

STOP CHASING ~~AGILE~~

**FROM PROCESS WORSHIP TO
REAL PROJECT PERFORMANCE IN
A CONSTANTLY CHANGING WORLD**

JEREMY BERRIAULT



Copyright © 2026 by Jeremy Berriault

ISBN: 978-1-60427-217-8

e-ISBN: 978-1-60427-872-9

Printed and bound in the U.S.A. Printed on acid-free paper.

10 9 8 7 6 5 4 3 2 1

Library of Congress Cataloging-in-Publication Data can be found in the WAV section of the publisher's website at www.jrosspub.com/wav.

This publication contains information obtained from authentic and highly regarded sources. Reprinted material is used with permission, and sources are indicated. Reasonable effort has been made to publish reliable data and information, but the author and the publisher cannot assume responsibility for the validity of all materials or for the consequences of their use.

All rights reserved. Neither this publication nor any part thereof may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopying, recording or otherwise, without the prior written permission of the publisher.

The copyright owner's consent does not extend to copying for general distribution for promotion, for creating new works, or for resale. Specific permission must be obtained from J. Ross Publishing for such purposes.

Direct all inquiries to J. Ross Publishing, Inc., 151 N. Nob Hill Rd., Suite 476, Plantation, FL 33324. For EU safety/GPSR concerns, email: gpsr@mare-nostrum.co.uk or contact: Mare Nostrum Group B.V, Doelen 72, 4831 GR Breda, The Netherlands.

Phone: (954) 727-9333

Fax: (561) 892-0700

Web: www.jrosspub.com

To my father.

And to my uncles, Brian and Jerry. You helped shape how I think about work, craft, and the value of the right tool at the right time.

I miss you both.

CONTENTS

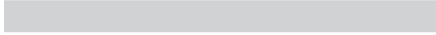
<i>Acknowledgments</i>	<i>ix</i>
<i>Introduction</i>	<i>xi</i>
<i>How to Read This Book (Depending on Your Role)</i>	<i>xxiii</i>
<i>About the Author</i>	<i>xxv</i>
<i>WAV Page</i>	<i>xxvi</i>

Chapter 1	The Agile Graveyard	1
	Agility Isn't New	4
	Agility as an Ecosystem	6
	So, Why Are We in a Graveyard?	6
	The Baggage We've Created	8
	The Coaching Problem (and Comfort Problem)	9
Chapter 2	The Framework Fiasco	15
	The Damage of Process Obsession over Outcomes	17
	When the Framework Becomes a Performance	20
	Scrum by the Book... and Still Failing	22
	Certification Culture: Fuel for the Fire	25
	The Real Question	26
Chapter 3	Influencers, Gurus, and Snake Oil	29
	How Bad Influencers Poison the Well	29
	When Organizations Fall for the Show	31
	Thought Leadership Versus Thought Marketing	34
	The Ethics of Influencers	38

	Leaders: How Not to Get Fooled	39
	If You're a Coach, Read This	41
Chapter 4	Buzzwords and Word Games	45
	A Guided Tour of the Worst Jargon	45
	Why This Matters	53
	How Language Gets Weaponized	54
	How Jargon Confuses and Divides	54
	How to Call It Out and Cut Through It All	55
	Underneath the Words, It's All Behavior	59
Chapter 5	Behavior First: The Heart of Real Change	61
	Behavior	62
	What Behavior-First Agility Actually Means	63
	Change Management Is Behavior Change	64
	How Do You Create a Behavior-First Culture?	67
Chapter 6	Team Well-being Over Everything	73
	Psychological Safety and Trust: Sprints and Story Points	74
	The AI Trap and the Hidden Cost of Efficiency	75
	Burnout, Stress, and Agile Done Wrong	77
	Well-being Is the System	78
	Stretch, Don't Snap: Workload and Calm Leadership	79
	Norms Around Availability: Remote, Hybrid, and Reality	81
	How Leaders Respond to Stress (Everyone's Watching)	82
Chapter 7	Leadership Lessons (the Hard Way)	85
	What Is a Leader?	85
	What Does Authentic Leadership Look Like When Things Go Sideways?	86
	Courage, Humility, and Learning from Failure	88

	Pulling It Together: How to Lead (Without the Hero Cape)	91
Chapter 8	Jiu-Jitsu, Puzzles, and the Art of Adaptation	95
	No Plan Survives First Contact	96
	Pressure Versus Panic	97
	Tapping: Failure as Feedback	98
	Different Partners, Different Games	99
	Puzzles: A Model for Problem-Solving in a VUCA World	99
	Pattern Recognition, Patience, and Partial Progress	100
	Collaborative Puzzles = Cross-Functional Teams	101
	Adaptability and Creative Problem-Solving Beat Rigid Plans	102
	Ego Versus Adaptation	103
	Five Lessons from the Mat and the Puzzle Table	103
Chapter 9	Real-World Wins (and Fails)	105
	Case Study 1: The Software Company That “Paused Agile”	105
	Case Study 2: The Jiu-Jitsu Club That Went Agile, for Real	107
	Where I Failed the Software Company	110
	The Jiu-Jitsu Club Surprise	112
	Ego Versus Adaptation: The Pattern That Keeps Repeating	113
Chapter 10	Continuous Improvement (Not Continuous Chaos)	115
	Real Improvement: Improvement = Saved Time (and Reduced Friction)	116
	Pointless Change: Motion Without Meaning	117
	How to Build Sustainable Systems (Without Getting Stuck)	119
	Practical Ways to Keep Improving Without Burning Out	121

Chapter 11	The Cost of Chasing Trends	129
	Shiny Objects Versus Real Agility	129
	Why Leaders Chase Trends (The Psychology)	129
	How Trend-Chasing Kills Culture	134
	The Real Cost in Time and Money	136
	The Cost in Team Morale and Capability	137
	Trend-Chasing Inside Agile Itself	140
	Red Flags: Are You Chasing Trends?	141
	How to Focus on What Actually Matters	141
	Guardrails for Evaluating New Trends	142
 Chapter 12	 The Future of Agility (If We Build Something That Lasts)	 145
	What This Future Is	145
	Here's the Future Worth Building	146
	What Makes That Future Possible	149
	What This Looks Like Day to Day	153
	Here's Your Part in Building It	156
	Final Thoughts: Build Something Worth Leaving Behind	159
 Conclusion		 161
 <i>Appendix A: The Decoder Ring</i>		 165
<i>Appendix B: Red Flags: Are We Improving or Performing?</i>		173
<i>Appendix C: Behavior-First Core Behaviors (The Ones That Actually Change Outcomes)</i>		179
<i>Appendix D: Continuous Improvement That Doesn't Break People (A Field Guide)</i>		189
<i>Appendix E: Metrics That Don't Lie (and the Ones That Do)</i>		195
<i>Appendix F: Conversation Scripts (Leaders/Coaches/Teams)</i>		203
<i>Bibliography</i>		211
<i>Index</i>		215



ACKNOWLEDGMENTS

Writing a book is weird. You sit alone, argue with yourself, and try to turn years of work into sentences that don't suck. You don't get through that without people around you keeping you honest.

To the clients and teams who let me into their world: thank you. You trusted me with the real stuff: conflict, pressure, politics, burnout, and the occasional miracle. This book is built from that reality.

To the people who reviewed drafts and gave feedback: thank you for calling out the weak parts, the repetitive parts, and the moments where I was being a bit of an ass. You were right more often than I wanted to admit. And you made the book better.

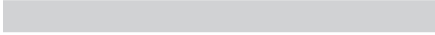
To the great leaders and coaches I've worked with: thank you for showing me what service looks like when it's not a LinkedIn post. You modeled the kind of leadership that actually helps people. And to the not-so-great ones: thanks for the material.

I also want to acknowledge the thinkers who shaped our understanding of learning, systems, leadership, and teamwork long before "agile influencer" became a job title. The ones who did the work shared what they learned and didn't try to turn it into a religion. Even when I disagree with parts of their thinking, I respect the contribution.

This book isn't a rejection of learning. It's a rejection of the gift.

To my family and friends: thank you for your patience while I disappeared into writing. I owe you time back.

And to anyone who's been going through the motions and thought, *There has to be a better way*: this book is for you.



INTRODUCTION

Why am I saying, “Stop Chasing Agile?” Why now? And why am I the one saying it to you?

A few years ago, I wrote *Beyond The Framework: Cultivating Agile Growth*. It was a guide I built and refined over time, based on what I saw actually work in organizations trying to get better, not just look better. The point wasn’t to pick the right framework. The point was building the conditions where improvement can stick.

In business, being “agile” isn’t a badge. It’s the ability to respond fast when the market shifts, customers change their minds, or competitors move. That takes more than a process. It takes an *ethos* and a culture that support adaptability, creativity, and collaboration—especially when things get tense. And here’s what I learned after that book: even when people understand this, they still get pulled back into the worshiping process. They start chasing labels. They start copying ceremonies. They start measuring activity instead of outcomes.

That’s where this book comes in. Over the past decade, I’ve heard the same tired lines on repeat:

“Agile is dead.”

“Agile doesn’t work here.”

“We’re too big to be Agile.”

“We’re too small to be Agile.”

Excuse after excuse after excuse. Most of it comes from the same sources: bad experiences, misinformation, and weak leadership. Honestly, it drives me nuts. I’m tired of people saying Agile doesn’t work when what they really mean is: We did it badly and don’t want to admit it.

Years ago, Dave Thomas said something that should've ended this whole debate: "Agile" is an adjective. When the *Manifesto for Agile Software Development* was written, that's how the word was meant to be used. Go read it. The word "Agile" describes *how* we work. It's not a brand. It's not a framework. It's not a certification ladder. In his GOTO talk, *Agile Is Dead (Long Live Agility)*, he makes the same point I'm making here: agility is about how you behave and adapt—not what you buy, and not what you call yourself.

But somewhere along the way, we messed it up. What started as a push for real change got buried under layers of nonsense. The industry turned into a disorganized mess. "Agile" became a label you can slap on anything: a process, a tool, a job title, a consulting package. "We're Agile now" became a slogan instead of a real shift in how people behave, decide, and lead. So when people say, "Agile is dead," I get it. If *that's* what you think Agile is, maybe it should be. But if your version of Agile is dead, that's not a funeral I'm attending. You can't kill an adjective. You can't kill Agile sales, Agile leadership, Agile development, or Agile marketing. Or anything else you want to make more adaptive.

Agile describes how something works. At its core, agility is about being adaptive, getting better than we were before by cutting red tape, becoming more self-sufficient, moving away from top-down command-and-control, and practicing adaptive leadership. It's not a badge. It's not a personality. It's a response to reality. And reality doesn't move at 1950s speed anymore. This is the 21st century. Information doesn't trickle out slowly and stay put for decades. Customer expectations shift. Markets move. Competitors copy what works. Regulations change. Technology changes. Sometimes the ground moves under your feet in a single quarter. In that environment, the old way isn't just cozy, it's costly.

So, when someone says, "Only this can work," or "We can't be Agile because X," what they're often really saying is: *I'm attached to the old way, and I don't want to change.* That's one of the most annoying things to hear, because it's giving up before you even try. It's also weird because everywhere else in life, we preach persistence. Don't give up. Try again. Keep pushing. But when it comes to agility, suddenly it's, "Pfft. Let's not try. It's new. The old way is nice and cozy."

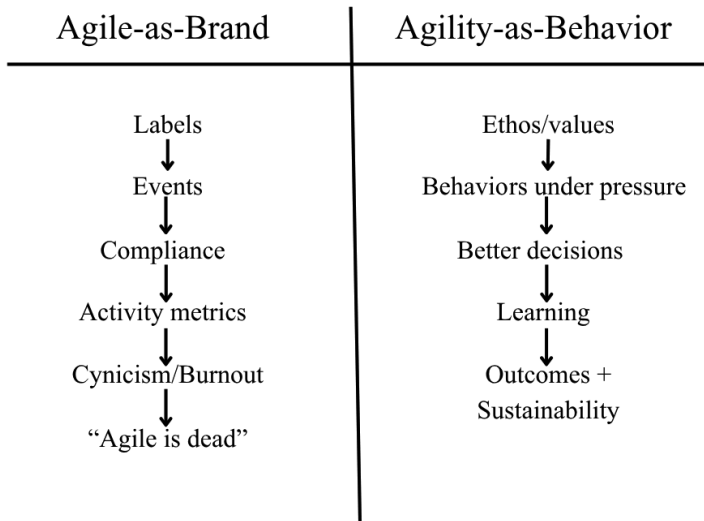


Figure I.1 | Same word. Two very different realities

Most of the ideas people now call Agile leadership, servant leadership, or transformational leadership have been around for decades. This isn't new stuff. We're just finally admitting maybe we should use it. And yet, instead of moving forward, we get stuck in the same arguments on repeat: “Agile is dead.” “Leadership will never buy in.” “Our culture won't change.” I've lost count of how many times I've heard people say, “That won't work here,” like it's a law of physics instead of a choice (see Figure I.1).

That's where the title of this book comes from. Stopping the chase isn't about burning it all down and going back to Waterfall. It's about stepping out of the noise and getting back to what matters: being better at what we do, using the tools that help, and dropping the ones that don't. Scrum, SAFe™, LeSS, Kanban, whatever you use, those are tools. Useful sometimes. Harmful sometimes. Never holy.

The point is simpler than the industry wants it to be: are we delivering value, learning, improving, and are people burning out or thriving? Right now, too many organizations are stuck. They'd rather argue about frameworks than fix behavior.

Ethos Before Mindset

One of the biggest blockers to real agility isn't process. It's a mix of cognitive bias, logical fallacies, and ego, especially in leadership. There's still this belief that if you're a manager, VP, or executive, you're supposed to have all the answers. You're the boss. You're the smartest one in the room. You decide; everyone else executes. That's not leadership. That's outdated, ego-driven nonsense.

Organizations are built on collective intelligence. If you've got more than one person in your company, the group is more intelligent than any single individual. Full stop. The moment a leader defaults to "what I say goes," they shrink the organization to the size of their own ego. Real leaders do the opposite: they focus on improving everyone around them. They build people up and create an environment where teams can think, experiment, and grow, without fear.

I learned this firsthand as a senior executive. Early in my leadership career, it took me a while to realize that my top priority wasn't running the show. It was helping the people around me progress. Their growth was the only way anything would ever work efficiently and effectively for the long haul. When leaders treat people like replaceable parts, they get compliance. When leaders treat people like capable adults, they get judgment, ownership, and adaptation, the stuff everyone claims they want. That's why I talk about *ethos* (see Figure I.2).

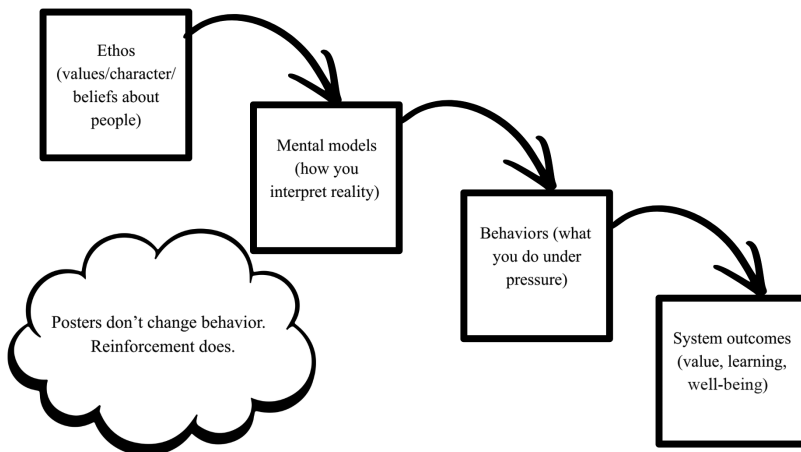


Figure I.2 | Posters don't change behavior, ethos does

Ethos shapes mental models, and mental models shape what people call *mindset*. The industry loves the phrase “the Agile mindset.” Honestly? Hard pass. You don’t start with a mindset. You start with ethos: your character, your values, and what you believe about people and work. Ethos isn’t set in stone; we can grow, change, and rebuild our character over time (I’m one of those stories). But it’s still the foundation. If you don’t examine it, any mindset change turns into window dressing.

You can go to all the Agile trainings you want, memorize the values and principles, and still behave like a command-and-control dictator on Monday morning. Real change starts with what you truly believe, not what you put on a poster. That’s a big reason so many Agile transformations have failed over the last 20+ years: they never changed character or values. They forced teams to “do Scrum” or “do SAFe™” and hoped speed would magically lead to success.

Sometimes things did move faster. But if you’re speeding up bad decisions or poor quality, you’re not improving; you’re just producing bad results more quickly. Garbage in, garbage out... only now it happens at lightning speed. Here’s a real example of how this plays out. I once worked on a massive system rewrite that replaced an entire legacy platform. The stated goal wasn’t better decisions, better data quality, or better customer outcomes. It was simply: Get the data faster. They were receiving data after about two weeks and wanted it within one. That was the win they were chasing. The part that still blows my mind is that they already knew the data was wrong. They weren’t fixing the quality or the underlying issues. They just wanted bad data delivered faster so people could clean it up faster. Think about that for a second. They went through all the motions—sprints, iterations, chunked work, ceremonies. On paper, it looked “Agile.” In practice, it was a costly way to do the wrong thing more quickly.

The program was planned for two years. After I left unceremoniously for raising uncomfortable but valid points, it ended up taking about four times as long as planned, and the budget increased substantially. Did they eventually get something that worked? Maybe. I spoke with the person who replaced me later, and he couldn’t even find the words to describe how it went. The whole thing was a perfect example of what happens when you prioritize speed and process over character, values, and outcomes.

Why This Book, Really

This book is my attempt to shut out the noise and get us back to what matters. Agility works, and it's needed now more than ever, whether you're building software, running a mom-and-pop shop, operating a retail store, leading a marketing firm, or working in the service industry. Real agility (not the branded, buzzword version) is what helps you survive change, make better decisions under pressure, and keep improving without burning people out.

If you think you can keep managing with old-school, top-down, I-have-all-the-answers leadership and still innovate, you're lying to yourself. This is a group effort. Organizations don't get better because one smart person has a strong opinion. They get better when leaders create the conditions for collective intelligence to emerge: clear priorities, quick decisions, honest feedback, and a safe space to experiment. This book is here to help you spot what's really getting in the way, understand why so many Agile efforts fail, and shift from mindset talk to ethos and behavior. My goal is simple: help you build something that works. Not perfect. Not trendy. Not branded. Just better.

With a title like *Stop Chasing Agile*, you might be wondering where this is going. Maybe you expected a framework takedown. Maybe you're waiting for the how-to part. Fair. Let's clear it up right now.

This book isn't a framework manual. I'm not here to teach you how to do Scrum properly, master SAFe™, or chase the latest shiny thing (whatever's trending this quarter). Frameworks are tools. They can be useful. They can also become a distraction. Either way, they're not the point. If what you want is step-by-step instructions, then YouTube, online courses, and certifications have you covered.

It's also not a "10 Steps to Agile" guide. I'm not serving up "7 Secrets to Better Retros," "12 Hacks to Transform Your Stand-Ups," or a checklist you can follow without thinking. There are already plenty of books and blogs for that. Some of them are even decent. But most of it still assumes the same thing: if you install the right process, behavior will magically follow. This book is about what actually makes agility work—regardless of the framework you use (or don't). It's about the human factors: ethos, decision-making, conflict, clarity, feedback, and what leaders do when the plan gets punched in the face. That's the leverage. That's the part

most Agile books politely step around. So no, you won't find a paint-by-numbers guide here. But if you're ready to stop worshiping process and start leading like a human, let's get to work.

So What Is This Book?

It's a noise filter. A practical lens for seeing what agility is, what gets in the way, and how to tell the difference between something that will actually help you and something that's just branding, ego, or recycled advice dressed up as something new. Because let's be honest: there's a lot of crap out there, and there's good thinking too. The problem is that most people don't have a clear way to separate signal from noise when they're being hit with frameworks, labels, certifications, and thought leaders all promising that this is the thing that will finally make Agile work. This book is here to help you cut through that, spot what's useful for you and your team, and understand agility as an environment, *an ecosystem*, not just a process. I'm not here to sell you my own framework. I'm here to help you see clearly and make better choices for your context.

This book is for business owners and founders, senior leaders and executives, team leaders and managers, coaches and Agile practitioners, and team members who want work to suck less and matter more. It's especially for people who've been burned by bad Agile: the big "transformation" that started with a rah-rah kickoff and died six months later; the rollout that created more frustration, more bureaucracy, and less trust; the experience that left you thinking, *If this is Agile, I don't want it*. If you've been through that and you're still here, still reading, it's probably because you believe, somewhere under the frustration, that work can be better. That teams can be healthier. That agility should mean something more than "we do stand-ups now." If that's you, you're in the right place.

This book isn't for people looking for a new label to slap on what they already do. It isn't for badge hunters chasing the next certification. And it definitely isn't for anyone who just wants their favorite framework validated and defended. If someone tries to build a certification out of this book, they've missed the point entirely.

I wrote this to open eyes, not to create another club. I want you to see what real agility means for your organization, your team, your leaders,

and for you as an individual contributor. This is about becoming better, not more “on brand,” not more compliant. Just better. And yes, this book comes from frustration with the current state of the Agile world. But it also comes from hope: hope that we can stop the worshiping process, stop arguing in circles, and move forward.

How This Book Is Structured

This book weaves together four main threads. You’ll see each one show up across the chapters, stories, and tools, sometimes quietly in the background, sometimes right in your face.

First, I name the noise. Not to be edgy for sport, but because a lot of organizations are stuck in patterns they can’t describe. When you can’t name what’s happening, you can’t change it. Second, I draw on decades of research and thinking about systems, leadership, learning, and human behavior that predated Agile Influencer becoming a job title. Third, the book stays behavior-first and ethos-first. I’m not interested in the installation process and hoping behavior follows. I’m interested in what people do under pressure, what leaders reward, and what teams learn to tolerate. Finally, I ground it in real stories, case studies, and small changes that actually work in the mess of real organizations. Let’s break those down.

Calling Out the Nonsense

You’ll see this thread in chapters like *The Agile Graveyard*, *The Framework Fiasco*, and the sections that follow influencer culture and what Dean Peters called *hyper-scrum-mentalism*, the habit of treating one framework as gospel and everything else as heresy.¹ If you’ve spent any time on LinkedIn, you’ve seen the graveyard posts: “This is dead.” “That is dead.” “Here’s the new phoenix; this is the optimal framework now.” It’s ridiculous. Most of it is rebranding, status games, and ego. This section is about naming what’s happening: framework worship,

¹ Dean Peters, “The 4 ‘isms’ of Agile Failures,” *Medium*, 2018, <https://medium.com/deanondelivery/the-4-isms-of-agile-failures-a40c960ace1f>. Peters’ idea of “hyper-scrum-mentalism” and his broader critique of Agile dogma were one of the sparks that pushed me toward writing this book.

certification culture, influencer economics, and shiny new labels slapped on the same old dysfunction. Not because I love ranting (okay, sometimes I do), but because you can't fix what you won't name.

Bringing in the Research (That's Been Around for Decades)

Most of what we'll talk about in this book isn't new. Research on motivation, team dynamics, leadership, and systems thinking has long existed, often dating back to the last century and, in some cases, even earlier.² Agile isn't some brand-new invention. The language is newer, sure, but the principles underneath it are old and well tested. People do better work when they have autonomy, mastery, and purpose. Teams outperform lone heroes when properly set up. Systems shape behavior far more than slogans do. Organizations that learn faster tend to stick around longer. None of that is controversial in the research. What's controversial is how often we ignore it in practice, then act surprised when a transformation turns into frustration. This part brings that research into the book for one reason: not to turn this into an academic slog, but to give you something solid to stand on. If you've been feeling like modern work has drifted into nonsense—more meetings, more process, less progress—you're not being dramatic. There's evidence behind what so many people feel in their gut. The clues have been there for decades. We've just been busy chasing new labels instead of applying what we already know.

Behavior-First and Ethos-First Agility

This thread puts behavior and ethos, your character and values, center stage. If you're wondering, "Is it behavior-first or ethos-first?" the honest answer is both. They're tied together. Your ethos shapes your behavior: the choices you make, what you avoid, what you tolerate, and how you

² For example, on motivation and human needs, see Douglas McGregor, *The Human Side of Enterprise* (McGraw Hill, 1960); Frederick Herzberg, *Work and the Nature of Man* (World Publishing, 1966); and Edward L. Deci and Richard M. Ryan, *Intrinsic Motivation and Self-Determination in Human Behavior* (Plenum, 1985). On team dynamics and group effectiveness, see J. Richard Hackman, *Leading Teams: Setting the Stage for Great Performances* (Harvard Business School Press, 2002). On systems thinking and learning organizations, see Peter M. Senge, *The Fifth Discipline: The Art and Practice of the Learning Organization* (Doubleday, 1990).

treat people under pressure. At the same time, consistently changing behaviors can reshape your ethos. Not overnight. Not with a slogan. But through practice and reinforcement. Think of ethos as the foundation and behavior as the building blocks. Without a solid foundation, the building won't stand. But without the blocks, the foundation never becomes a home. That's why this book treats Agile values as lived practices, not words on a poster or a slide deck.³ If your values only show up in a kickoff deck, they're not values. They're marketing. And yes, this means changing some behaviors. Maybe even confronting parts of your character. That's the uncomfortable part. But if you're still reading, I'm betting you're not here just for a better process. You want to show up better, help your team get better, and make your organization better. This isn't the "think Agile thoughts" section. This is the "look in the mirror" section.

We're going to dig into what you do under pressure, what you really believe about people and work, and how those beliefs leak into the way you lead, decide, and respond. Not "memorize the Manifesto." Not "say the right words." More like: change how you behave, change what you tolerate, and change what you reward. Until those three things shift, nothing else—frameworks, tools, events—will matter for very long.

Real Stories, Case Studies, and Small Changes

You'll get real stories in this book, some that make me look good, and some that don't. That's the point. If all you ever hear are polished success stories, you don't learn much. The valuable stuff lives in the messy middle: the times we got it wrong, the times ego got in the way, and the times we thought we were doing Agile when we were really just rearranging the furniture. Most of what works isn't dramatic. It's small, consistent changes in how we lead, communicate, decide, and learn. Those changes don't make for flashy LinkedIn posts, but over time they're what make an organization genuinely adaptive. They're also the only kind of change

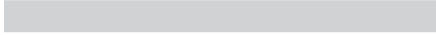
3 On underlying values and assumptions shaping behavior and culture, see Edgar H. Schein, *Organizational Culture and Leadership* (Jossey-Bass, multiple editions). On mental models as the often-invisible lenses through which we interpret reality, see Peter M. Senge, *The Fifth Discipline* (Doubleday, 1990). On mindset as beliefs about learning, ability, and challenge, see Carol S. Dweck, *Mindset: The New Psychology of Success* (Random House, 2006).

that tends to survive the next re-org, the next executive shuffle, or the next new operating model announcement.

I've had to unlearn this myself. Early on, I thought leadership meant control: be the boss, have the answers, keep a tight grip. Turns out that grip was the bottleneck. Things improved when I stopped trying to control people and started fixing the system where they were trapped. Then, I helped them grow instead of squeezing them for output. That's what this book is about. Not a new framework or label, but a different way of leading real humans through hard work.

One Last Thing Before We Dive In

Before we get into the chapters, here's the deal. This book isn't here to defend a framework or hand you a new method to install. It's here to look at what actually drives results when things get messy, and what we each reinforce in the system when we're under pressure. If you're open to questioning assumptions, habits, and yes, ego, keep reading. If you're willing to look in the mirror and ask, "What am I contributing to the way work feels and flows here?" you're exactly who I wrote this for. And if what you really want is a story where the method is always right and the people are always the problem, parts of this will feel uncomfortable. That discomfort isn't the point by itself, but it can be the doorway to better conversations, clearer decisions, and a healthier way of doing hard work together.



HOW TO READ THIS BOOK (DEPENDING ON YOUR ROLE)

Not everyone comes to this book from the same place. Some of you are executives. Some are managers. Some are coaches, Scrum Masters, or consultants. Some of you are on the front lines, trying to get work done without losing your mind. You'll notice different things in here depending on where you sit, and that's a good thing.

If you're an executive or senior leader, read this book as a mirror, not as a manual. Pay close attention to the sections on ethos, power, and ego, and to the stories where leadership behavior quietly kills adaptability. Don't just focus on what teams should do. Watch for the system signals you're sending, what gets rewarded, what gets punished, and what people learn is unsafe to say out loud. As you read, ask yourself: Where am I part of the problem? What am I rewarding, real improvement or good-looking activity? What am I tolerating that I'd never admit publicly? If you only use this book to judge your teams and never question your own behavior, you'll miss the point.

If you're a middle manager or team lead, you're probably stuck in the squeeze: pressure from above, pain from below. Read this book to name what you're seeing, find language to push back constructively, and spot small, fundamental changes you can make within your span of control. You may not be able to change the whole organization, but you can change how it feels to work for you, and that matters more than most people realize. Ask yourself: What can I change in how I lead, even if nothing above me changes yet? Where am I unintentionally passing pressure and fear down the chain? What's one behavior I can model that makes it safer for my team to tell the truth?

If you're a Scrum Master, Agile coach, or consultant, this book might poke at your identity a bit. That's intentional. Read it as a challenge to any framework loyalty you've picked up over the years, and as a reminder that your job is to change behaviors and systems, not just run events. Use it as a toolkit for spotting hype, both within the industry and within your own practice. Ask yourself: Where have I been selling process instead of helping people think? Am I more attached to my favorite approach than to my client's reality? What would it look like to be behavior-first, not framework-first, in my work? If you feel defensive at points, don't rush past it. That's usually where the useful learning is.

If you're a team member, individual contributor, or frontline worker, you might not have "leader" in your title, but you're the one living with the consequences of all this. Read this book as validation that you're not crazy, as language for what you're experiencing, and as a way to spot when you're being sold something that looks good but doesn't help. Start with what's in your control right now. Look for small ways to nudge your and others' behavior in a better direction. Ask questions that challenge assumptions and invite honesty. Speak up constructively when it's safe, and when it isn't, find allies who can help carry the message. You don't have to wait for a formal transformation to start acting with more clarity, courage, and care. Ask yourself: What do I want to start asking my leaders and coaches that I haven't asked before?

If you are in hybrid roles (manager and contributor, coach and internal leader, founder and everything else), read the book once with one hat on, then again with another. Different parts will sting. Different parts will feel like relief. Different parts will give you ideas you didn't see the first time. However you read it: if you're willing to question your own assumptions, you'll get something out of it. If you're only here to defend your favorite framework, you won't. Use this book as a mirror and a starting point for better conversations, not as a new set of rules.

Happy reading!

ABOUT THE AUTHOR

Jeremy Berriault is a human-centered performance architect and founder of Berriault and Associates Consulting Group, where he helps leaders recognize when performance comes at a human cost—and rebuild teams that can adapt, perform, and sustain success. His work is grounded in a behavior-first philosophy: real agility isn't a title or framework, but how people show up and work together under pressure. Drawing on decades of experience, he has developed practical tools and operating rhythms that cut through competing priorities, sharpen decision-making, and embed change that actually lasts. Jeremy has supported leaders across industries—from software and finance to hospitality and fitness—earning recognition for helping teams build clarity, trust, and healthier execution.



A seasoned advisor, speaker, and coach, Jeremy is known for translating complex organizational challenges into clear, actionable insights. His approach blends research-backed assessments with hands-on workshops and coaching, ensuring that transformation efforts are both practical and durable. Over a career spanning more than 30 years, he has held roles from quality assurance leadership to enterprise delivery and executive positions, guiding cross-functional teams toward meaningful business outcomes. He holds an MBA and graduate management credentials from Athabasca University, along with a Bachelor of Applied Science in Psychology from York University. Outside of his professional work, he trains in jiu-jitsu—a discipline that reflects his belief in honest pressure, immediate feedback, and continuous improvement.



This book has free material available for download from the Web Added Value™ resource center at www.jrosspub.com

At J. Ross Publishing we are committed to providing today's professional with practical, hands-on tools that enhance the learning experience and give readers an opportunity to apply what they have learned. That is why we offer free ancillary materials available for download on this book and all participating Web Added Value™ publications. These online resources may include interactive versions of material that appears in the book or supplemental templates, worksheets, models, plans, case studies, proposals, spreadsheets and assessment tools, among other things. Whenever you see the WAV™ symbol in any of our publications, it means bonus materials accompany the book and are available from the Web Added Value Download Resource Center at www.jrosspub.com.

Downloads for *Stop Chasing Agile* include the Appendices at the end of this book for use in your organization.

CHAPTER 1

THE AGILE GRAVEYARD

It's a dark, gloomy night. You're walking through an old graveyard. The air is cold enough that you can see your breath. Leaves rustle in the wind. A thin mist curls around your feet with every step, like the ground itself is exhaling. You sweep your flashlight across the grass, and the shapes start to appear: gravestones, row after row, stretching out into the fog. You step closer to one and angle the light. The engraving comes into focus:

"Scrum—Dead."

You move to the next stone:

"SAFe™—Dead."

Then another:

"Agile—Dead."

And another:

"Jira—Dead."

Everywhere you look, it's the same story. Dead. You feel the weight of all this loss pressing down on you. Everything you've worked with, everything you've tried to use to create value, is apparently buried here. Whatever hope you had starts to feel a little shaky. If all of this is dead, what's left?

What's left is what doesn't fit on a slide deck: how you show up when things get hard. Because if you're going to do any of this, you might

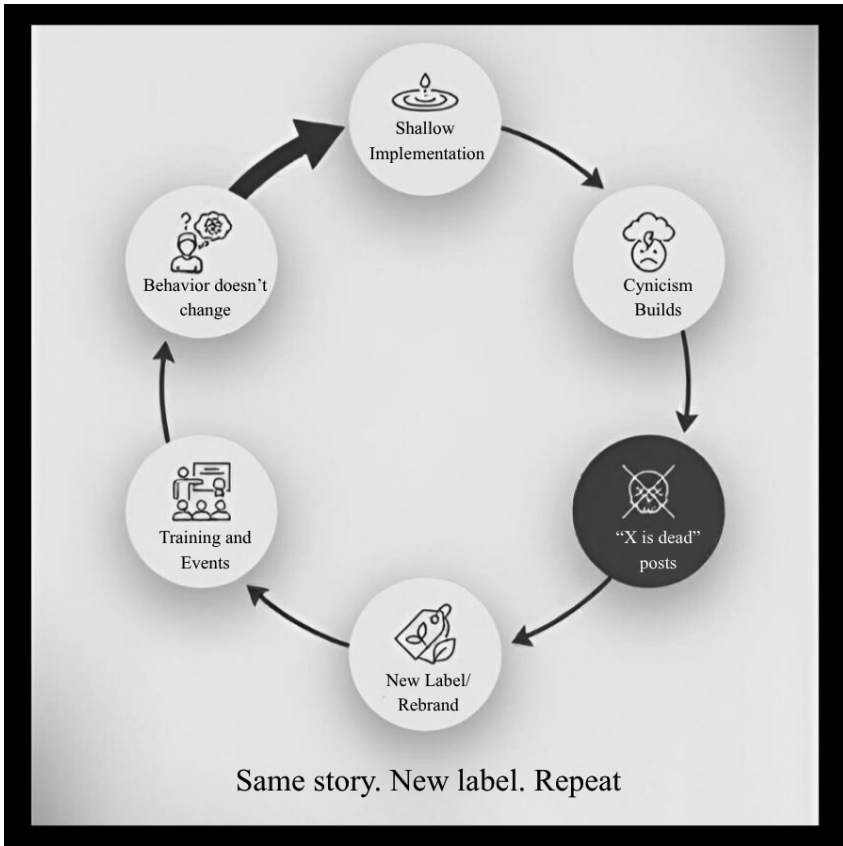


Figure 1.1 | The Agile graveyard cycle

as well build something that lasts. According to the loudest voices on the internet, this graveyard is real (see Figure 1.1). Everything is dead, useless, obsolete, or not fit for the modern world. It's a constant stream of funerals, delivered with absolute confidence and zero curiosity.

So here's the question you have to ask yourself as you stand there with that flashlight: Is any of this really dead? Or are we just addicted to declaring things dead because it gets attention, clicks, comments, shares, speaking slots, and the next wave of hype? You see the cycle everywhere: on LinkedIn, Reddit, conference stages, and in blogs. Post after post, talk after talk, someone's always declaring, "X is dead," "Y is obsolete," "Z is the only way forward now." We end up stuck in a never-ending funeral for last year's ideas, always chasing the next big thing, always

ready to bury what came before. And the irony is that most of the time these things aren't dead at all. They're just out of fashion. Or someone has a new product to sell.

Meanwhile, the real work, the stuff that helps teams and organizations get better, gets lost in the noise. Here's what doesn't change just because the internet got bored: there's decades of research showing that self-managed teams can outperform traditional, tightly controlled ones; that people tend to feel and perform better when they have autonomy; that work gets done faster and with higher quality when teams are trusted and supported; and that leadership approaches beyond command-and-control aren't just trendy—they're effective. People respond to them because they're human.^{1,2} Before we start digging more graves, we should pause and ask a better question: Are we learning from what works, or are we chasing the next funeral?

The need for agility isn't new. The human principles behind it—adaptation, learning, trust, and autonomy—have existed for centuries. What's new is the name, the branding, and the industry that grew around it. Agility didn't begin with 17 people at a ski resort writing the Manifesto for Agile Software Development in 2001. That wasn't the birth of agility. It was a moment where people shone a light on something that already existed. What those folks did in 2001 was courageous. They looked at the way software was being built and said, "We can do better." Their focus was software, sure, but the ideas spread because the underlying problems weren't unique to software. The same patterns show up in marketing, operations, service work, retail, healthcare, and small businesses: unclear priorities, slow decisions,

1 On self-managed teams and the conditions under which they outperform traditional, tightly controlled structures, see J. Richard Hackman, *Leading Teams: Setting the Stage for Great Performances* (Harvard Business School Press, 2002). On autonomy, intrinsic motivation, and performance, see Edward L. Deci and Richard M. Ryan, *Intrinsic Motivation and Self-Determination in Human Behavior* (Plenum, 1985), and later work on Self-Determination Theory, which shows that autonomy, competence, and relatedness are key drivers of well-being and effectiveness at work.

2 For research on leadership approaches that outperform traditional command-and-control and foster learning and performance, see Amy C. Edmondson, "Psychological Safety and Learning Behavior in Work Teams," *Administrative Science Quarterly* 44, no. 2 (1999): 350–383; Amy C. Edmondson, *The Fearless Organization: Creating Psychological Safety in the Workplace for Learning, Innovation, and Growth* (Wiley, 2018); and Ronald A. Heifetz and Marty Linsky, *Leadership on the Line: Staying Alive Through the Dangers of Change* (Harvard Business Review Press, 2002).

fear-based leadership, and systems that punish learning. Agility grew. It evolved. That's the whole point.

Agility isn't what's dead. What's buried here are the half-baked, cargo-cult implementations that never had a chance, because they tried to install a process without changing behavior, ethos, or leadership. But instead of learning from that, we blame the tombstone and move on to the next fad.

Agility Isn't New

I'm not going to claim agility has been around since the dawn of humankind 6 million years ago (thank you, Google), but it has been around a long time. If you zoom out and look at how we've evolved as a society, you start to see it everywhere. Any time people had to innovate, adapt, learn from feedback, or change direction when reality changed on them, they were doing the thing we now call Agile, long before anyone put it in a full-day workshop.

One of the biggest myths we've built over the last couple of decades is that an Agile environment must always produce innovation as the end goal. If it's not churning out disruptive products and viral success stories, it must be failing. But innovation isn't the endpoint. It's a byproduct. The same goes for speed. People love to talk about going faster as if that's the whole point of agility. It isn't. Speed is what you get when you remove friction and confusion from a system that's already set up to learn and adapt.³

The real purpose of agility is simpler and more grounded than the industry makes it out to be: adapt to your environment, respond to variables rather than pretending they don't exist, replan when new information comes in, and improve as you go rather than locking yourself into a bad plan and hoping for the best. That's the job. Not "move faster at all costs." Not "ship more." Learn, adjust, and get better, without breaking people.

You can see these principles even in places you'd assume are rigid, like the military. On paper, the military is strict and deeply hierarchical. But if you look at how effective units operated in the world wars and

³ This aligns with the Agile Manifesto's emphasis on "responding to change over following a plan" (2001) and with research in Lean and DevOps that frames speed and innovation as outcomes of healthy systems focused on flow, feedback, and learning. See, for example, Nicole Forsgren, Jez Humble, and Gene Kim, *Accelerate: The Science of Lean Software and DevOps* (IT Revolution Press, 2018).

beyond, you see something more practical than pure top-down control: small teams adapting in real time to chaos on the ground. You see commanders giving intent and boundaries, then trusting the people closest to the action to make decisions. They called it decentralized command.⁴ It's not that different from what we mean when we talk about self-organizing teams. There's a clear overall goal. Teams are given intent and constraints. They're trusted to figure out how to achieve the goal. In *Extreme Ownership*, Jocko Willink frames decentralized command as a core leadership principle, and when you read the stories, something clicks: we're totally fine trusting small teams to make life-or-death decisions in chaotic, high-risk environments... but we somehow can't trust a software team to decide how to slice a feature without three layers of approval. Think about how backward that is.

On the battlefield, leaders set clear intent, teams understand the mission and constraints, and decisions get pushed down to the people closest to the action, because that's where the information is. In most organizations, leaders set vague slogans, teams get tasks instead of intent, and decisions get pulled up to the people farthest from the work, because that's where the titles are. We accept decentralized command when bullets are flying, but when it's just code, marketing campaigns, or service work, suddenly we need command-and-control. That's not a capability problem. That's an ethos problem.

And just so we're clear on what I mean by ethos: ethos is your lived values—what you actually believe and reward, not what you print on a poster. If you believe people can't be trusted, you'll build systems that treat them like they can't be trusted. You'll add layers of approval, status reports, and signoffs that quietly say, "We don't believe you'll do the right thing unless we watch you." If you believe people are capable, you'll build systems that let them prove it. You'll design for clarity, feedback, and autonomy instead of control.⁵ That belief, your ethos, is what shapes the environment.

4 On decentralized command and mission-type tactics in military history, see Martin van Creveld, *Command in War* (Harvard University Press, 1985), and discussions of *Auftragstaktik* (mission command) in German and Allied doctrine. For a modern leadership framing, see Jocko Willink and Leif Babin, *Extreme Ownership: How U.S. Navy SEALs Lead and Win* (St. Martin's Press, 2015).

5 On how leaders' underlying assumptions about people (trust vs. control) shape the systems they build, see Douglas McGregor, *The Human Side of Enterprise* (McGraw-Hill, 1960), which contrasts "Theory X" (people must be controlled) with "Theory Y" (people are capable and self-motivated), and Edgar H. Schein, *Organizational Culture and Leadership* (Jossey-Bass, multiple editions).

Agility as an Ecosystem

This is what I mean when I talk about an Agile environment, or better yet, an ecosystem. Different language, same principles. The goal of an Agile environment isn't perfection. We're never going to get there. Perfection is a moving target, and chasing it usually makes people miserable, risk-averse, and scared to experiment. The goal is to create a work ecosystem where a few healthy practices become the norm. Ideas flow instead of getting stuck in someone's inbox for three weeks while everyone pretends they're "blocked." Work gets done in a way that feels sane, not like a constant emergency where every day is a new crisis. People don't dread Monday morning because they aren't bracing for another arbitrary fire drill. They have space to think things through instead of reacting to the latest ping, and new solutions can come from anywhere, not just from the highest-paid person in the room.

In a healthy ecosystem, new ways of working get explored, tested, and either adopted or dropped based on what works, not based on who suggested them. Red tape that doesn't serve a real purpose gets removed. Unnecessary bottlenecks and handoffs are reduced. Teams can get to the finish line without being blocked at every turn by outdated command-and-control. That was the good intention. That's what agility was meant to support. Not a specific set of events. Not a specific brand of framework. A different kind of ecosystem.

Somewhere along the way, we buried that intention under process worship, tool obsession, and a whole lot of one-true-way thinking. We made the ecosystem about the plants, the tools, and frameworks, instead of the soil: the ethos, structures, and behaviors that let anything grow in the first place (see Figure 1.2).

Later in the book, we'll dig into agility as an ecosystem; how leadership, structures, incentives, tools, and behaviors interact; and how small changes in one part can either support or suffocate the rest. For now, hold on to this: if your environment makes it hard for ideas to flow, for people to think, and for teams to adapt, it doesn't matter what you call it. It isn't Agile.

So, Why Are We in a Graveyard?

If everything is supposed to evolve, why hasn't agility? Why are we still struggling with it? Why do we keep hearing *Agile is dead*? It's not

The Agile Graveyard

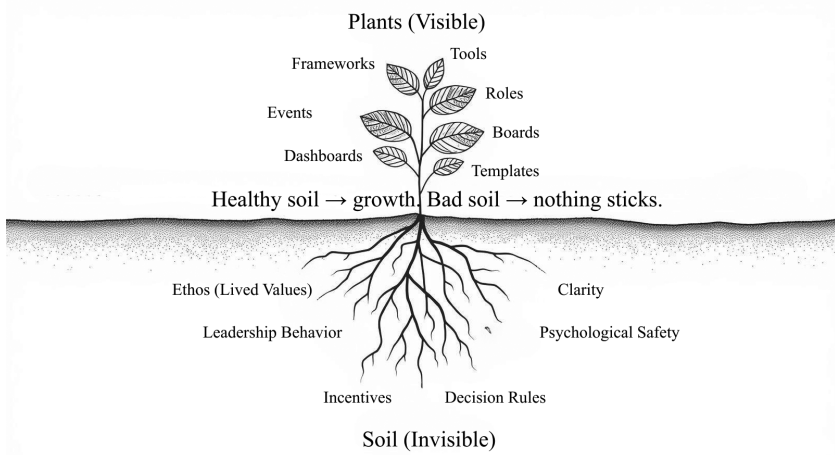


Figure 1.2 | Plants versus soil (visible versus invisible)

because the ideas are broken. It's because we've lost our way. We've let other things get in the way: leaders giving lip service to Agile and work-life balance while behaving the same old way; organizations slapping Agile labels on rigid structures; and frameworks being treated like religion rather than tools. Almost every leader can say the right buzzwords: *We're an Agile team. We care about work-life balance. We empower our people.* The harder question is: How many leaders live it when it costs them something? If you walk around most organizations and ignore the posters, you'll see a very different story—Agile teams that still need three approvals to release anything; work-life balance cultures where people get praised for answering emails at 11 p.m.; and empowered teams that get overruled the moment their decisions make someone uncomfortable. On the wall, it's Agile. In the room, it's the same old command-and-control.⁶

So we fall back into the same patterns, the same behaviors and structures that caused the problems in the first place. And in this

⁶ On how underlying culture and structures pull behavior back to the status quo, even when organizations claim they want change, see Edgar H. Schein, *Organizational Culture and Leadership* (Jossey-Bass, multiple editions), and Chris Argyris, *Overcoming Organizational Defenses: Facilitating Organizational Learning* (Allyn & Bacon, 1990).

graveyard, we've seen the movie before. Things "die," then come back as a phoenix... or a zombie. They get resurrected, renamed, and rebranded. Just like wash, rinse, and repeat.

We even have our own rituals: Plan–Do–Check–Act. We treat it like gospel. We treat agility like it's just another cycle to run through: plan the transformation, do the training, and check the boxes. Then we end up with Frankenstein frameworks—pieces of this, fragments of that, stitched together and sold as *the hybrid that will finally work*. You've probably seen it: a bit of Scrum here, a slice of SAFe™ there, some Kanban terms sprinkled on top, and a custom maturity model to make it look scientific. When it doesn't work, we still don't question ethos, behavior, or leadership. We don't ask what we actually believe about people and control. We don't ask what we're rewarding. We don't ask what we're afraid to let go of. Instead, we shrug and say, "Agile doesn't work here." And then we go back to the graveyard and add another stone.

The Baggage We've Created

All these failures, half-baked transformations, and endless rebrands have turned the term *Agile* into baggage. And not the good kind. Not the suitcase you bring back from a great trip, full of souvenirs, photos, and memories. This is the kind of baggage that's been loaded with rocks, heavy, unpleasant, and passed around by people who don't care what's inside. That's what we've done to Agile.

We've created an industry where the word itself is loaded. You can feel it the moment it comes up in a meeting. People roll their eyes. Teams groan when a transformation is announced. Leaders treat it like just another management fad that will blow over if they wait long enough. That's the graveyard you're walking through, not a cemetery of tools and frameworks that failed us, but a graveyard of misuse, misunderstanding, and missed opportunities.

This is a pattern you've probably seen up-close. An organization sends teams to training. Maybe it's an intense full week. Maybe it's one day a week for three months. They bring in an Agile coach. They roll out a framework. They run some sprints. They add stand-ups and retros. They put up a board on a wall or in Jira. On the surface, it looks like a lot is happening. There's motion. There are new words. There are new meetings.

People can point to activity. And then someone, usually in a leadership meeting, says the magic words: *We're an Agile organization now.*

Meanwhile, under the surface, almost nothing that matters has changed. Leadership hasn't changed how they behave. Decision-making is still top-down. Priorities are still unclear and change on a whim. People are still afraid to speak up when something's broken. The system still rewards compliance and punishes bad news. The only difference is that now the same old problems are happening inside a new set of ceremonies. But hey, teams had training. So we must be Agile... right? This is how you end up with "Agile is dead" stories. Not because the ideas are flawed, but because the implementation was superficial and misaligned with how the organization actually functions. In other words, the label changed. The behavior didn't. Implementation: dead.

The Coaching Problem (and Comfort Problem)

There are great Agile coaches out there. You've probably met a few. They're the ones who coach, not just present a deck. They sit with teams and leaders. They ask uncomfortable questions. They don't just teach people how to do stand-ups; they help executives change how they show up. They push on character and values, not just events and tools.

And then there's another pattern: coaching that stays safely at the team level. A coach runs a two-day training, hands over a slide deck, and moves on. Or they spend months tweaking boards and stand-ups because leadership is "out of scope," too political, or too hard to confront. The problem is that teams don't operate in a vacuum. You can't workshop your way out of a system that rewards interruption, avoids trade-offs, and punishes honesty. Even the best coach can't fix one core issue: many people don't actually want to change. Comfort is powerful, even when the current mess is clearly not working. What looks like resistance to change is often resistance to loss⁷. Loss of control. Loss of status. Loss of certainty. Loss of familiar routines and the stories people tell themselves about why their role matters. Change is constant; we adapt to weather, markets, and technology all

⁷ Ronald A. Heifetz and Marty Linsky famously argue that "people do not resist change, per se. People resist loss." See Ronald A. Heifetz and Marty Linsky, *Leadership on the Line: Staying Alive Through the Dangers of Change* (Harvard Business Review Press, 2002).

the time. What people resist is what they might have to give up in order to change.

You can hear it in the questions nobody wants to ask out loud: *If we really empower teams, what happens to my role? If we stop doing things my way, do I still matter here? If we admit this isn't working, what does that say about my past decisions?* No coach, no framework, and no tool can override that if the people in power are more committed to protecting comfort than improving the system. It's easy to blame coaching. The deeper problem is comfort. If you want a modern example, look at artificial intelligence (AI). It's the latest tool shaking up industries, and people worry it will take jobs or disrupt everything. Underneath that fear is the same old resistance to loss: control, certainty, and familiar ways of working. And this isn't new. In the early to mid-1990s and into the 2000s, software testing underwent a similar shift toward test automation.⁸ I was working with testing groups, leading testing projects and programs, and the fear was intense: *If we don't learn automation, we'll lose our jobs. Automation will replace testers.*

My argument then is the same one I'd make about AI now. Yes, automation replaces repetitive, low-value tasks. Yes, it handles regression and repeatable checks faster than a human ever could. But that's the point. Used well, automation frees people to be more valuable, not less. It creates room for more strategic thinking about risk, better test design, deeper collaboration, and better judgment about what matters and why.

AI is no different. Used well, it can help us do things better, reduce grunt work, and force us to level up our thinking and skills. Used poorly, it amplifies existing dysfunction, accelerates poor decisions, and makes people feel even more disconnected and threatened. The pattern repeats: a new tool appears, fear spikes, and people use it to either evolve or reinforce the same old problems. It's not the tool. It's the implementation. The same applies to agility. Here's an analogy that still fits. Imagine a zoo putting a jaguar in a polar bear enclosure. Technically, the jaguar will survive. It can breathe, move, and eat. But it's going to be miserable.

⁸ On recurring fears about automation and job loss, and how technology tends to reshape work rather than simply eliminate it, see James Bessen, *Learning by Doing: The Real Connection between Innovation, Wages, and Wealth* (Yale University Press, 2015), and Erik Brynjolfsson and Andrew McAfee, *The Second Machine Age: Work, Progress, and Prosperity in a Time of Brilliant Technologies* (W. W. Norton, 2014).

Wrong climate. Wrong terrain. Wrong everything. The problem isn't the jaguar. The problem is the environment.

That's what a lot of Agile transformations feel like. You take a team and slap Agile language and events (stand-ups, retros, boards, and sprints) on them while keeping everything else the same. The rigid hierarchy stays. Command-and-control behavior stays. Fear stays. On the surface, it looks like something changed. Underneath, nothing did. You've put agility, the jaguar, into a hostile environment and then blame the jaguar when it doesn't deliver. And bad implementations aren't neutral. They're cruel. They convince people that agility itself is the problem, when the enclosure was never designed to support it in the first place.

Whether you're building an Agile environment, introducing a new process, adopting a new tool, or rolling out AI and automation, it comes down to one thing: behavior change. If behaviors don't change, nothing else really does. The ethos stays the same. The values stay the same. The way decisions get made stays the same. And if those don't change, your implementation is dead on arrival, no matter how many new terms or tools you throw at it.

To make agility work, really work, you have to change how leaders behave, how teams interact, how decisions are made, and how you respond to uncertainty and feedback. That's the difference between a story that ends with "Agile is dead" and a story that ends with "Our implementation was shallow, misaligned, and never had a chance." Let me give you a real example of how this plays out. I once worked with an organization that said they wanted to "be more Agile." Not just in name, they wanted to be more adaptive, more efficient, and more effective across the whole organization. So we started the right way: with an assessment. We looked at how work flowed, how decisions were made, how teams interacted, and where the real bottlenecks and pain-points lived. From that, I built recommendations to support real agility across the system: structural changes, changes in leadership behavior, and new ways of working that would make it easier to adapt and respond, not just for a single team but across the whole organization. On paper, they agreed. In practice, what they really wanted was something else: Just train the teams.

What they meant was: teach teams how to “be Agile.” Show them techniques and tools. Help them run events, use boards, estimate, slice work, and build better habits. But don’t touch the rest of the system. Don’t change policies. Don’t change decision-making. Don’t change leadership behavior. Don’t change what gets rewarded. The plan was straightforward. There were eight teams. We’d work with two teams at a time over roughly three-and-a-half months per pair, meeting weekly with each team to teach techniques, tools, and habits meant to improve adaptability and flow. My intent was twofold: help the teams genuinely improve their work and, in turn, show leadership what else needed to change beyond training, including structures, policies, decision patterns, and their own behaviors.

Here’s what happened. After the first two teams completed their cycle, the organization stopped the transformation. No work with the remaining six teams. No bigger organizational changes. No follow-through on assessment recommendations. They got some training, then pulled the plug. And what happened next is exactly what you’d expect: the teams gradually reverted. The new techniques were forgotten or used only partially. The environment around them hadn’t changed, so the change couldn’t stick. This wasn’t a failure of the teams. It wasn’t a failure on the part of the people trying to help. It was a failure of implementation, driven by leadership’s fear of loss. Leadership wanted the benefits of agility without changing the system that made agility impossible. They feared losing control, status, and a familiar way of leading. So the transformation died. Not because agility doesn’t work. Not because the teams couldn’t handle it. It died because leadership chose comfort over change.

When you’re walking through the Agile graveyard, it’s tempting to blame the tools. It’s easy to say, *Scrum failed us*, *SAFe™ doesn’t work*, *coaches didn’t deliver*, or *the teams weren’t ready*. That story is comforting because it lets us point to something outside ourselves. But if you look closely at most of these stories, the pattern usually isn’t that Agile is broken. It’s closer to this: we tried to change everything except how we behave. We changed terminology, but not how decisions get made. We added events, but didn’t make it safe to tell the truth. We introduced boards and tools, but didn’t create clarity or trust. We trained teams but

never touched leadership ethos. That's not an agility problem. That's a behavior problem.

When implementations fail, it's almost never because a framework is inherently flawed or a team is incapable of learning new techniques. It's because leaders won't relinquish control. It's because people are more afraid of loss than of committing to growth. It's because the organization's character and values stay the same. You can't build an adaptive environment on fear, ego, and control with a thin layer of empowerment slogans on top.

So if agility isn't dead, and I don't believe it is, and if the tools aren't the real issue, and if the graveyard is mostly full of bad implementations, then we have to stop asking: *Which framework will save us?* and start asking harder questions: *What behaviors are we rewarding, what do we value when things get messy, and who are we, leaders and teams, when things get hard?* Frameworks aren't the enemy. The way we cling to them, sell them, and weaponize them is.

That's where we're going next. In Chapter 2, we'll dig into The Framework Fiasco: how frameworks turned into religion, why "by the book" so often goes off the rails, and what changes when you stop treating a framework as your identity and start treating it as one tool in a much bigger toolbox. If you're willing to examine your relationship with frameworks and your need for certainty and labels, turn the page.

