



BY CLAIRE DRUMOND

Is the Agile Manifesto still a thing?

Standing amidst the technological revolution, we find ourselves asking if the Agile Manifesto should still be our guide as we move into a world defined by continuous innovation. This short, but game-changing document helped take us from shipping products like they were cargo on boats to same-day delivery by drone. But today, we're less pioneers, and more like explorers on the seas of continuous improvement, which makes us wonder, is it time to improve the Manifesto, too?

The origin story

In early 2001, against the backdrop of the Wasatch Mountains, in Snowbird, Utah, 17 people met to discuss the future of software development. The group's members shared a frustration about the current state of affairs, even if they disagreed about how to remedy the situation.

The problem, they agreed, was that companies were so focused on excessively planning and documenting their software development cycles that they lost sight of what really mattered—pleasing their customers.

Companies may have touted corporate values like “excellence” and “integrity,” but these values did little to guide people—especially software developers—toward a better way. That needed to change. Many of the Snowbird 17 already had ideas about how to usher in software development’s new era. The trip to the mountains was their chance to hash it out.

The Agile Manifesto emerged from this extended weekend at just 68 words, and the short and sweet document went on to change software development forever. In the nearly two decades since its creation, these words (and the 12 principles that follow) have been embraced (in varying degrees) by countless individuals, teams, and companies.

culture, defined

Today's agile landscape can seem cluttered with methodologies that promise to take agile ideals and turn them into real-world realities. But today's methodology madness isn't anything new.

The Manifesto itself was born out of a need to find a common ground among [Scrum](#), Extreme Programming, Crystal Clear, and other frameworks.

"They were starting to see that there was something common that they were doing. But at the time, they were very much competitors, at least competitors in thought," said Ian Buchanan, Principal Solutions Engineer for DevOps at Atlassian. "When you put that into context, the fact that they could agree on some set of anything is kind of profound."

The Snowbird 17 wanted to see if representatives of their different disciplines could agree on something—anything. And to their surprise, they did. They agreed on a set of values that defined a culture.

Here they are:

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.

Kent Beck

James Grenning

Robert C. Martin

Mike Beedle

Jim Highsmith

Steve Mellor

Arie van Bennekum

Andrew Hunt

Ken Schwaber

Alistair Cockburn

Ron Jeffries

Jeff Sutherland

Ward Cunningham

Jon Kern

Dave Thomas

Martin Fowler

Brian Marick

Sign up for more articles

Email

email@example.com

Subscribe

The [Twelve Principles of Agile Software](#), also a product of the Snowbird summit, expand on the handful of sentences that

make up the values.

And that's it. Since then, the Agile Manifesto site has changed minimally, if at all. But the world surrounding agile couldn't be more different.



The great agile debate

The Snowbird 17 succeeded in unifying their differing points of view under a few core tenets, but the debate didn't end there. In some ways, agile has shattered into many more ways of operating than the visionaries came to the table with in the first place. It seems as if everyone has their take on agile.

Today, there's SAFe. There's LeSS. There are applications of agile that don't have anything to do with software development, even though the Manifesto starts off by saying: "We are uncovering better ways of developing software by doing it and helping others do it."

According to TechRepublic [[linked](#)], NPR used agile to reduce programming costs by 66% and mentions three other non-tech applications of agile practices.

Dave West, CEO of Scrum.org, who travels to various organizations observing agile practices, called out a research team that's using agile to develop a cure for genetic blindness using viruses.

Indeed, embracing agile outside the realm of software has caught on, but it's not necessarily what the Manifesto's originators intended.

"It's not that it can't be interpreted, but it takes deeper understanding to make sure the ideas are translated with fidelity," Buchanan said.

That deeper understanding isn't always available—even within software development.

The agile industrial complex

Many argue that "faux agile," as it's also called, and its evil twin "dark agile," are exacerbated by the monetization of agile

—adapted from [The Agile Industrial Complex](#) by Matt Heusser

education and consulting. Some even go as far as calling the organizations behind this monetization “The Agile Industrial Complex.”

“There’s cargo cult agile where you’re doing and saying the right things, but you don’t understand the fundamental principles. You’re not getting the results,” Buchanan said.

Some, and many do, consider Atlassian a culprit of this since our tools enable agile frameworks like Scrum and [kanban](#). But our belief is that agile is a cultural value, and teams should be empowered to work how they best see fit. Agile frameworks work alongside cultural values, but if you don’t have the cultural default, then what you do could turn out flawed from the get go.

Call it “faux,” “dark,” or “cargo cult,” these agile subversions often lead to situations that fly in the face of the Manifesto’s intentions—micromanagement, burnout-rate pacing, lack of delivery, and adherence to process over principles register as the most egregious—even if their practitioners come with a certificate. Unfortunately, these dark agile experiences cause some people to swear off agile altogether, (or [rewrite](#) it to reflect their real-world experience with it.)

Ron Jeffries, one of the Snowbird 17, has attempted to address these aberrations with this [qualification](#):

“Here and in other writings, I use the quoted word ‘Agile’ to refer to the many instances, approaches, and processes that use the word ‘agile’ to describe themselves, but that do not necessarily adhere to the letter or spirit of Agile Software Development we wrote about in the Agile Manifesto. I will sometimes refer to ‘Faux Agile’ for emphasis, or to ‘Dark Agile’, which I use to describe so-called ‘Agile’ approaches that have really gone bad. I might refer to ‘Manifesto Agile’ to mean the core ideas from the Manifesto, in which I still believe.”

But given agile’s wide, and sometimes misguided, adoption, is the Agile Manifesto still a document worth referencing?

Is the Manifesto still relevant?

After speaking with hundreds of Atlassian customers, internal and external Agile Coaches, enthusiasts and avid practitioners, not to mention the inordinate amount of time we spend reading about it on social media, I can say with confidence that the answer is: Yes. The Manifesto is still relevant—perhaps even more now than ever.

My colleagues Dan Radigan, Senior Enterprise Agile Coach, and Ian Buchanan, who are out working with customers everyday, both confirmed that they highlight the Manifesto with new customers on a regular basis.

Tanner Wortham, Agile Coach and Senior Technical Program Manager at LinkedIn, says he often quotes it as well. Wortham, who spent 10 years in the Marines, said he started practicing agile even before he knew there was a name for it. To him, it was just called “leading Marines.” But, to Wortham, putting a name on something is a big first step to tackling it.

“Until you can name something, you really don’t know what to do about it. I think that’s what the Manifesto did. It gave it a name. And they called it agile. I think it was something that was already occurring. But when they named it, they could start more easily identifying it.”

Scrum.org CEO West notes that agile principles aren’t really new at all. They’re just being applied in a different way.

“When I look at the tenets behind the manifesto, these aren’t tenets that were invented by us,” West said. “They’re the tenets of the scientific method. Galileo used them. Archimedes used them.”



The tenets behind the #agilemanifesto weren't invented by us, they're the tenets of the scientific method. Galileo used them. Archimedes used them.

Perhaps the Agile Manifesto’s biggest accomplishment is codifying a way of thinking that hadn’t yet been used for software development, which is certainly no small feat.

What does it all mean?

So, the agile principles existed before the Agile Manifesto. People applied them to software development. Those values were captured in the Agile Manifesto. Then, people took the Manifesto’s principles and started applying them to their own work. With all the idea recycling, is it time for an Agile Manifesto update?

Not necessarily.

When something as culturally important as the Manifesto comes along, you might be able to reinterpret it, but there’s nothing quite like the original. So, instead of trying to officially update it, maybe it’s best to figure out how to apply it to yourself, your team, or your organization.

“In a lot of ways, the Manifesto is a basis of a conversation,” Wortham said. “Here’s how I interpret it. How do you interpret it? All right, let’s figure out how to work together.”

In that vein, perhaps what’s important isn’t one blessed document that everyone can agree on, but whether or not a group of people (from a team to an entire organization) can apply the ideas in the Manifesto to their specific situation

without losing sight of its spirit. And if we can do that well, the possibilities are unlimited.

"I think if we do it right, the world is in a position to be amazing. We can solve cancer. My children will probably live to be 150, to 175," West said. "I think that we can do it, and I think that we will."

Special thanks to Amanda O'Callaghan, Ian Buchanan, Dan Radigan, David West, and Tanner Wortham for sharing your insights and expertise for this piece.



CLAIRE DRUMOND

Claire Drumond is a marketing strategist, speaker, and writer for Atlassian. She is the author of numerous articles published on the Trello and Atlassian blogs and is a regular contributor to various publications on Medium including Hacker Noon, Art+Marketing, and PoetsUnlimited. She speaks at tech conferences around the world about agile, breaking down silos, and building empathy.



TUTORIAL

Learn scrum with Jira Software

A step-by-step guide on how to drive a scrum project, prioritize and organize your backlog into sprints, run the scrum ceremonies and more, all in Jira.

[Read this tutorial →](#)



UP NEXT

How to Run an Agile Retrospective Meeting with Examples

Instructions and variations on the classic team retrospective. Popular with agile developers, but not just for software teams anymore.

[Read this article →](#)

Agile Topics

Agile project management
Scrum
Kanban
Design

Software development
Product management
Teams
Agile at scale
DevOps

Sign up for more agile articles and tutorials.

Email

[Subscribe](#)



Up Next
[Scrum →](#)