented, single-pass software development methodologies. The work done in each stage of the workflow is isolated from the other stages, oftentimes with a formal sign-off process similar to the levels of a waterfall; water never travels back up to other stages in a waterfall.*

In essence, waterfall refers to phased development approaches; that is, any time there are various, distinct phases that occur in the development lifecycle with formal hand-off or sign-off procedures as a requirement for promotion to the next phase, we are talking about waterfall.

Figure 1-1 is an example of what a 12-month waterfall project looks like from a process flow perspective.

Notice that there is no tested working code until 9 to 12 months into the project. The value delivery in this case is delayed instead of occurring gradually throughout the lifecycle of the project (see Figure 1-2). If funding were cut somewhere along the way, there might not be any usable portion of the software; the entire thing would be thrown away. Also, the customers and stakeholders have no confirmation as the product is developed that they are actually receiving value for their investment or that the product will meet their expectations. They only have a "promise" of what the product will look like, what it will do, etc.

In theory, waterfall is very logical, and as arrogant and egotistical human beings, we believe that we can account for all contingencies, risks, and unknowns prior to the start of a particular development effort. However, history has shown time and again that we are notoriously bad at clairvoyance, fortune telling, seeing the future, and most of all, estimates. Our abilities to predict and mitigate risks, which include changes in customer requirements, market direction, and technical failures, are not the greatest.

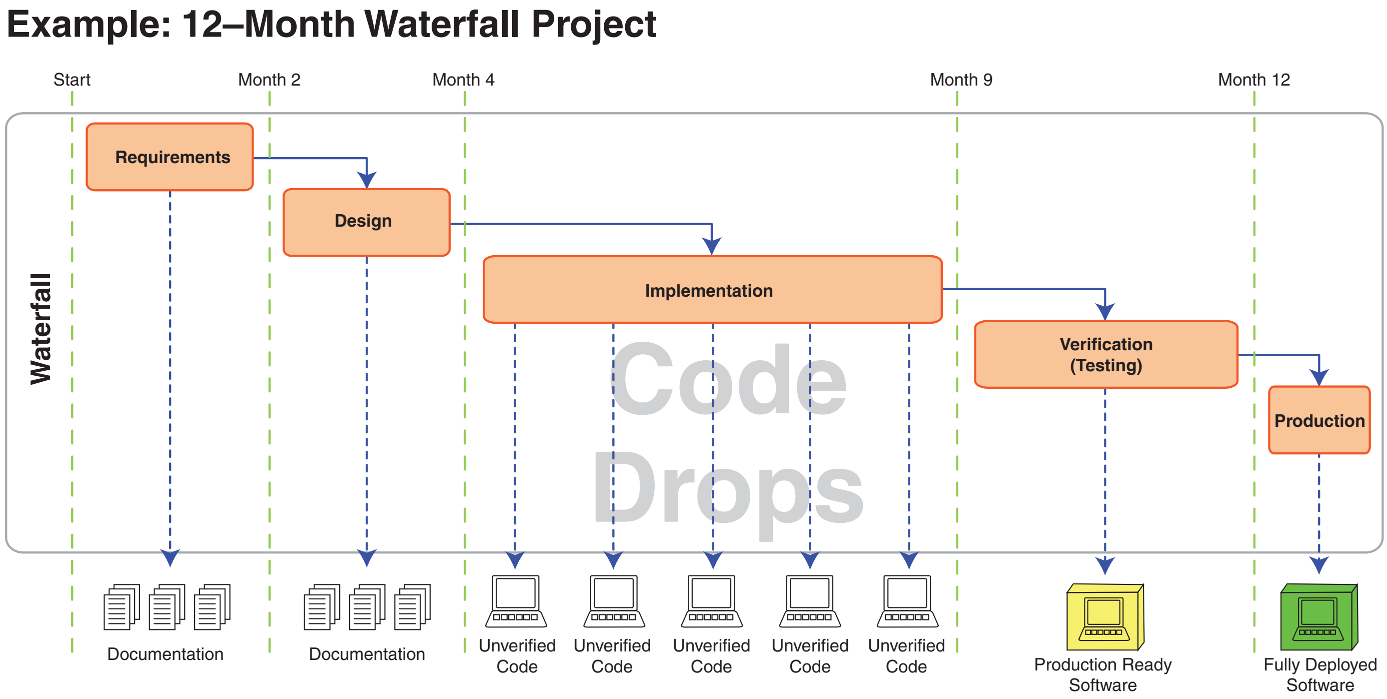


FIGURE 1-1 Waterfall delivery model

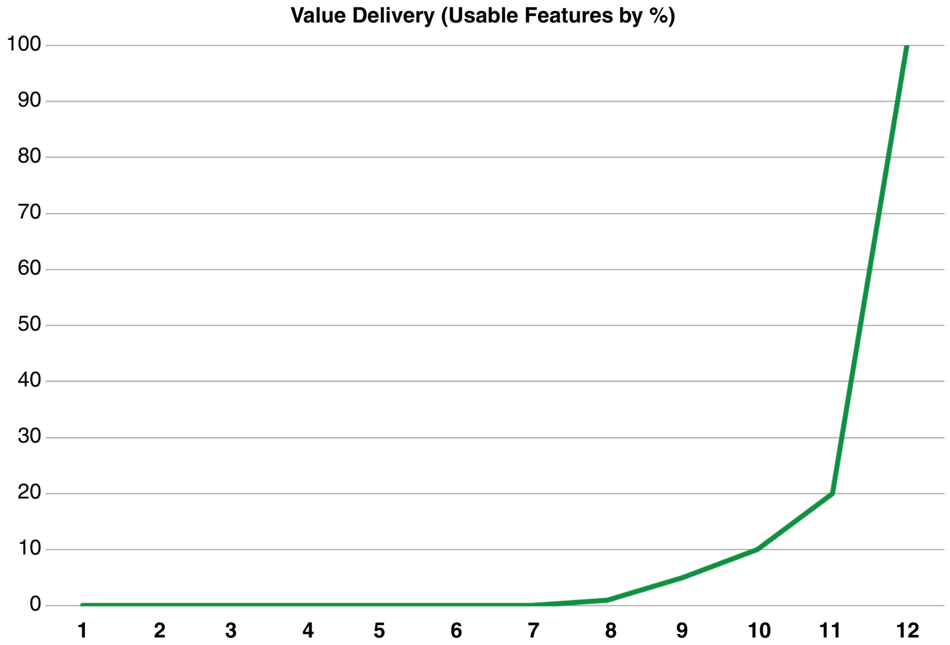


FIGURE 1-2 Waterfall value delivery

Waterfall can work just fine in an organization that has projects where there are very specific requirements with no possibility that these requirements will EVER change. In order for this to be the case, there must be absolute certainty in terms of business strategy, customer needs, and stakeholder alignment AND absolute certainty in terms of technical design, architecture, and the implementation of the solution without any risk at all that change may be involved. Figure 1-3 shows Ralph Stacey's various classifications of organization based on these parameters.

In short, waterfall works in organizations where there is no change or uncertainty, but rather complete agreement and certainty around vision/strategy and technology.

Very few organizations have absolutely no change at all or uncertainty. Instead, most organizations struggle to implement rigid, predictive processes and experience issues when they encounter inevitable changes in customer demand, technology advances, legislative decisions, societal changes, shifts in workforce dynamics, and various other factors that affect development and delivery of products and services.

To more effectively deal with change, and even embrace it as a strategic and competitive advantage, organizations are better served using an iterative/incremental

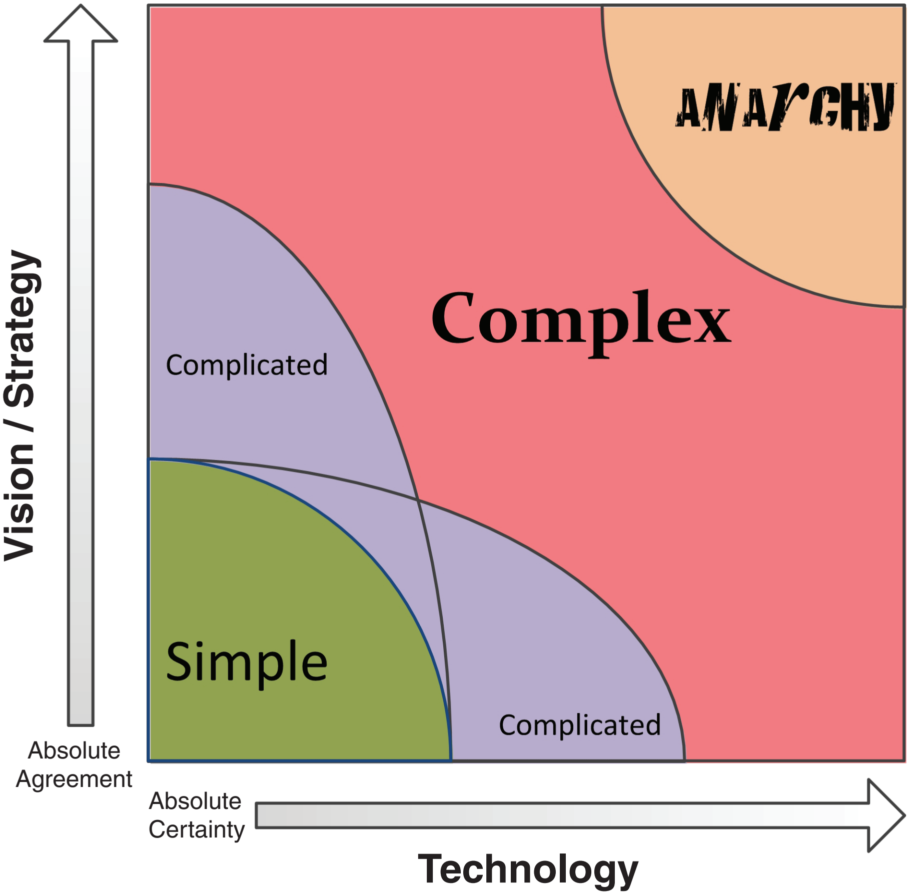


FIGURE 1-3 Stacey diagram—organizational complexity

development and delivery approach for their products and services. This approach is also known as Agile.

Agile, strictly speaking, refers to the values and principles set forth by the Manifesto for Agile Software Development (aka "The Agile Manifesto"):

> *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.

Twelve principles are also cited, which build on top of these values:

We follow these principles:
Our highest priority is to satisfy the customer
through early and continuous delivery
of valuable software.

Welcome changing requirements, even late in
development. Agile processes harness change for
the customer's competitive advantage.

Deliver working software frequently, from a
couple of weeks to a couple of months, with a
preference to the shorter timescale.

Business people and developers must work
together daily throughout the project.

Build projects around motivated individuals.
Give them the environment and support they need,
and trust them to get the job done.

The most efficient and effective method of
conveying information to and within a development
team is face-to-face conversation.

Working software is the primary measure of progress.

Agile processes promote sustainable development.
The sponsors, developers, and users should be able
to maintain a constant pace indefinitely.

Continuous attention to technical excellence
and good design enhances agility.

Simplicity—the art of maximizing the amount
of work not done—is essential.

The best architectures, requirements, and designs
emerge from self-organizing teams.

At regular intervals, the team reflects on how
to become more effective, then tunes and adjusts
its behavior accordingly.

Values and principles begin to describe and define the culture of organizations. When combined with the practices and other factors, we begin to see a full picture of the organization's culture.

Agile does not specifically prescribe any practices per se. The 12 principles point people in the right direction, and the individuals decide what they hold true and believe in. However, if the people, management, and culture of the organization are not aligned with the values and principles, then there will be difficulties implementing Agile practices, regardless of what framework is selected. Simply stated: by definition, what makes an organization Agile is its beliefs and core values, not the practices it follows.

Agile frameworks such as Scrum, Kanban, and xEtreme Programming (XP) provide some lightweight "rules" by outlining practices that will help organizations embrace the values and principles in order to become more Agile in their thinking.

Scrum, for instance, defines a minimal set of roles, artifacts, and activities. All of these elements work together to produce valuable, working software every 1 to 4 weeks. In the worst case, the Development Team is producing a shippable increment of product value once a month instead every 12 months.

In Figure 1-4, we see what a 12-month project would look like in Scrum. Notice that there is shippable code EVERY Sprint. In this case, the Scrum Team has decided to release that shippable code into production every other Sprint.

This reinforces the stakeholder(s) decision to fund this product development effort. Every Sprint, they get to see SOMETHING delivered to them, rather than yet another visit back to the "promissory note."

Imagine that you are paying someone $750,000 to build a home for you. However, the builder is telling you that you won't be able to see it, walk through it, etc., until it is completely finished. There is a significant risk that the house won't be what you want in the end. You might change your mind about cabinets or the layout of the home. The builder may have misunderstood your requirements about cabling or how to orient the fixtures in the bathroom.

Now, suppose the builder puts stakes around the property and invites you out to do a "preliminary walk-through." You notice that you would prefer to have the house rotated about 45 degrees counterclockwise on the lot. Then, when the builder has poured the foundation, you are invited out to walk the slab and basement. It looks great and meets your expectations.

As the home emerges, you have confirmation along the way that it is meeting your expectations, and there are opportunities for you to make subtle and even significant changes.

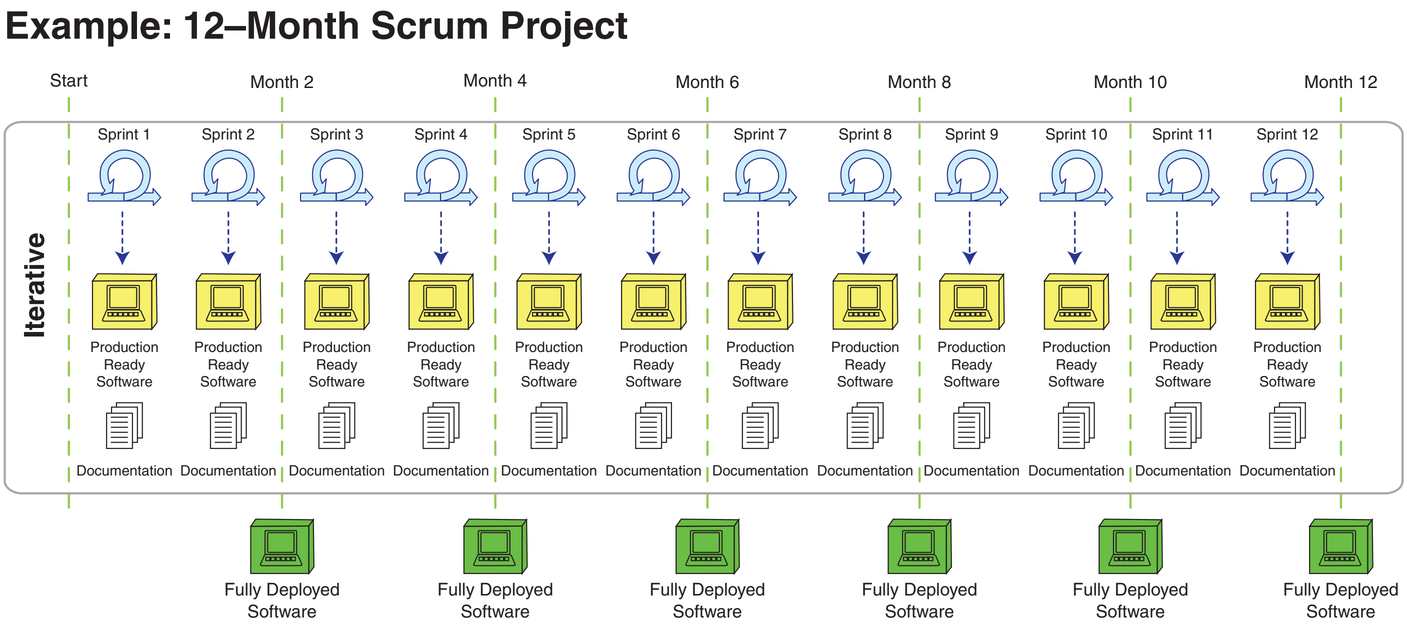


FIGURE 1-4 Scrum delivery model

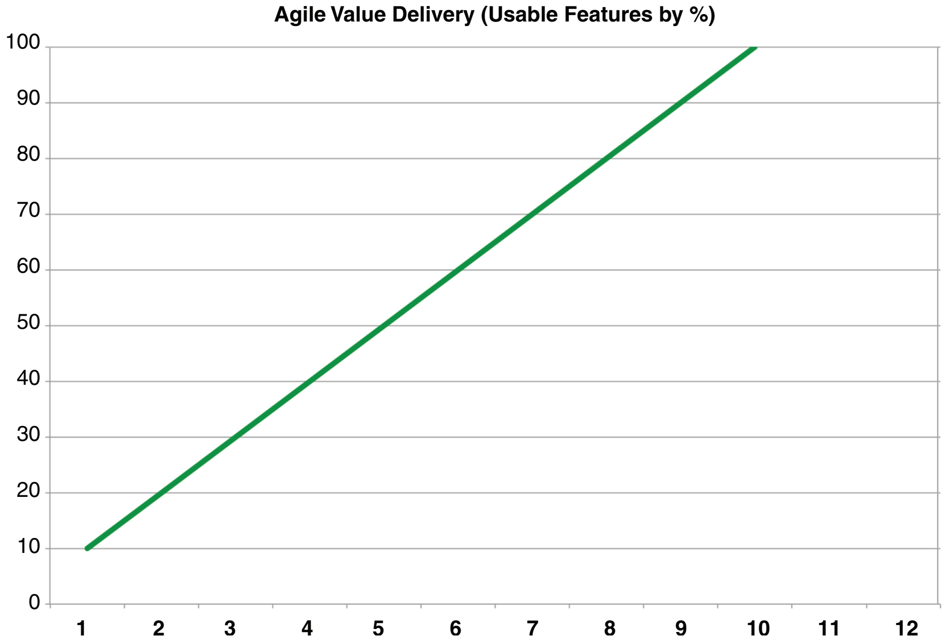


FIGURE 1-5 Value delivery in terms of usable features

This is similar to how Scrum enables customers to see the value they are receiving as it is delivered (see Figure 1-5) and provides them with opportunities to subtly change their minds along the way so that they are assured they will have what they want in the end.

Thus, the primary benefit of Agile versus traditional project delivery is that Agile provides opportunities to inspect and adapt while iteratively delivering increments of value throughout the product lifecycle, rather than just at the end.

An "Agile" Experiment

Here is an experiment I would like everyone to try.

For one full week, whenever you see or hear the word "Agile" (or "agile") in a sentence about project management, software, development, etc., substitute "Agile" with the following phrases:

. . . thinking for yourself . . .

. . . talking to people . . .

. . . producing things that people want . . .

. . . looking at what's possible . . .

. . . being less of a jerk . . .

. . . solving problems with the simplest solution . . .

. . . delighting customers . . .

. . . being a whistleblower . . .

. . . taking responsibility and initiative like an entrepreneur . . .

What's the point of this experiment?

I am seeing and hearing more and more about "Hybrid Agile," "MANAGED Agile Development," "Scaled Agile," etc., which leads me to believe that most people aren't getting what "Agile" really means, or rather, the spirit of what the word represents.

Perhaps the Agile Manifesto has become too much of a mantra? We all can recite the values. I have found that many who claim to "know Agile" haven't seen the 12 Principles Behind the Manifesto or have forgotten about them, etc. However, those also provide more of a clue as to what Agile means.

In the phrases I listed, I have tried to avoid using too many of the buzzwords, which are becoming meaningless and diluted—for example, collaboration, value, innovation, empowerment, etc. These are often tossed around without much thought of what they mean or what the implications are. People just want the benefits and R without the I.

Did some of these shock you a little bit?

Hopefully.

There IS a bit of humor intended there, but also some reality.

I am not implying that Agile means "hippie love fest," but it certainly means getting along better with the people you work with and the customers you serve.

Anyone who works serves a customer. You could even view your relationship with your employer as serving your customer.

As you try out this experiment and find that these phrases don't fit with something that is labeled "Agile," it's probably a good sign that it isn't . . .

Differences Between Agile, Lean, Six Sigma, PMP, and Other Methodologies

This is a great question that has frequently been repeated in my classes in various forms. In other words, it comes up even if the class participants do not always ask about all of these methodologies listed, sometimes including Scrum, software development lifecycle (SDLC), and others. I think it's important to first clearly define the difference between a methodology and a framework before we begin to look at these approaches and others.

A methodology is a system of methods, tools, and practices that explicitly outline phases and what must be done in a procedural manner. A framework, on the other hand, loosely defines practices and patterns, but allows for customization and flexibility in the implementation of these practices and patterns.

As we have discussed under the question "Waterfall vs. Agile," Agile itself is neither a methodology nor a framework. Rather, Agile is a collection of values and principles that represent a philosophy and a way of thinking about value delivery. Because the authors of the Agile Manifesto were pioneers of their day who were advancing their own practices in software product development, Agile is a sum-total representation of many different approaches, not the other way around. I view Agile as a way to define the unity of purpose between Scrum, eXtreme Programming, Kanban, Adaptive Software Development, Rapid Application Development, etc.—somewhat like an umbrella term to describe all that is common among these practices in terms of their beliefs and the spirit of what they are trying to accomplish.

Agile shares with Lean the idea of delivering value by reducing the amount of waste. In some ways, Lean and Agile can be virtually the same thing. However, Lean has some very specific defining elements, such as the five principles:

1. Identify value
2. Map value stream
3. Create flow
4. Establish pull
5. Pursue perfection

Along the way, Lean identifies seven wastes that can exist in any manufacturing process, regardless of the type of manufacturing:

1. Transport
2. Inventory
3. Motion
4. Waiting
5. Overproduction
6. Overprocessing
7. Defects

We then look for ways to avoid or reduce these wastes while also considering the theory of constraints around external dependencies; that is, we build systems around the issues that we cannot control and then seek to optimize system performance instead of local performance, which does not increase overall throughput.

Thus, it isn't a question of "Scrum or . . ." or "XP or . . ." These approaches are not 100 percent mutually exclusive of each other. In fact, many of the practices overlap and those that do not are complementary and compatible to a large degree.

Six Sigma (or 6σ) is a collection of practices and tools, albeit much more focused on improvements via measurement and metrics than on organic feedback loops. At the core of Six Sigma is the overall Lean goal of reducing waste and maximizing value.

However, because of the emphasis on quantitative vs. qualitative measures, many Agile proponents feel that Six Sigma is too heavyweight (and even a bit misguided in its application). Six Sigma can provide justification for changes and improvements for industries that are subject to very stringent regulatory compliance requirements.

For software systems that do not have a great risk or threat to human life or general welfare, improvements to performance and fitness of purpose are more effectively driven by customer interaction than an examination of metrics.

Kanban is framework that raises the transparency of work flowing through a workflow (system) by tracking the states of the work from beginning to completion. The key feature with Kanban is that each of these workflow states has a "work in progress" (WIP) limit, which transforms the entire workflow into a pull system instead of a push system. With a push system, it is possible for work to be continually sent downstream in the workflow in spite of a bottleneck or overflow condition. With a pull system, additional work cannot progress through the workflow until there is an opening in the next workflow state downstream. The only exception is an expedite queue, which allows violation of WIP limits in those rare instances when there is work that is critical in nature.

Kanban can be used in concert with Scrum with a degree of success, especially when Development Teams are focusing on establishing a consistent flow of features that meet the Definition of Done and are continuously deployed rather than batched into Shippable Product Increments. The Scrum aspect comes into play in order to take advantage of the improvement mechanisms (aka feedback loops) provided by Scrum Ceremonies (aka Activities). For instance, the team would select a regular interval to reflect on its own growth and maturity and to discuss ways of improving how they work together as a team. This is the Scrum notion of a Sprint Retrospective. They would also regularly reflect on the product itself to ensure that it is meeting customer and stakeholder expectations. This is the notion of a Scrum Sprint Review.

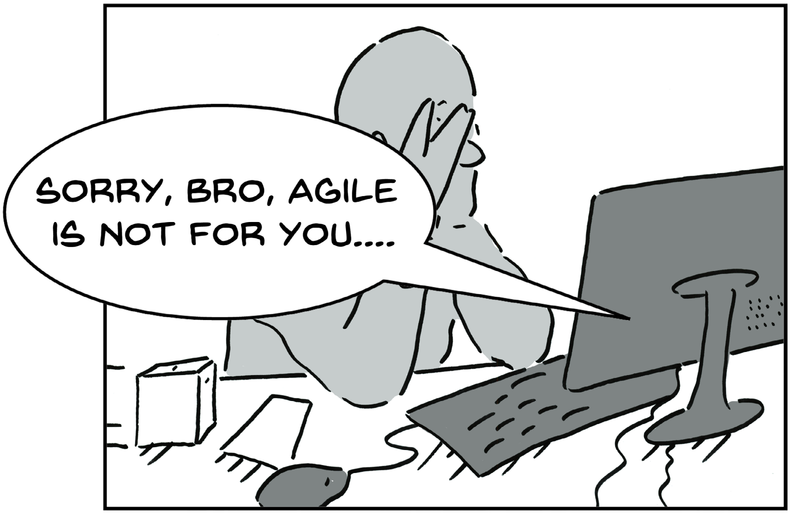
eXtreme Programming involves engineering practices and less formality around roles than does Scrum. Some of these practices include continuous integration, test-driven development, pair programming, the use of acceptance criteria, refactoring, collective code ownership, etc.

There are other sets of practices such as Rapid Application Development, Dynamic Systems Development Method (DSDM), and Crystal Clear, which have all lent their influence and inspiration as inputs to the Agile Manifesto. Even some aspects of Rational Unified Process (RUP), vis-à-vis "The New, New Product Development Game" (Takeuchi and Nonaka, 1986) and some parts of traditional project management have arguably had influence on Agile. That is, project management is not specifically mentioned anywhere, but the activities of "managing projects" happen inherently and as needed throughout the various practices of Agile.

More important than any of this is the need for organizations to establish cultures that embrace change, learning, compassion, responsible growth, curiosity, experimentation, play, margin (à la Andy Stanley in *Take It to the Limit*) as a definition of sustainable pace, and other key dynamics that improve the quality of life for customers, workers, and the world at large. Organizations that embrace learning and promote the free exchange of ideas without fear of negative consequence excel in creating innovative products that delight their customers.

According to Steve Denning in his book *Radical Management* the ONLY thing that matters is the pursuit of customer delight, which can be accomplished through the practice of continuous innovation. Denning concludes that focus on profits, costs, and shareholder value does not produce the desired result of increased profits. If we focus on delivering products that delight the customer in every way, profits will naturally follow.

Agile Is NOT for You . . .



You are talking with two consultants about how to transform your organization into an Agile company so that you can go faster and keep ahead of your competitors.

You explain to them that your stakeholders have a need to know what is going on, and to that end, there are reports that need to be created. Furthermore, your product is governed by Sarbanes-Oxley controls, which absolutely MUST be followed. Funding for your projects is allocated 2 to 3 years in advance and is based on detailed estimates, which are in turn derived from the Work Breakdown Structure (WBS) for the project. None of this can change.

You continue to explain that the teams are geographically distributed across four continents and 10 time zones and that restructuring the teams and bringing them together, even just for release planning, is not possible. Your organization also has no budget to buy things like webcams, additional monitors, team rooms, or even open wall space.

The ScrumMaster in your organization will be in charge of 10 to 15 teams and may have other responsibilities as well. Also, the Product Owner is really in charge of an entire business unit and won't be writing any User Stories, so the BAs will need to be a Product Owner proxy for each team.

After spending 20 minutes relating all of this (and more) to the consultants, including how none of this is negotiable, you ask, "So, what are your thoughts on the best way to implement Agile here?"

There is silence.

After what seems like an eternity, one of the consultants clears his throat and says, "Agile is NOT for you."

Did you really just hear that??? Doesn't this guy get it? Is he really so independently wealthy that he is going to throw away the money that is on the table??

There's a saying that goes: "If you didn't get the joke, then the joke was not for you."

If your company is not open to change or trying new things or running experiments in order to learn, then Agile is NOT for you.

If your organization is not interested in having happy employees by focusing on people, or they aren't willing to make an investment in tools or to look at simple measurements of customer satisfaction such as the Happiness metric or Net Promoter Score, then Agile is NOT for you.

If you are not willing to explore what "possible" really means and thus have an open mind to doing the unconventional, then Agile is NOT for you.

I'm sorry. I really am.

Agile is not magic. We can't produce something from nothing or make other trade-offs go away. In order to get X, then you must do Y. You can't expect to maintain the status quo AND improve. It's simply not the real world.

Agile is all about embracing the uncertainty of change and learning how to use it to your advantage.

As a consultant, I often test the waters a bit when going through the discovery stage with a new client. I might say something that represents a worst-case scenario to see if they are prepared to go there if it comes to that. I also ask a lot of questions about how they think about people, constraints, etc.

My educational background was in law. I am inclined to look at possibilities. I often find myself in a workshop or training session orating as if I were in court:

> "In your expert opinion as a senior software developer, is it POSSIBLE that you could build production-ready features, albeit very small slivers, that are capable of functioning from end to end by cutting through the entire architecture?"
>
> "No. We can't produce anything of value in less than 6 weeks."
>
> "So, it's NOT POSSIBLE to release a single field on a web page with a submit button that applies some business logic and then inserts a value into a table in a database that only includes that field? That's absolutely NOT POSSIBLE??"
>
> "Um, well, yes."
>
> "I rest my case, your honor."

Becoming Agile means being open to possibilities and options.

In a sense, BEING Agile is like acknowledging and understanding what innovation truly means in the same sense that an artist understands what creativity means. Is someone who simply slaps paint on a canvas with no understanding of what they are doing considered an artist? Most of us would say that they are not.

Likewise, I can explain the values, principles, practices, and dynamics of an Agile culture to someone, but I can't tell them how to be innovative. That's something that has to come from within.

It's uncomfortable, change.

And, through discomfort, we learn and grow.

If you are comfortable with how everything is going, then you aren't learning.

If you aren't comfortable with the prospect that Agile is going to make you uncomfortable, then sorry, Agile is NOT for you . . .

Marketability of Scrum Certification and Consistency of Employment

There are varying opinions on professional certifications in general—not just Scrum-specific or even Agile certifications. We have to consider a few different important points when thinking about certifications.

First, all certifications, licenses, degrees, and other formal declarations are limited in what they claim to prove. At a minimum, a doctor must be an MD to practice medicine. What does this advanced degree REALLY prove? Honestly, nothing more than the fact that the individual has attended the basic training, course of study, practicums, and tests to be eligible for the license. There is no test in the world that assesses accurately how good a person will be at their job, whether they really care about others and the world, or whether they are a completely ignorant moron, jerk, etc.

I could be a brilliant doctor but have horrendous bedside manner, like Gregory House, from the TV show *House*. No one likes House. No one seeks House because they really WANT to—they seek him out when they absolutely HAVE to. The same is true with attorneys and many other professions.

Certifications are nothing more than a statement of minimum qualifications, not maximum potential. Training lays the foundation; coaching enables thrivability. If someone simply attends a Certified ScrumMaster (CSM) course but spends no additional time reading, learning, experimenting, etc., then the training was fairly useless. They will have enough information to survive in their workplace and nothing more . . . hopefully.

However, if the person develops and cultivates a thirst for knowledge and an appetite for information and discovery, then they will not only survive but thrive in the workplace. As with most things in life, you get out of it what you put in.

Second, let's consider the perspective of folks who are making hiring decisions. There are literally hundreds of thousands of applicants in the marketplace these days. I oftentimes hear people bash certifications because they say that it's a racket and that organizations should just talk to candidates and learn what the candidate knows to make a hiring decision and not base it on anything certification related. I mostly agree that the FINAL decision to hire or not needs to be based upon more than just the certification. Absolutely.

The Scrum Guide doesn't say much (or anything really) about hiring. According to the Scrum Alliance, the hiring decision is "outside the scope of Scrum," which I think is technically true but a cop-out. I believe that the Scrum Team, by virtue of the fact that they are supposed to be self-managing and self-organizing, should collectively be deciding who joins their ranks.

However, here's the deal. The people on the Scrum Team are usually so busy with deliverable work for the product that they simply would not have the time to have a lengthy conversation with hundreds of thousands of candidates who CLAIM they have the experience necessary. So, they still rely upon HR departments to weed through the applicants and come up with a short(er) list of likely candidates.

So, certifications are kind of like a ticket to the dance. You may be a great dancer, good looking, charming, etc., but you don't get into the party without a ticket and certainly no chance of dancing with anyone unless you are inside where the music is. You could start your own dance—and some people do. That's great.

I know some really brilliant people who absolutely refuse to get certified on the sheer principle of it. But then, they are constantly asking me if I can help them get ScrumMaster or coaching jobs. I usually tell them, "First, help yourself. Then I will help you as best I can."

In terms of marketability, as with most certifications (and products/services in general), there is a definite lifecycle involved based on diffusion of innovation. That is, I think about E.M. Rogers' "Categories of Innovativeness," which talks about how most of the value and opportunity lies within the first half of adoption after the Innovators have taken the risks and sufficiently socialized the idea so that a wave of Early Adopters begins to take the idea viral (see Figure 1-6). At that point, the Early Majority finally joins the bandwagon when it's "safer," but also when the impact of the idea has begun to wane. By the time the Late Majority realize that the idea is not just a fad, it has become something that they MUST HAVE or they are left behind. And, of course, the Laggards are those who are clearly left behind.

Most of the competitive advantage (marketability) of any certification is going to be within the Early Adopter phase, when the market is beginning to learn the value of the certification but few people have it. It's difficult to become involved as an Innovator unless you are involved in the actual creation of the certification or are close to those who are developing it.

Those who delay certification until the Late Majority phase will find that their efforts in job searching may be hampered by the fact that they aren't seen as being competitive in the marketplace. Those making the hiring decisions are likely to bypass 100,000 potential candidates for ScrumMaster (in their area) and focus on the 10,000 who have the CSM certification. Further, those who understand the certification hierarchy may focus on the 100 candidates (in their area) who have the Certified Scrum Professional (CSP) certification.

If nothing else, certifications demonstrate a continued interest in growth and expanding one's own knowledge and skills. I have no hard data to support the following claim, but it seems to me that those who are constantly pursuing additional

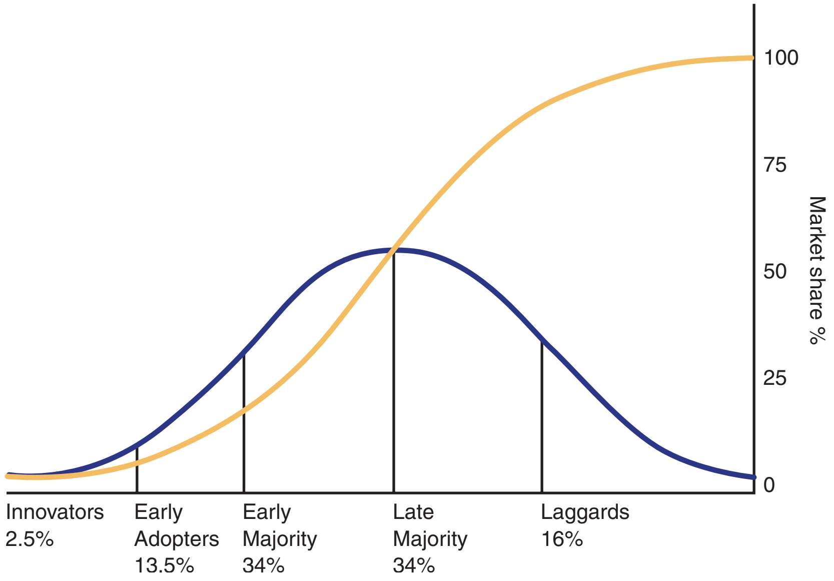


FIGURE 1-6 Rogers' diffusion of innovations curve
Source: Based on Rogers, E. (1962). *Diffusion of Innovations*. Free Press, London, NY, USA.

learning, certification, etc., are also those who are continually reading new books, blogs, etc. They are hungry, lifelong learners.

In terms of marketability overall, I think we are still a long way off from any of the Agile certifications being in the Late Majority phase. Various reports over the years show clearly that the enrollment numbers for Scrum certification courses continue to rise exponentially.

My personal recommendation to anyone wishing to gain a competitive edge in the marketplace is to evaluate where you are in relationship to the certifications that are available to you (and have been available to you) and get caught up on those . . . but don't stop there. If you are eligible for the CSP but don't have the CSM yet, go earn the CSM and then immediately apply for CSP. (Continue to learn, grow, and expand your mind. Pursue other certificates and programs, etc., that confirm your knowledge.)

But most importantly, seek to be the best at what you do, regardless of what it is. A certification, at the end of the day, is just a piece of paper. Your interactions with others and the relationships and REAL contributions you make are the REAL you.

Certify THIS . . .

There are folks who believe certifications are meaningless. That's nothing new.

However, lately, there are some who seem to have taken it on as their life's mission to rail against certification and even training in general. In fact, some have taken to mudslinging in their diatribes about training and trainers.

I get their point.

And I tend to agree with some of the reasons why they are against certifications and training. However, I do not agree with their "solution," which is not really a solution at all. It's more of a lack of a solution or disorganized inaction—a complete hands-off "let people just explore" approach with no standards at all.

(I also don't appreciate the ad hominem attacks on and insinuations about trainers and our general character. Not cool.)

If I am a hiring manager, recruiter, or even a member of a development team responsible for filling a spot on my team, I don't have time to talk to every one of the 1,000 to 2,000+ people who are interested in and believe they MIGHT be qualified for the single position we have open. I need some kind of criteria for establishing at least a baseline for knowledge so that I can narrow the list down a bit. I would like to see some kind of preliminary proof that someone has taken an active interest in his or her career/lifelong learning as shown by their accomplishments (e.g., certifications).

That's only ONE purpose that certifications serve: they are kind of like maintaining a learning log, which is something else that I do and encourage others to do as well. It's not the ONLY thing that I find valuable, however, not by a long shot.

If the person I am talking to, who has several certifications, can't articulate the concepts clearly by teaching it back to me with examples and analogies, etc., and they are unable to demonstrate how they have applied the knowledge, how they have grown since achieving the certification, how they see the limitations of the certification, and so on, then I would not be interested in hiring that person.

Let's take a different view along this same perspective . . .

Assume that you are in the camp that believes in traditional medicine. You would not seek medical advice or treatment from someone who is not at least an MD. (And, the law would agree.) That's just a MINIMUM level of certification that you require or take for granted. You also look for advanced certifications such as orthopedic, OB/GYN, cardiology, gastroenterology, pediatrics, etc.

AND, even beyond all this, you look for doctors who fit your style, culture, and have a great personality and demeanor (bedside manner)—doctors who you can "work with" in addressing your health concerns.

Imagine you were "hiring" a doctor who specializes in oncology to treat you and thousands doctors applied. Would you simply start talking to each one in sequential order without at least looking to make sure that the applicants were board certified in oncology?

Maybe you would. Maybe you would get lucky and only those who know oncology would apply.

However, maybe some doctors who THINK they know oncology would apply. Would you want that person treating you?

I majored in pre-law in college. I have a deep interest in many matters of law. I follow court cases, read opinions, review laws very carefully, etc. (I watch *Law and Order* . . .) I am REALLY passionate about law. Would you like for me to represent you in court? I promise, I will do my best.

They say: "A man who represents himself has a fool for an attorney." You would be extremely foolish to hire me as your attorney because I never went to law school. I am not certified by a bar association the American Bar Association (ABA), or in any state. I might even be BETTER than some of the people who are certified by the bar. But, still, I am not, and that's the price of admission to the "party."

When hiring for a ScrumMaster position in your organization, you can expect thousands of applicants to respond who think they can do the job. Are you prepared to simply start interviewing ALL of them, or would you maybe want to look and see who is a CSM first? Those folks have at least been through a two-day class and passed a test on Scrum. Maybe that represents 500 of the 1,000.

What about looking at those who are CSPs before you dive in with the CSMs even? That might be 50 out of 100 people. You know that those folks have not only taken a two-day training and passed a test, but also have documented considerable experience in Scrum and have been keeping up with their community involvement and learning by obtaining Scrum Education Units (SEUs) with the Scrum Alliance.

I might further look to those CSPs who are in the process of pursuing advanced degrees in organizational development, change management, psychology, sociology, an MBA, or some related discipline that they can use to make themselves more effective. That might be only 10 out of 100 people.

If I get through all 500 of the people who have at least a CSM certification, then I guess I would seek out the folks who have nothing . . .

Certifications are not the be-all, end-all in evaluating a person's skillsets, worth, or potential. However, they CAN be a reasonable place to begin a conversation with someone.

Another point I would like to make here, because it seems like the right time and place . . . If you are considering buying training for your organization but you don't care about certification and think that you can use that as a bargaining chip to get a huge discount: think again. The difference between certified and noncertified training is $50 per person (for the CSM and the Certified Scrum Product Owner or CSPO). That's how much the registration fee is for each student. The effort is exactly the same whether I am certifying people or not. (Also, if your employer is buying training but not letting you get certified . . . um, well . . . WHY NOT???)

Now, I know some folks are sitting there reading this thinking, "Of course you are saying this, Daniel, because you specialize in training and certification. It's very obviously self-serving."

Sure, you could interpret my support of certifications that way.

Or, you might consider that I became a Scrum trainer BECAUSE I believe in certifications as a baseline to begin with . . . and you would be correct.

I want to ensure that when someone walks away from MY class that they are passionate and on fire about learning and growing and that they understand Scrum as best they possibly can from a two-day workshop. I also provide other value-added services to my students well after they take my courses and get their virtual piece of paper (i.e., the certification). I like to help inspire them to keep learning and provide answers to the questions they may have.

Oh, just a bit of history and trivia for you . . .

My interest in certifications goes way back to when I was working at Project Management Institute and led the charge with developing the Project Management Institute–Agile Certified Professional (PMI-ACP) certification, along with several other passionate thought-leaders of the industry; that is, back when I had no other vested stake in having people become certified at all.

Getting the Most Value from Gatherings, Conferences, and Other Events

I tend to go to a lot of industry events for IT and, specifically, Agile software development. In fact, I have now been chair, keynote speaker, reviewer, and volunteer for many of these and have reprised these roles numerous times.

I have a deep insider secret that I want to share with everyone who attends events around the world. This is HUGE so keep it to yourself and really process it.

Ready?

The event(s) that you attended/are planning to attend are NOT about YOU.

[Take a deep breath]

<rant>

That's right. As difficult as it may be to accept, these events are NOT custom tailored for all of your personal needs. The events are designed to meet the basic needs of hundreds and even thousands of people. Even at the workshop or classroom level, it's not solely about YOU.

Picture in your mind the last time you planned a dinner with your close family and/or friends. Was it very easy to pick the restaurant? The dates/times? Did everyone enjoy themselves? Were there issues, complaints, etc.? Did you achieve your purpose?

Now try planning an entire day's worth of meals for that same group of family and friends . . . and activities to keep them happy for the day. Now, multiply that by 3 to 5 days. Now, multiply that by 600 to 2,400 people . . . from 36 to 100 different countries and cultures around the world . . . at a venue in a country that is outside your own.

Are you starting to get an idea of the complexity involved with just the logistics portion of planning these events? Many people I talk to won't even try to plan events on a smaller scale because it is too challenging, let alone a large-scale event.

On the other hand, I hear from many people how they would love to do "event planning," how cool it is or how "neat" it would be. I just smile and listen to them describe the Shangri-La of arranging the event . . . from the perspective of someone who likes to plan and go on *vacations.* Most people have no sense of the bigger picture and what it takes to coordinate large-scale events.

When we solicit feedback from the attendees, I am really saddened to read how much focus there is on things like water and coffee supply, available selection for dietary restrictions, frequency of breaks, and other logistics-related items.

My advice is this:

- If you want great coffee all day long, go to Starbucks.
- If you need to eat every 1 to 2 hours and/or have exotic dietary restrictions, bring a snack.
- If you tend to get cold, bring a sweater.
- If you tend to get hot, wear shorts.
- If you don't like a session, move on to another one.
- If you don't like any of the sessions, start submitting YOUR topics.
- If you don't like any of this advice and are still unhappy with the event, stop going . . .

-OR-

Give some constructive ways to change it that revolve around the content and substance of the event. But in doing so, don't dwell on the lower-level needs in Maslow's hierarchy because as a human being, those needs will inherently dictate what you do, where you do it, and when you do it. Your body will figure it out.

Instead, think about how many other people are there at the event. What would be beneficial to the greatest number of people?

Or, maybe there are topics that are extremely and critically important for a significant number of people.

Or, maybe if the topic is so narrow in scope, you need to plan your own event with a very small number of those subject matter experts (SMEs) that represent that niche topic.

Or, maybe if you are looking for very specific personal advice, mentoring, coaching, etc., you (or your organization) needs to hire a coach so that your individual or organizational agenda is met.

When you go to large events, try to take the perspective of the event conveners and staff. They are busting their asses and brains to please as many people as they possibly can within the constraints they have. It's not that they don't care about you and what you want or need. It's more like they have 600 to 2,400 "yous" to deal with.

If something isn't meeting your expectations, chances are it isn't because the staff didn't think about it.

There probably isn't all-day coffee and water because the really nice venue (did you notice how nice it was?) wants to charge about $130,000 for that service. Are you willing to pay an extra $200 on your registration to have coffee and water all day, each day? That's about $67 a day for a 3-day event or about 10 high-end Starbucks drinks per day.

Not everyone wants coffee. Are you willing to pay that much so that the coffee drinkers can get their fix? Personally, I am too focused on learning and interacting to worry about the other things.

Want coffee? Go to Starbucks.

Or your favorite local coffee shop.

Or, just go to the venue restaurant and order a coffee to go. Win-win. You get EXACTLY what you want.

We are serving hot, steaming, rich, full-bodied conversations, talks, and workshops about Agility.

My goal as an event convener is to take you out of your comfort zones a little bit, to whet your appetites and make you hungry from an intellectual perspective.

From a physical perspective, I just want to make sure we are reasonably meeting your minimum biological needs so that you can sustain that brain of yours to absorb, think, feel, express, etc.

I love, absolutely LOVE, when someone comes to me and says something like " . . . there was this one moment when you were talking about the butterflies and you mentioned that you would not be amongst them because you aren't 'pretty' enough. It had a bit of a chilling or closing effect to me, even though I know you were trying to be funny. I thought 'Well, Daniel is a nice looking guy. If he doesn't feel like he is attractive enough to be a butterfly, then I definitely am not eligible.'"

THAT was useful to me.

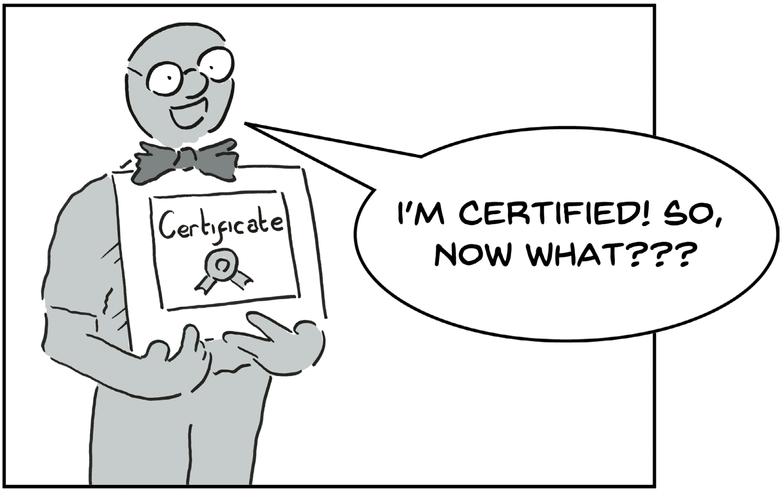
When someone complains about the venue, I get really bored. If there is a fire, pull the fire alarm and dial 911 (or similar emergency number). Otherwise, maybe just try to get engaged to the point where you don't notice those things.

We are not planning these things to be little vacations for you.

I am hoping they will be deep and meaningful experiences for you that will help you to grow and will last a lifetime.

</rant>

I'm Certified—So, NOW What?



First of all, CONGRATULATIONS!!

You have taken the time to earn a certification, a certification that can help to distinguish you from the thousands of other folks out there who have similar qualifications but no certification. However, you may be wondering, "What's next?"

Great question!

Your certification, although extremely valuable, doesn't completely define who you are. In fact, I would say that earning ANY certification is only the start of your journey or perhaps one stepping stone along the path of lifelong personal development. I share this not only as a friendly bit of advice, but also by way of a testimonial.

I have had various different certifications over the years, which have been useful in demonstrating baseline knowledge and understanding in terms of Agile/Scrum concepts and project management. However, true engagement, enrichment, and satisfaction have come via the other activities that I have undertaken, which go well beyond just certification.

My goal here is to suggest some ways that you can enhance your skills, qualifications, and overall value in the marketplace by engaging in ongoing learning and involvement in the community. This shouldn't be seen as a checklist to be completed. It's merely a collection of ideas for HOW to improve in case you might be stuck or overwhelmed by all the choices or you aren't sure where to begin.

First, if you are interested in going beyond the CSM, CSPO, or CSD, you may want to consider the Certified Scrum Professional (CSP) certification.

One perspective on the CSP is that it is the next logical step up the certification ladder at Scrum Alliance, and thus, one more way to differentiate yourself from others in the job market. Even if you are not interested in pursuing CSP, the guidelines entitled Earn SEUs for Your CSP are located on the Scrum Alliance Web site and are fairly comprehensive in terms of ways that people can get involved and improve themselves. In particular, the various categories listed detail many ideas for how to get involved with the community by giving back and how to continue on with your learning journey.

Giving back to the community and helping others is not only a great way to network and build a support system of professionals and even friends, but also results in a heightened sense of gratification and peace. Doing good makes us feel good. And that can take on many forms from simply answering someone's question with your own perspective to helping them as a mentor.

Another perspective on the CSP is this: if you are doing great things for the community and for yourself and you have a bunch of experience with Scrum, why not get "credit" for it by applying for the CSP? This has been my approach more or less throughout my career. I am going to read books to improve myself and my approach to coaching and helping organizations. I also have a servant's heart in terms of helping people to grow and become what they want to be. So, along the way, I figured "Why not apply for Certified Enterprise Coach (CEC)? I am doing all that stuff already."

However, the main emphasis here is this: be hungry and thirsty for knowledge.

Certifications can sometimes help provide a plan for learning in the face of an insurmountable amount of knowledge and resources. "Where do I start???" Well, pursuing a certification many times unlocks paths for additional learning and provides people with ideas and inspiration for continuing on with more focused study in an area that they are passionate about.

Another idea is to check out the Agile Trainer site. This is a nonprofit sister site sponsored by Apple Brook Consulting that offers free help and information to those seeking to become a Certified Scrum Trainer (CST). However, the resources listed and such can be beneficial to ALL who seek to become more effective with Agile practices. This site will continue to grow and change over the months to come and will eventually be a centralized repository for many other Agile resources. If there is anything you would like to see added to the site, I am happy to take suggestions and implement these moving forward as they make sense.

Another idea is to attend a Scrum Gathering or other similar event where there will be hundreds, if not thousands, of other individuals who share your passion about Agile. I have attended many different events over the years and always find that I am energized, inspired, and have a huge list of new resources to add to my toolkit and things to explore further. I also find that my network has grown even BIGGER!! I have made some very dear friends from going to events. It's a great way to get involved in general.

Finally, one of the most important things you can do for yourself is to become 100 percent self-sufficient and self-reliant. I am a firm believer in seeking help and asking questions of others, especially those who are more experienced and knowledgeable than I. However, nothing is more frustrating to people than when someone asks a question about something that they could easily discover in a few minutes by searching Google, Wikipedia, or any other well-publicized resource online.

I can remember growing up without the Internet, back in the 1970s. When I started school back in 1975, my parents invested in two different sets of encyclopedias, and we frequently spent time at the public library. When I asked questions about things that were general knowledge, my parents would gently remind me, "Well, what does the encyclopedia say?" Over time, I learned to first look things up when I had a question. If I couldn't find what I was looking for, I would go to my parents and say "I was looking for X in the encyclopedias but didn't find it. Can you help?" or "I looked up X in the encyclopedia and it says Y. But, I don't understand. Can you explain it to me?"

There are two key things to take away from all this:

- As the saying goes, "Sometimes, to go faster, you have to slow down." Slow down. Read things and learn for yourself.
- If you want help, give help. Chances are, you have already received help, if you pause and reflect on how you have reached where you are in your career. Heck, if you are reading this, then, obviously you can read. Someone helped you with that. Also, you have access to a computer and an e-mail account. Someone probably helped you along the line with that also.

If you need help beyond all this, please reach out and let me know. I am happy to help others because I have received a great deal of help over the years. Lord knows I have needed it!! :)

Goodbye, My Friend

After 3 years of running the Scrum Punkin' Chunkin' Simulation exercise in my CSM and CSPO courses, the time has come to move on.

I first conceived of this idea back when I was going through my CST application process. I had done smaller simulation exercises in my classes, being a big proponent of Sharon Bowman's *Training from the Back of the Room* approach. The idea of bringing something unique, fun, and yet educational to Scrum courses was very appealing to me, and I began to think about what would distinguish this classroom experience from others.

As a resident of Delaware, I find that many people are unfamiliar with our culture. Other than being the FIRST state and the second smallest, the place where everyone incorporates, and the home of Joe Biden, what is it that really puts us on the map??

And then it hit me:

The World Famous Punkin' Chunkin'

If you have not heard of this fantastic event, then you have truly been missing out on a concentrated dose of Americana.

Delaware ("De La Warr") is roughly bisected by the Delaware-Chesapeake canal. Above the canal is Wilmington, which is home to many financial institutions and pharmaceutical and chemical companies. Newark (pronounced "New-Ark," not "Newrk," like in Jersey) is also above the canal and is home to the University of Delaware. Below the canal are massive tracts of farming land before arriving at the renowned Delaware Beaches.

Back in the mid-80s, the farmers would celebrate the harvest by having a pumpkin-throwing competition. As the years passed and one-upmanship prevailed, the competition began to include increasingly more powerful and elaborate machines to throw the pumpkins.

In recent years, the competition has grown to be a three-day weekend event, which is sponsored and featured by the Discovery Channel's Myth Busters syndicate. The air cannon class of machines can shoot a standard 10" pumpkin almost a mile . . . yes, that's right, almost a mile. Those machines can also cost almost $500K and are a huge marketing opportunity for various firms around the world.

I have also been a huge fan of constructive toys over the years having grown up on a diet of Erector Set, LEGO, Tinker Toys, etc.

Then it came to me: "How about a mini Punkin' Chunkin' in the class with different teams building scaled-down models with LEGO?" I set about trying to build a machine with the massive collections of LEGO we have using various rubber bands and some stress pumpkins I had ordered online. The result was like a nuclear explosion of LEGOs all over our living room.

Back to the drawing board . . .

Tinker Toys were too expensive, fragile, and heavy. Erector Set: WAY too heavy. "What else is out there??" I thought. And I went searching.

Enter K'NEX.

I had never played with these because they were a little after my time. They have various different mechanical structures and longer pieces with a bit stronger interlocks than LEGO. They still have interlocking blocks, like LEGO, but pieces that are more suitable to the task at hand.

These totally ROCKED!!

I bought a bunch of K'NEX, rubber bands, more stress pumpkins, and some banker bags to assemble kits for the class. A Scrum Team would compete with other Scrum Teams and would include three to nine Dev Team members, a ScrumMaster, and a Product Owner, depending on the size of the class. With five kits, I have been able to conduct the exercise with classes of up to 55 people.

The whole point and goal was to practice Scrum by building a machine in three condensed sprints of 1 hour each; reflecting on the progress, lessons learned, changing requirements, etc.; and culminating in a RELEASE (literally) by way of competition to see who could shoot the farthest.

The results have been phenomenal.

Initially, I didn't know what to expect. A team shot 8 feet, and I was really impressed. Even more importantly, I observed that the exercise surfaced team dynamics, impediments, and dysfunction that perfectly emulated product development efforts in the organizations I have coached.

Some folks jump right in with a positive attitude and a mind toward the Art of the Possible: "What CAN we do here???" Others focus their efforts on blaming: lack of "technical" knowledge, not enough parts, not enough time, and so on. ScrumMasters have been command and control in their efforts, and Product Owners have been completely disengaged.

Subsequent classes have set records for shooting the pumpkin, all using the same kits, which were randomly shuffled periodically. A team shot 15 feet. Then another team hit 28+ feet; their launch hit a window that was 28 feet away, about 3 to 4 feet up the window. For this facility, there was no larger space to see exactly how far they were launching overall. So, we called it 28 feet.

Finally, a team shot 34 feet using a very simple design at a CSM course I did for VersionOne in Alpharetta, Georgia. The team members were average people, not mechanical engineers. Not engineers at all, really. I think there were some sales folks on the team. But they had a really great team dynamic and working relationship with each other. They didn't let themselves be daunted by conflict or egos. They followed Scrum practices and used the learning and feedback loops to improve. I was so happy and proud of the team.

I also noticed something else:

The other team felt pretty crappy in the wake of the team that shot 34 feet.

In fact, what started off as a friendly, light-hearted competition aimed at teaching Scrum has evolved into a bitter, intensive rivalry with a "win at any cost" theme. I have noticed teams ignoring my coaching and training throughout the exercise. They don't bother with what they have learned the previous 1.5 days, etc. I suppose that this in itself could be parlayed into a teaching moment . . .

However, I have lost my passion for the exercise itself.

The kits have seemed to become VERY heavy. I feel like I am more impatient with the excuses I have heard over and over about not enough parts, not enough time, and so on. Most importantly, I don't like that people feel bad that they didn't produce a machine that shot anything at all, let alone firing 8, 15, 28, or even 34 feet.

I began to brainstorm again on what might be MORE meaningful on multiple levels for the classes. What would involve a low barrier of entry in terms of technology? What is something that EVERYONE around the world could draw upon in their experience? What would be fun but is not perhaps being done by EVERY other trainer out there?

How about a Scrum game?

Yes. That's it. We will use Scrum to build a game that teaches people Scrum. And the students will be responsible for ensuring that the minimum viable product (MVP) for the entire exercise teaches all elements of Scrum at some level. The Shippable Product Increment (SPI) for each Sprint would be some form of playable game that evolves iteratively and incrementally until they have the MVP.

I have been running this now for the last 5 months in the CSPO class as a pilot, and the results have been very favorable.

ALL teams can produce some kind of game. There isn't the pressure of a competition that shifts focus away from Scrum. In fact, the focus most definitely stays on Scrum because the game itself must teach Scrum (Bowman, 2008). What a fantastic concrete practice!!

Furthermore, from a logistical standpoint, the supplies are much lighter and can be much more flexible, depending on variations in availability around the globe. That is, in some countries, construction paper is not available. Neither are the traditional Post-it notes, voting dots, etc., that I am familiar with. However, we can still definitely run the game.

I bring multisided, multicolored dice; little plastic gaming pieces in the shape of pirates, skeletons, army men, orcs, elves, Star Wars characters, etc.; rulers; scissors; tape; and glue sticks. All of these things weigh less than one kit for the Punkin' Chunkin' simulation.

I also bring construction paper and my standard array of Post-its, drafting dots, voting dots, etc., which are consumed during the class so I don't have to transport any of that back with me. Thus, this is a much more adaptable simulation for the class and can accommodate even larger classes. (I have used this successfully with classes as large as 75 people.)

And so it's time to say goodbye to a good friend who has served me well over the last 3 years. I am left with many fond memories of the Punkin' Chunkin' Scrum Simulation exercise. Perhaps someday, I will have a reunion or a larger opportunity for some team, somewhere, to break the 34-foot record.

Thank you to all my students who made this an awesome experience for me. I thoroughly enjoyed learning from you all!!

Closing

In this chapter, we have considered various concerns associated with adopting an Agile mind-set and what it means to value agility in general. We have defined what it means to BE Agile versus simply DOING Agile.

The foundational seeds have been planted so that learning can continue and flourish by pursuing additional resources and knowledge. The reader has been encouraged to get involved with a call to action for promoting community.

In the next chapter, we discuss factors that organizations face as they begin to pursue Agility and how they can position themselves for success by having healthy expectations and a healthy mind-set.

INDEX