



# Agility

A white paper from 3Back, LLC



[3Back.com/Agility](http://3Back.com/Agility)

# Agility

## A White Paper from 3Back, LLC

*The essence of Agility is adapting to, and exploiting, the realities we see;  
all Organizations and People should be as Agile as they can be.*

**Prerequisites:** None

### Table of Contents

Agility and Accountability.....	1
Planning, Goals and Strategies.....	2
‘Agility’ is a Military Word.....	3
Mental and Physical Agility .....	3
Team-Based Agility.....	4
The Agile Manifesto.....	5
Summary.....	6

Have you ever been working on something and had something go ‘sideways’ on you? Maybe you’re working in the garden and you hit a big rock or run into to tree root. Maybe you’re cooking a meal and realize you’re missing an ingredient. Maybe you’re assembling an IKEA® bookshelf and realize you used the bottom shelf on the top. When something like this happens, when something goes wrong, when something goes sideways, you usually ask yourself the questions: “*Now what? What do I do now?*”

When this happens, you have just entered Agileville, where things can go wrong and you must expect the unexpected – there are surprises around every corner. In Agileville, your plans don’t always work out, and you will often have to make decisions about what to do next. In Agileville, you **know** you don’t know what you want to know, and you’re ok with that.

We need Agility to deal with the unexpected. We make plans, and those plans often fall apart or need to change. Predicting what will happen is hard, especially if you’re looking for details, and Agility is what we need when our plans are not working out – which is often the case.

## Agility and Accountability

The essence of Agility is making necessary decisions when things ‘go sideways’ and being accountable for those decisions. Agility is a high-discipline activity; your decisions should be informed and well thought out, and you must be able to explain them to others. There are no ‘*I don’t knows*’ in Agileville; if you made a decision, you *must* be able to explain it – this is the essence of accountability.

Often, you are accountable to a person or group, but sometimes you are simply accountable to the universe. In this discussion, we are not using ‘accountable’ in the ‘blameworthiness’ sense; we are using ‘accountable’ in the ‘answerable, account-giving’ sense. Being ‘accountable’ means that you can be ‘held to give an accounting.’ That is, when the right people ask, you must explain yourself to them. You must be able to answer questions like: “*Why did you do that?*” and “*Why did you make that decision?*”

Now, making decisions is risky, and it takes courage, especially in an organization that does not honor and respect decision-making. In many organizations making a ‘wrong’ decision can be dangerous for the decision-maker, so plans are often followed by a disastrous conclusion rather than corrective action being taken. The less blame-setting an organization’s culture is, the easier it is for that organization to be Agile.

Healthy organizations *expect* good decision-making and their culture is not about being ‘passive-aggressive,’ ‘apathetic,’ ‘fake-nice,’ or having a ‘mob-mentality.’ Healthy organizations expect engaged, thinking people to make the *best* decisions they can *based* on the information they have *when* the decision needs to be made. Healthy organizations expect an ‘accounting’ so that information can be exchanged and learning can occur – this is how organizations improve.

Often the hardest part of an organization becoming Agile is changing its culture. Becoming Agile is much more than simply changing the processes, practices, patterns and rules; it is about changing the organization’s collective mindset. This is hard! But even within organizations with cultures that are antithetical to Agility, a sub-organization might be able to become Agile *if* it has a leader who protects it from the rest of the organization *and* fosters an Agile-friendly culture within the sub-organization.

In addition, each subsection of an organization may require Agility that is unique to itself – the problems the subsections run into and the decisions they make are fundamentally different in some way. Therefore, an important part of an Agile organization’s leadership is to recognize the different Agile needs amongst subsections and deal with those differences appropriately.

## Planning, Goals and Strategies

Now, even though we don’t expect our plans to work out in Agileville, that doesn’t mean we don’t do Planning. We do Planning to determine Goals and Objectives, to provide Strategies, to establish baselines, and to set expectations. We like to have ‘some idea’ of what we’re doing, and ‘having a plan’ makes us comfortable and establishes a context within which we will be Agile.

Let me talk about Strategies for a minute. A Strategy is different from a Plan. A Plan tells us what we’re supposed to do, and how we’re supposed to do it – a Plan can be thought of as a series of Steps to follow, like assembly directions or a recipe. A Strategy is different. A Strategy is like a *high-level* Plan, and it consists of a series of Goals rather than a series of Steps. Of course, one person’s Step is another person’s Goal, so the Strategies and Plans often need to be decomposed and become more and more detailed until they get to a point where the Steps/Goals are actionable.

Each Goal in a Strategy needs to be ‘squishy’ enough to allow for Agility within it; the Goal can be achieved in many different ways, and all you have to do is find *one* of them, and it may not be the one your boss had in mind. Each Goal may come with an associated (internal) Plan, but the Plan is not ‘written in stone’ – it is expected that enough Agility will be added to the internal Plan in order to achieve the Goal. It’s all about *figuring it out* as you go.

In other words, Strategies supply guidance to decision-makers while they are being Agile. A Strategy to achieve a Goal is basically a series of sub-Goals and decision-making guidance that (hopefully) allows an organization to achieve the overall Goal while allowing the organization the freedom to be as Agile as it needs to be while doing so.

Of course, the Goals could be unachievable or the Strategy itself could be wrong – there are no guarantees. This simply means that an even higher-level of Agility is needed – the top-level Goal needs to be changed... being Agile is no guarantee of success, it just improves your odds – impossible is still impossible.

## 'Agility' is a Military Word

My first encounter with the word 'Agility' in a management context was in 1983 when I was studying the Army's AirLand Military Doctrine in an Army leadership course. The AirLand Battle Doctrine was about how the Army and Air Force were going to work together to beat the Soviet Union in Europe – how we were going to 'win' WWII – and it was based on four basic tenets: Agility, initiative, depth and synchronization.

Each of these tenets is interesting in its own right, but in this white paper we're focusing on Agility. The military definition of 'Agility' is unsurprising: Agility is *adapting to, and exploiting, the chaos of the battlefield*, and the goal is to do it faster and better than your enemy does.

I spent a lot of time in the Army training on Agility. Some of my fundamental take-aways were:

- Nothing is risk-free. All layers (echelons) of Command must take informed risks, and the organization must support this risk-taking;
- Orders should state the Commander's Intent (what is needed and why), and Commanders should not attempt to control every action of their subordinates; and
- People should have the greatest possible freedom to carry out their Orders; they should have initiative, be flexible and quick-minded, and improvise as necessary.

Caution: Please do not insult people who have been trained in the military's 'Command & Control' concept by saying it is not Agile. The 'Command & Control' concept has been abused by people in the non-military sector who have misunderstood the term. The term has been misconstrued by many people to describe a bad behavior. In the military, 'Command' and 'Control' are two different concepts, and the phrase exists to remind us that they are separate and should not be combined. To make this point, I sometimes do the following exercise in my classes: "We know two things: that 'Command and Control' is bad, and that 'Command' and 'Control' are both necessary. Discuss..." The results of the discussions are usually enlightening and produce an 'aha' moment for many students.

## Mental and Physical Agility

We often speak of Agile processes. For example, we refer to Scrum, eXtreme Programming (XP) and Kanban (among others) as Agile processes. But processes can't be Agile, only people can be Agile – only people can make decisions and be accountable for those decisions. So what do we mean by an Agile process?

Decision-making requires situational awareness; the decision-maker must understand the situation, or the decision is just a guess, a gamble. If an organization's process facilitates situational awareness, we refer to this process as 'Agile' and we say that this organization has physical Agility. Generally speaking, physical Agility means that your process uses frequent exploratory feedback to produce the information you need to have to make decisions – and it gives you this information quickly and frequently.

However, you are not actually Agile unless you make those decisions – you don't wait for those decisions to be made by the universe or somebody else. Mental Agility refers to making decisions when they need to be made and requires flexibility, initiative, quick-mindedness, risk taking, aggressiveness and improvisation.

So, becoming Agile requires two things: making your organization and processes more physically Agile, and making your culture and people more mentally Agile. There are many Agile processes, practices, philosophies and frameworks, and your organization should embrace and adopt some of them. But that's just the start.

The hard part is going to be changing your people and culture to become more mentally Agile. This is not easy and requires learning and practice. People don't wake up one day being able to make good decisions. To quote Mark Twain: "Good decisions come from experience. Experience comes from making bad decisions."<sup>1</sup> It is only through the repetition of making decisions that we gain the experience in decision-making that will help us become mentally Agile.

So, we've got to change our culture to force/allow people to make decisions *all* the time. We must emphasize to them that while there are constraints and rules about decision-making, decisions are made by people doing the best they can, and they are held accountable for those decisions. As a boss, I often tell my folks that it is *their job* to make decisions, *my job* to enable them to make decisions, and *our job* to discuss the decisions that have been made.

The question "Why did you do that?" is not a challenge; it is simply a question. The respectful conversation that follows is how we learn; when I ask this question I am hoping to learn how and why you did something, and I am hoping you will learn what I would have done. Having this conversation makes us both smarter and makes our organization stronger as a result.

## Team-Based Agility

We usually think of Agility in terms of a Team, which is a group of people who have common Goals and work together to achieve Results. There are two types of people on (or near) a Team: workers and deciders. The workers are the people who actually do the work, and the deciders determine what Results should be achieved. Not surprisingly, then, there are two types of Agility on (and around) a Team:

- *Worker Agility*: Workers are the Team Members who have the necessary knowledge and skills to work together to produce the Results. Their intent is to achieve the Results, and they must do whatever it takes to do so. They must show initiative, collaborate amongst themselves, demonstrate *informal* leadership as necessary, and make decisions that get the job done. This is *tactical Agility*, as it's about how, not what. A Team that does this is called *self-organized*, as the Team organizes itself to get the job done. The Team Members often demonstrate *emergent leadership*<sup>2</sup> within this self-organization. Basically, worker Agility is simply about doing whatever it takes, within the given constraints, to produce the Results that have been asked for.
- *Decider Agility*: Deciders are people who decide which Results the Team should produce. They can be on the Team (like a Team Captain) or outside the Team (like a Coach or Manager). Deciders are demonstrating *formal* leadership and own the Team's Plan. Through feedback and observation, they determine what is and isn't working, and use this information (along with appropriate Strategic guidance) to do *informed re-planning*. This re-planning, changing the Team's direction, is the Agility that the deciders own. This Agility is *strategic Agility*, as it is about *what*, not *how*. Basically, decider Agility is about deciding what Results the Team should produce in order to achieve the Team's Goals and Objectives – and doing this in a potentially complex, changing, environment.

So, worker Agility is tactical (about *how*, not what), and decider Agility is strategic (about *what*, not how). It can get interesting. For example, on an American Football Team, the Coach usually calls the Plays – identifying the Results to be achieved. However, the Quarterback has the authority to change the Play (call an audible) if the situation on the field requires it. Normally, the Plays (including the audible) are highly scripted, and the Team just runs the Play – there is little or no tactical Agility. However, sometimes the Play gets 'busted' (everything goes sideways), and the Team goes into scramble-mode and self-organizes to try to make the best of it. In any case, once the Play is over the Coach re-evaluates the situation and calls the next Play – thus exhibiting strategic Agility.

---

<sup>1</sup> Mark Twain. (n.d.). AZQuotes.com. Retrieved February 19, 2016, from AZQuotes.com Web site: <http://www.azquotes.com/quote/610491>

<sup>2</sup> Emergent Leadership: Leaders arise from the group and have their base of power from followers rather than a higher authority.

# The Agile Manifesto

The Agile Manifesto brought Agility to software. In 2001, a group of developers who had each been doing this type of development for years got together in Snowbird, Utah in order to explore what they had in common – what they could learn from each other.

The result was the Agile Manifesto:

## **Manifesto for Agile Software Development**

*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.*

Even though it is called the “Agile Manifesto,” only one of the four preferences (the fourth one) is about Agility. This can (and does) cause some confusion, but the value of the Agile Manifesto makes it worth it. The Agile Manifesto did three great things for the software community:

- It talked about Values, rather than practices, which allowed the conversations about Agility to include both practices and people.
- It gave Preferences, rather than dogmatic guidance, which allowed for a wide range of projects that could call themselves Agile, depending on how they balanced the preferences.
- It firmly established the word ‘Agile’ as the generic term for this wide-ranging family of tools, teams, and Organizations.

The “Agile Manifesto” also causes a couple of problems:

- It buried ‘Agility’ as merely one the four preferences (we prefer Responding to Change over Following a Plan), allowing people to believe that being Agile is merely a choice, and not an imperative<sup>3</sup> and
- It defined the word ‘Agile’ in a way that led to confusion about what it means. The definition moved from “*responding to change*” (its traditional definition) to “*believing in the Agile Manifesto*” (which is only applicable to software development).

All in all, however, the Agile Manifesto is a very good thing.

---

<sup>3</sup>I can think of *no* situations where ‘Following a Plan’ is preferable to ‘Responding to Change,’ as long as we agree that an appropriate response to change includes “*we considered it, and we decided to ignore it for now.*” In other words, we should always be Agile, not simply Following a Plan for its own sake.

## Summary

Agility is nothing new; it has a long history. The study of Agility is a science; the military has been studying it for centuries. Using Agility can help organizations and people succeed in their efforts, but doing Agility well requires mastery, study, and practice.

Some level of Agility is natural. Most of us live our lives in an Agile way, doing reality-based planning and replanning every day. We don't even realize that we're being Agile, we're just 'doin' what we gotta do...'

However, being Agile in a thoughtful, purposeful way can lead to increased success for both us and our organizations.

## Primary References

1. Dan Rawsthorne and Doug Shimp, *Exploring Scrum: the Fundamentals*, 2<sup>nd</sup> Edition, 2013, <http://www.amazon.com/dp/1461160286>
2. FM 101-5, *AirLand Battle Doctrine*, August 1982.
3. FM 6-22, *Army Leadership: Competent, Confident, and Agile*, October 2006.
4. The Agile Manifesto, <http://agilemanifesto.org>

## About the Authors

**Dan Rawsthorne** has developed software in an Agile way since 1983. He has worked in many different domains, from e-commerce to military avionics. He has a Ph.D. in Mathematics (number theory), is a retired Army Officer and a Professional Bowler and Coach. Dan is very active in the Agile/Scrum community and speaks quite often at conferences and seminars. He is a transformation agent, coaching Organizations to become more successful through Agility. His non-software background has helped him immeasurably in his coaching: his formal training in mathematics guides him to look for underlying problems rather than focus on surface symptoms; his military background helps him understand the importance of teamwork and empowerment, and his work with bowlers has helped him understand that coaching is a two-way street.

**Doug Shimp** has worked in the technology field since 1992 and has played many key roles on software teams, including Coder, Tester, Analyst, Team Leader, Manager, Coach and Consultant. Doug's passion is for team learning to improve product development, and he is a leader in the area of Agile/Scrum transitions and applied practices. He believes that the core basis for applied Agility is that 'You must see the result for it to be real; otherwise it is all just theory...' Much of his experience with teamwork and Agility comes from outside the software field, including an earlier career as an owner/manager of a painting company, which enabled him to learn about small-team dynamics in a very hands-on way.