

# BECOMING AGILE

in an imperfect world



GREG SMITH  
AHMED SIDKY  
FOREWORD BY  
MARY POPPENDIECK



***Becoming Agile***  
*... in an imperfect world*

by Greg Smith  
and Ahmed Sidky

Chapter 7

Copyright 2009 Manning Publications

# *brief contents*

---

## **PART 1 AGILE FUNDAMENTALS AND A SUPPORTING CASE STUDY ..... 1**

- 1 ■ Moving to agile 3
- 2 ■ The story of Acme Media 17

## **PART 2 GETTING STARTED ..... 25**

- 3 ■ Are you ready for agile? 27
- 4 ■ The fitness test: all about readiness assessments 43
- 5 ■ The importance of obtaining executive support 58
- 6 ■ Improving buy-in by creating a core team 66
- 7 ■ The mindset of an agile leader 73
- 8 ■ Injecting agility into your current process 87
- 9 ■ Selecting a pilot project 105

## **PART 3 KICKING OFF..... 113**

- 10 ■ Feasibility: is this project viable? 115
- 11 ■ Aligning the pilot team with the project 136

|               |   |            |
|---------------|---|------------|
| <b>PART 4</b> | <b>POPULATING THE PRODUCT BACKLOG.....</b>            | <b>151</b> |
| 12            | ■ Feature cards: a tool for “just enough” planning    | 153        |
| 13            | ■ Prioritizing the backlog                            | 170        |
| 14            | ■ Estimating at the right level with the right people | 183        |
| <b>PART 5</b> | <b>ENOUGH INFORMATION FOR SCHEDULING.....</b>         | <b>193</b> |
| 15            | ■ Release planning: envisioning the overall schedule  | 195        |
| 16            | ■ Iteration planning: the nitty-gritty details        | 204        |
| <b>PART 6</b> | <b>BUILDING THE PRODUCT .....</b>                     | <b>221</b> |
| 17            | ■ Start your engines: iteration 0                     | 223        |
| 18            | ■ Delivering working software                         | 230        |
| 19            | ■ Testing: did you do it right?                       | 244        |
| <b>PART 7</b> | <b>EMBRACING CHANGE.....</b>                          | <b>253</b> |
| 20            | ■ Adapting: reacting positively to change             | 255        |
| 21            | ■ Delivery: bringing it all together                  | 277        |
| 22            | ■ The retrospective: working together to improve      | 297        |
| <b>PART 8</b> | <b>MOVING FORWARD.....</b>                            | <b>311</b> |
| 23            | ■ Extending the new process across your company       | 313        |

# *The mindset of an agile leader*

---



A few months ago, Greg was contacted by a friend with a problem. The friend had let a compliance project slip through the cracks. The compliance deadline was year-end, which was a mere five weeks away. Failure to comply could mean serious governmental repercussions. Greg's friend asked for help in creating an agile team and doing an agile project in the following 5 weeks.

This would be a great time for us to tout how agile came in and saved the day, but that would be a lie. Greg did help his friend prioritize his work and make the deadline, and they did follow some agile principles along the way, but they didn't

put an agile team or process in place. Why? Because it takes time to establish an agile methodology. Teams need time to feel comfortable with agile processes, and they need time to learn how to interact with each other. Managers need time to learn how to lead in an agile environment. The team needs to use an agile process for several months, and *then* major benefits will begin to become apparent.

Migrating to agile means more than changing your process. It also requires a change in culture. For most companies, changing culture is the most difficult part. We believe this is true for several reasons:

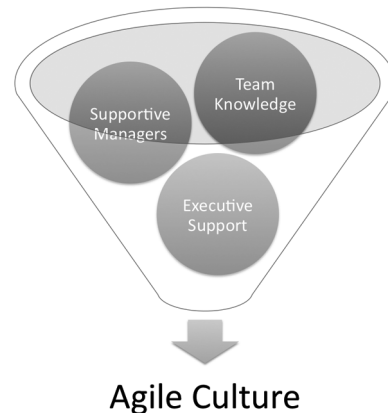
- Regardless of whether it's successful, companies get comfortable with their existing development process.
- Many people still believe that requirements change because they're poorly managed. They can't comprehend a process that embraces changing requirements.
- Most managers have been trained to control events. Empowering the development team to deliver and co-own the project isn't intuitive for managers.
- In larger companies, whole groups are dedicated to regulating and overseeing projects. An agile team has less need for these services, so some employees may feel that their jobs are threatened.

There are numerous other reasons, but we believe these are at the center of the issue.

You should address these issues in two ways. First, you must address the culture needs of each group head-on. We'll show you how to do that in this book by laying out a game plan for obtaining support from line management, the team, the individual, and executive management.

Second, you must establish practices that foster an agile culture. Practices such as high customer involvement, testing early, and collaborative decision making promote an agile mentality throughout the company. You'll see these practices as we follow the case study through the pilot project.

The information in this chapter establishes the foundation that allows an agile process to thrive (see figure 7.1). Similar to software development, if you get a good foundation in place, everything else is easier to do. If you don't, you'll fight the foundation with every change you make. Let's start by looking at the skills required for a good agile coach.



**Figure 7.1** An agile culture is established when three major groups come together within a company. Executive management endorses the agile principles, working managers learn to coach instead of direct, and the project team understands and supports agile principles and practices.

## 7.1 The role of an agile coach

The goal of this book is to convert you and your company into a self-reliant team that can design and maintain your own development process. But out of the gate, you'll have limited knowledge of agile, and you'll be looking for good leadership and guidance from an agile coach.

You use a coach to help you understand where you can become more agile, how to address your constraints, and to train your team on agile principles and practices. Let's start by discussing how to find an agile coach that meets your unique needs.

### 7.1.1 Attributes of a good coach

We've worked with several good coaches. Here are some of the areas to consider when you're looking for a coach:

- *Find a coach with proven results and references*—A good coach has experience in several industries with several flavors of agile. You should be able to speak to their references and validate that value was provided.
- *Don't hire a company, hire a person*—People often ask us for advice in selecting agile consulting companies. In our experience, the quality and experience of individuals within a company can vary greatly. When you need help, you can look at companies; but identify and assess the specific individual who will potentially be working with you.
- *Avoid a cookie-cutter approach*—Some individuals may tell you that one flavor of agile works for everyone. Some companies only support migration to Scrum. In our opinion, Scrum may work for you, but a consultant should evaluate your circumstances before committing to an approach. A good coach will help you evaluate your circumstances before committing to an approach.
- *What about certification?*—A certificate does not validate a coach's skills, but it shows their dedication to their occupation. For the most part, certification is lacking in the agile community today, but ScrumMaster certification and other credentials are available. Recently, the University of Washington opened an agile certificate program.
- *Is chemistry important?*—When you interview a potential coach, you'll get a feel for whether you click with them and whether their approach to migration makes sense to you. Regardless of credentials, you should be wary of selecting someone you aren't comfortable with.
- *Soft skills are critical for a coach*—You should select a coach who can motivate and inspire all the personality types on your team. They should have good interaction skills. Soft skills are discussed in detail in section 7.2.1.
- *Find someone you already know and trust*—As we've mentioned in previous chapters, Acme Media was lucky enough to have a connection to a good agile coach before the company began its search. If you don't know anyone personally, you may have friends at other companies who have moved to agile, and they may be able to provide recommendations.

Johanna Rothman, a well respected agile coach and consultant, recommends that a coaching engagement should always have a deadline so the team doesn't become reliant on the coach. This is a great recommendation, and we agree with it. We often see the coach and core-team leader working together initially; the coach becomes less involved as the team matures.

Now, let's discuss how an agile coach will help you.

### **7.1.2 Training and mentoring the core team**

Your company will need some level of training to begin your move to agile. You may have enough experience in-house to lead your migration, but it's helpful to obtain a third party to coach your team and provide an outside perspective. By using a third party, you also demonstrate that management is neutral and open to new ideas.

Training should happen within a few days of the kickoff with the core team. Determining the level of training is tricky. You want to provide enough information so the team understands the agile principles and their value. But you don't want to train to the point that you hand them a methodology—especially somebody else's. The team should combine agile principles with their knowledge of your business to create a methodology that is effective for your company.

You and your coach must use your own judgment to decide how deeply to train your core team. The assessment in chapter 4 will help here, allowing your coach to understand the existing level of agile knowledge, practices, and culture within your company. Here is our suggested outline for training:

- 1 Begin training with the information in chapters 1 and 3. Explain to the team where agile came from, what makes it work, places where it's working, and why it hasn't faded away. This training should be focused around the agile principles and understanding how agile improves the process. This training should take 1 to 3 days. (Note that it would not be a bad idea for your team to review chapters 7 through 22 in this book. It will allow them to envision how to create and test a more agile lifecycle. It will also show them how agile practices tie to principles.)
- 2 Give the team a few days to absorb the principles, and then train them on the phases of agile detailed in appendix B. We've chosen phase names that map well to names used in traditional software development, which helps with the training process. This will help the team begin to connect principles to practices.
- 3 Use the case study in this book, along with your coach's knowledge, to quantify what agile looks like in practice. The example in chapters 9 through 22 shows Acme Media implementing its own online auction application and the inner details of the methodology the company creates. (In addition, appendix B provides a walkthrough of a project going through a basic agile lifecycle.)

After training is complete, the team will work with your coach and begin the design process by documenting the existing processes.



**Do you really need training and coaching?**

Many people believe they can do their own training on agile, using books or in-house knowledge. We've worked in environments where the team had good agile knowledge, but in most instances we recommend going with some level of agile coaching.

When Greg worked for the *Seattle Times*, his team started a migration to agile with two team members who had worked on an agile migration at another company. These team members provided mentoring and guidance to other team members, but they still decided to bring Jim Highsmith in for basic agile training and to provide coaching as they created their custom process. This worked well because the team respected Jim, and the team tapped into his breadth of agile knowledge when considering practices and techniques.

A coach can be expensive, and you'll want to use your coaching hours effectively. In our experience, we've provided coaching during initial training and then returned one or two weeks later to answer team questions and do more detailed training related to agile practices. We've also provided a lot of coaching via telephone and email, which holds down travel-related charges.

Next, you need to prepare your line management team. Let's look at how things change for managers in an agile environment.

## **7.2 Agile management: more shepherding, less directing**

Do you remember a commercial for a company named BASF a few years ago? Their slogan was, "We don't make a lot of the products you buy. We make a lot of the products you buy better." This is true of the agile manager.

An agile manager never writes a line of code, never documents any requirements, and never tests a feature. Instead, an agile manager does the following:

- Helps the development team track true status
- Encourages the automation of redundant, repeatable tests
- Mentors the team on agile processes and demonstrates their value
- Helps the team break their work into small chunks that can be delivered quickly
- Ensures that the work being delivered is in tune with the customer's needs
- Acts as a buffer for outside interruptions and limits team distractions

Jim Highsmith offers a good explanation of *light-touch leadership* in an agile environment:

*While Light-Touch Leadership may be "light" in terms of decision making, it's heavy in articulating goals, facilitating interactions, improving team dynamics, supporting collaboration, and encouraging experimentation and innovation. These characteristics of a leader are more critical to success than delegation of decision-making authority, but decision making is still an important piece of the leader's role. When a good Light-Touch Leader is working, she or he is nearly invisible. Things seem to happen smoothly and the teams operate seemingly without a leader.*

An agile manager provides leadership without using formal power. Instead, the manager leverages the respect they earn from the team as they establish a history of working together to successfully deliver projects.

What does a manager need to do to establish a record of successful project delivery? Let's start with the soft skills.

### 7.2.1 **Soft skills**

If you look up *soft skills* on the United States Air Force website, you'll find, "A set of skills that influence how we interact with each other. It includes such abilities as effective communication, creativity, analytical thinking, diplomacy, flexibility, change-readiness, and problem solving, leadership, team building, and listening skills."

This definition is an excellent prescription for the behaviors an agile manager needs to subscribe to:

- Effective communication, to ensure that the team is synchronized on information
- Analytical thinking, to help the team brainstorm solutions when they encounter a challenge
- Diplomacy skills, to ensure tactful communications that don't offend or touch on sensitivities
- Great listening skills, to not only ensure accurate understanding but also enhance relationships with others

In summary, the manager should behave in a way that enhances human relations (see figure 7.2).

Diane Ehrlich, PhD, of the Human Resource Development program at the University of Illinois, defines soft skills as "[t]he skills needed to perform jobs where job requirements are defined in terms of expected outcomes, but the process(es) to achieve the outcomes may vary widely." This is a good description for agile development in general. You have a desired output (a project), and the way to achieve that output may vary wildly depending on the specific needs of the project.

Now, let's discuss how soft skills are used.

### 7.2.2 **Working with other managers**

Let's look at team management from the perspective of the person who spends the most time with the team: a project manager or ScrumMaster. These individuals usually lead a group of people who are not their direct reports. In order to do this, the project



**Figure 7.2** An agile leader brings their soft skills together to shepherd the team versus directing them.

manager or ScrumMaster must have the respect of the line managers who own the functional teams. The key is to ensure that the line managers buy into agile concepts before asking the project team to.

The line managers need some level of training before you pursue an agile migration. This training can come from any resource, internal or external; but during this training, managers need to normalize on their support of the principles. You don't want to ask the manager's team to buy into the process before the managers do.

You must also consider roles when working with other managers. Although everyone is flexible in the tasks they perform in an agile environment, everyone will have areas of responsibility.

Consider the development team. The development manager usually acts as a technical mentor and also assigns tasks to the development team. Historically, the development manager may have been in charge of reporting status for the team, too. This changes in an agile environment. Agile teams perform a 10-minute daily stand-up meeting that allows the entire team to discuss what they did, what they will do, and any roadblocks they have encountered. Team members speak for themselves, and status isn't passed to a go-between manager. Traditional managers will need to learn how to provide value and interact in this open atmosphere.

### **7.2.3 Working with stakeholders**

Stakeholders are also vital to your project success. *Stakeholders* are those who have interest in or influence on the project. Typical stakeholders include senior management along with indirect customers such as support teams, maintenance teams, help desks, third parties that integrate with the system, and other related product groups within the company.

All the soft skills mentioned earlier are useful when you're working with stakeholders. The stakeholders may not be the project's main customers, but you want them to feel valued. You should demonstrate good listening skills and make sure they know you understand their needs. You also need to demonstrate diplomacy and not upset the stakeholders by consciously providing information in a way that will inflame or incite them.

### **7.2.4 Demonstrating value**

The most important role of the agile manager is to exemplify the agile principles and live them daily. If you want the team to follow you, you must provide a strong example. There are numerous principles to emulate and follow. Here are the ones that provide the most impact.

#### **"JUST ENOUGH" PLANNING**

In traditional project management, you identify features and then specify their requirements. Typically, an analyst wants to answer every question possible in the specification so the development process isn't impeded by a missing requirement.

In agile planning, you want to plan “just enough.” Just enough planning to determine which features you want to build. Just enough coding to demonstrate the feature to the customer and verify that you’re on track.

Old planning habits are among the hardest habits to break with a traditional team, and the agile manager needs to champion the just-enough mentality on a daily basis. You can also emulate this behavior by creating project plans the same way: a plan that has just enough information to get to the next level of the project, not a complete work breakdown structure before development has even begun.

#### **ALWAYS READY TO STOP, DROP, AND DELIVER**

Agile development is performed in iterations to enhance urgency and to support early delivery of the most valuable functionality. The project manager needs to infuse this mentality into the project team.

You need to get the team to inject the same urgency into an iteration that they do with a final deployment deadline.

#### **UNRELENTING PURSUIT OF CUSTOMER VALUE**

An agile manager is always thinking about the customer and their needs. *All other measurements of a project are meaningless if the product delivered is of no use to the customer.* Follow these three steps to ensure that you address the customer’s needs:

- 1 *Clearly define the customer(s).* Many projects get underway with an incomplete understanding of who the customer is. Make sure your customers are clearly defined and their specific needs are clear.
- 2 *Develop a relationship with the customer.* Get to know the customer well, and integrate them into the project team. Use your soft skills to collaborate with the customer frequently and make sure they can be easily accessed by the team.
- 3 *Be an advocate for the customer at all times.* When features are being discussed and the customer isn’t present, put your customer hat on and envision what their response would be to the discussion. Share those thoughts with the team.

#### **ENSURE TECHNICAL EXCELLENCE**

The technical skill sets of agile managers vary. A manager can come from a classic Project Management Institute (PMI) background, can be a former developer, or may have worked as a business analyst in the past. Regardless of technical knowledge, all agile managers can push the team to pursue technical processes that embody agile beliefs. Here are a few of the best practices for obtaining technical excellence:

- *Create a process for continuous code integration.* As functionality is completed, developers integrate their work into the existing code base. The key is to integrate as small pieces of functionality are completed, as opposed to waiting for a complete feature. This practice identifies code issues early and minimizes the complexity of tracking them down.
- *Automate testing wherever possible.* Work with the team to automate testing wherever possible. This is usually easiest to do with regression testing. You can also automate daily smoke tests to speed up testing.

- *Perform a daily build/smoke test.* Related to automated testing, a daily build also helps mitigate risk by identifying code issues early. The daily test focuses on ensuring that the application's critical pieces are still functional.
- *Consider scalability.* As an application is being developed, the team should consider future growth. What will happen if the application is extremely popular and usage exceeds expected volumes? The team can consider scalability as they design, keeping scalability in balance with simple design.
- *Follow the principle of simplicity in design.* As we mentioned in chapter 1, you should avoid cowboy coding and deliver to the minimum requirement. You should also create the simplest design that will work.

When you've learned how to lead an agile team, you can begin teaching the team how to take part in ownership of the process.

### 7.2.5 **Leading the team to ownership**

In 1998, Arthur Andersen published a book titled *Best Practices: Building Your Business with Customer Focused Solutions*. One of the best practices outlined in the book is the ABO Continuum. This continuum identifies a vital element in introducing change to an organization: ensuring ownership of the change.

The continuum promotes the belief that organizational change goes through the following three steps:

- *Awareness*—In this phase, information about the change is shared early and informally. For example, during a team meeting, a manager can say, "The executives are discussing improvements for our development process." The manager can also indicate when they think they will hear more, and see what the team reaction is.  
  
The value isn't so much in what is said as in *when* it's said. Every individual has their own timeframe for evaluating a change. The earlier you can make a group aware of a potential change, the better your chances of getting them to buy into the change when you're ready to roll it out.
- *Buy-in*—This phase occurs when you roll out the change and begin implementing it. You created awareness earlier, and you're looking for the team to consider the change and to use it with your guidance.
- *Ownership*—The team has tried the change, begun to believe in it, and adopted it as a standard practice. They don't need management to encourage them to use it. They believe in it and will use it without being prodded.

The ABO Continuum is a great approach for rolling out an agile methodology.

Now that we've outlined the characteristics of an agile management group, let's discuss creating an agile mentality within your project teams.

### **The ScrumMaster**

Scrum has become one of the most popular agile packaged methods. The ScrumMaster is at the heart of Scrum. This individual isn't a manager but more of a process facilitator and guide. A ScrumMaster does the following:

- Helps the team develop practices that support agile principles
- Acts as a guide in training the team on how to be agile and use Scrum
- Removes impediments that prevent the team from delivering software
- Shields the team from corporate bureaucracy and activities that don't add value to software development
- Champions engineering excellence and processes that support the creation of shippable software
- Ensures that the team has direct access to the customer

We believe Scrum is a good agile framework, especially when there is urgency to establish a development process quickly. But we worry that some teams can become too dependent on the ScrumMaster.

Our opinion is based on something we learned when we became certified as ScrumMasters. Our instructor told us that ScrumMasters are the key to Scrum's ability to transform the organization. He also told us that ScrumMasters are responsible for team health. We understand that not everyone was taught this principle when they became ScrumMasters, but we are still concerned that many people believe the ScrumMaster is the sole owner of team health.

Over time, we've come to dislike the thought of one person with so much responsibility. In our experience, leads and managers have shared ownership when we migrated to agile. Our teams included definite agile experts and leaders, and we frequently asked those experts for guidance; but we never asked the experts to own the process or team health. We did this collaboratively as a leadership team. We've found this method to be successful because we do get expert opinion, but we don't relinquish ownership of team health or the development process to one person.

## **7.3 Creating a team with an agile mindset**

An agile team comes across as poised and ready for wherever the project may lead them. Agile team members don't fear uncertainty; they look forward to the challenge and know they will succeed.

Where does this air of self-assurance come from? Does this attitude reflect the type of people who were hired? Or does it reflect the processes that are being used? Is the attitude a byproduct of executive support? Does confidence come from a history of successful deliveries?

The answer to all of these questions is *Yes*. Each of these items supports the effectiveness and self-reliance that is inherent in an agile team. In some ways, creating an agile team is like baking a cake. You can obtain the ingredients exactly as the recipe requests, bake at the suggested temperature, and let the cake cool the specified time

before applying the icing. But what happens if you're at high altitude and you forget to make the necessary adjustments? The cake rises too quickly and then turns out too dry. Or what if someone jumps up and down in the kitchen while the cake is baking? The cake collapses and never rises.

In this section, we'll give you the ingredients for creating your agile team. In subsequent chapters, we'll walk you through "high-altitude baking" and how you should adjust your recipe accordingly.

### 7.3.1 Culture and roles

We find it hard to describe agile team culture in a sentence, but we can easily describe it with several words. The words that come to mind are *collaborative, open, passionate, courageous, honest, lighthearted, driven, synchronized, customer focused, funny, responsible, innovative, and successful*.

The culture is one of low politics and high transparency. Words are honest but not abrasive. Status is discussed in matter-of-fact terms. The team focuses on the situation, not the person.

Estimates are honest. There is no padding to make the work easier to do. There is no lying about how long it will take, in order to appease management.

Another nuance of an agile environment is the roles the team members play. Other than as suggested by Scrum, agile doesn't specify what team-member roles should be. In our experience, this hasn't been an issue. The teams we've worked with didn't change their roles after they migrated to agile. We still had developers, testers, project managers, product managers, customers, DBAs, and operations personnel.

What did change for those teams was attitude. After we migrated to agile, we rarely heard a team member saying something like "development isn't responsible for that" or "quality determines when the code is acceptable." We saw many more team decisions and much more collaboration around problem solving. A problem wasn't tied to one role that had to solve it. Instead, it was tied to the project, and *the team had to solve it*. An agile team focuses on the goal, not their job descriptions.

The last item related to culture is diversity. If you don't have a diverse team, your agile process can lead to *groupthink*. Groupthink happens when team members want to get along with each other so desperately that they won't voice their opinion when they disagree with an idea. This is a definite danger with agile. People assume collaboration means harmony and always getting along. They think that if they start agreeing with each other all the time, they're being collaborative. In fact, good collaboration often includes disagreement.

The reciprocal of groupthink is diverse opinion that is spoken freely. This is what you want in your agile environment. A good example of this occurred during the Apollo 13 space mission. In this instance, an explosion occurred aboard the spaceship while it was on its way to the Moon. The ship and crew were saved with a little luck and some spectacular collaboration.

**A classic groupthink example: the space shuttle disaster of January 28, 1986**

The space shuttle Challenger was preparing to launch on a cold day—the weather was colder than it had been for any other space shuttle launch. One of the engineers from a company that supplied parts to the space shuttle warned that there could be risk in launching. He was concerned that the O-ring seals his company provided might fail in the low temperatures because they had never been tested below 53 degrees Fahrenheit. The engineer shared this concern during a teleconference with NASA, and NASA urged him to reconsider his recommendation to not launch. The pressure from NASA persuaded the company to acquiesce to the request and overrule their engineer's warning. Subsequently, the O-rings failed just after launch, leading to the death of the entire Challenger crew (Griffin 1997).

As the Apollo 13 crew experienced various issues in trying to return to earth, the support team on the ground went through days of brainstorming and collaborating to solve the problems. No one team member had more influence than another in suggesting a solution, and “getting along” wasn’t a requirement. When problems were discovered, ideas were discussed passionately until the group reached consensus.

Culture isn’t an optional ingredient in your agile recipe. The majority of the team must embrace the agile culture or you won’t be agile—you’ll just be a team that calls itself agile and goes about business as before.

Let’s take a moment to look at the building block of the team: the individual.

**7.3.2 Characteristics that influence individual performance**

Not everyone on your team needs to be competent and mature, but you should put a system in place that breeds competency and helps the entire team become competent over time. But just as in traditional development, competency alone doesn’t guarantee team success. Several factors affect the productivity of an individual. Let’s review a few of them.

**MOTIVATION AND REWARD STRUCTURE**

A talented, mature individual won’t stick around to work on your agile projects if their efforts aren’t rewarded. A person who is talented can frequently choose where they want to work. It’s up to the company to create an environment that attracts and retains talented individuals.

In simplest terms, behavior reflects incentives. What incentives can you provide to attract talented individuals to your agile team?

Consider the following items related to motivating and rewarding the individual:

- Is the mission of your company clear? Has it been clearly communicated to each individual? Employees want to know where the company is going and how their projects tie to the vision.
- How is health of your company? Are you doing well financially? Are you a start-up fighting to survive? Company health can tie to motivation in two ways. First,



if you're healthy and growing, you can convey this message to employees and tell them that you offer stability, growth opportunity, raises, and potentially equity. If you're struggling to survive, the message is the importance of the project and how it affects the destiny of the company. *Everyone wants to work on projects that are important.*

- The agile environment stresses the value of the employee beyond their job title. They make management decisions and are responsible for proactive communication. Talented individuals welcome this environment. Employee evaluations should recognize and evaluate collaboration skills.

Another factor related to employee motivation is career stage.

#### **CAREER STAGE**

As you migrate to agile, you must consider various approaches to moving your employees to an agile mindset. To help you determine the approach to use, consider where each employee is in regard to their career. Here are the main stages and suggested approaches:

- *New employees*—Employees who are in a stage of rapid learning and trying to understand the company and processes around them. They're dependent on others to get things done, and they're working to become independent from support. Such employees enjoy learning agile because it levels the playing field for them. They're at ground zero, just like senior employees, and they're comforted by the fact that everyone is learning agile together. They should also do well using the methodology because they don't have a lot of previous experience to bias them.

You don't have to do anything special with these folks. Just be sure they get the same training as everyone else and that they're offered the same opportunities as other team members.

- *Individual contributors*—The employees who make up the bulk of your teams. They aren't new, and they aren't supervisors or managers. They have a medium to large amount of experience, and they may have chosen not to become managers but instead to become an expert in their functional area.

These folks require the most management, and you must address their needs individually. Some general tips for motivating these employees are as follows:

- Give them an area to own and be responsible for in your migration.
- Give them an opportunity to use and share their expertise.
- Give them a chance to be innovative and unique.

A lot of these employees are looking for growth and embrace agile. Some of them are just getting comfortable with the way things have always been done and resent having to learn another new thing. Be patient with the "resenters" and remember them when the time comes to criticize the agile design: their feedback will be valuable.

- *Coaches*—Employees who are motivated by sharing their experience and mentoring others. They're also looking for an opportunity to renew and revitalize themselves. An agile migration project is just what the doctor ordered for these employees.

Give these employees leadership opportunities during the migration, such as resolving design issues or leading the team to consensus. They can also be on the forefront of receiving agile training and can mentor novice employees on the process.

## **7.4 Key points**

The key points from this chapter are as follows:

- Moving to agile requires a change in practices and culture.
- Moving to agile takes time. For optimum results, you need to allow time for your company to digest the change.
- An agile coach will help you move to a more agile process by mentoring and training your team.
- An agile coach will help you assess your team's ability to increase agility and also help you design a more agile process.
- Managers need to learn how to lead in an environment with empowered teams. Managers will earn their money by knowing when to lead, when to help, and when to let the team run on its own.
- Team members can maintain their existing roles, but a long-term goal for your team is to cross-train and to minimize dependency on specialized skills.
- You must consider the needs of individuals when you move to agile. Address the needs of the new employee, the individual contributor, and the coach.

## **7.5 Looking forward**

In this chapter, we completed the organizational aspects of preparing for migrating to agile. Now we're ready to roll up our sleeves and start doing the work. In chapter 8, we'll join the Acme Media core team and watch as they review their existing process and look for ways to inject agility.

# BECOMING AGILE ...in an imperfect world

Greg Smith • Ahmed Sidky

Foreword by Mary Poppendieck

**P**eople know about the benefits of agility but many have been reluctant to make the transition themselves. This book will help overcome the barriers that hold them back. Whether you're a small shop or part of a large corporation, you can benefit by learning the practices it recommends. They'll make your dev processes faster, more flexible, and more cost effective.

**Becoming Agile** addresses the real issues—including the needs of executives, managers, and the development team during the transition. It starts by helping you gauge how agile you already are. Then, it shows you how to create a process that supports the realities of your environment so you can transition gradually to an agile way of working.

## What's Inside

- Obtain team buy-in for the move to agile
- Assess what practices you can pursue with low risk
- Reach the right level of agility for your organization
- Create an agile process within your specific constraints

**Greg Smith** is a Senior Project Manager, ScrumMaster, and an Agile college instructor. Greg has helped teams become agile at the start-up and enterprise level. **Dr. Ahmed Sidky** is an experienced agile coach, helping agile teams around the world. Called "Dr. Agile" for the creation of his readiness assessment tool (doctoragile.com) and the Sidky Agile Measurement Index.

For online access to the authors, code samples, and a free ebook for owners of this book, go to [www.manning.com/BecomingAgile](http://www.manning.com/BecomingAgile)

"A thoughtful, practical set of steps ... to become agile."

—From the Foreword by  
Mary Poppendieck

"Makes agility rhyme with reality."

—Valentin Crettaz  
Web Architect, Consulthys

"This book is a *necessity* for teams looking to become agile..."

—Randy Miller, Software Project  
Recovery Expert, Microsoft

"The first book I can give to anyone on an agile team, regardless of their role."

—Paul King, Director, ASERT

"A 5-star guide"

—Robert Dempsey, CEO,  
Atlantic Dominion Solutions



MANNING

\$44.99 / Can \$56.99 [INCLUDING eBook]