

Technology Adoption Lifecycle



The Technology Adoption Lifecycle was developed by Joe M. Bohlen, George M. Beal and Everett M. Rogers at Iowa State University built on earlier research conducted there by Neal C. Gross and Bryce Ryan.

image: https://thebullrun.files.wordpress.com/2012/11/talbellcurve_single1.jpg

6 Things Everyone Needs to Know About Agile



Jason Little | [Follow](#)



235



36



71

You could argue Agile has crossed the chasm, and you could also argue that Agile is dead. I suppose you could also argue that Agile used to mean something before it became big business.

We live in a world where **knowledge isn't power anymore** because Google knows **everything**, although, I see people on Linked In calling themselves experts and thought-leaders who clearly don't know how to use Google.

Anywho, let's get to it. Odds are you are being thrown into Agile in one way or another, and there are **some basics you absolutely need to know about Agile**. Stick around to the end if you're hiring an agile coach, scrum master, or kanban flowmaster for what to look for in someone whom you're interviewing.

1) **The Agile Manifesto**: <http://agilemanifesto.org> Go read it. And [read the history](#) of why, and how, it came to be. Oh, and [don't skip the principles](#). Then listen to [Agile Uprising's](#)

Scrum in 1993/1995 (depending on whom you want to believe) and they don't hate each other anymore so The Scrum Guide has been consolidated after they split their efforts between Scrum Alliance and Scrum.org

3) **Kanban**: Read [Kanban: Successful Evolutionary Change for Your Business](#) by David Anderson. He was the first to write about using Kanban in software. While you're at it, read [Personal Kanban](#) by Jim Benson, the arguable creator of Kanban.

4) **Extreme Programming**: Read [Extreme Programming Explained by Kent Beck](#), then read this [XP primer](#), then watch Kent Beck's '[20 Years Later](#)' talk on XP. While you're at it, read about the [C3 project](#).

5) **"Agile Testing"** - Read Brian Marick's [original post here](#), and Lisa Crispin's [awesome refresher here](#). I put 'Agile Testing' in quotes because testing is a team sport. Don't use this as a way to make your testers agiler, that's just dumb. Oh, and Grammarly said to use 'agiler' instead of 'more agile'.

6) Read [everything by Jerry Weinberg](#). Well, that might take a lifetime, so [start with these](#). Just about everyone who's considered to be an '*agile thinker*' (whatever that means) has been influenced by Jerry in one way or another.

Honourable Mention: Look up the [Manifesto authors](#) in your spare time.

Why Write This?

More and more people are getting introduced to Agile in big corporate settings via big frameworks and have no idea what the [history of Agile](#) is. **The 6 things on this list are foundational in nature, table-stakes if you will, and if you haven't heard of them, you sure as hell shouldn't be coaching a team or organization on Agile.**

There's certainly nothing wrong with the glut of methods/tools/processes/frameworks that have been invented since 2001, **but those are not the places to look first.**

Most of us old dogs moved on from Agile years ago, but now that the rest of the business world is catching up, we've had to [make Agile modern](#), [take Agile back](#), and [Rise Up](#) against the evil corporations who seek only to monetize Agile, or use Agile to beat their teams into submission.

Yes, there's a bit of snark sprinkled on this post, but for crying out loud, if you're passing yourself off as an agile consultant, agile coach, or agile thought leader - (*I've seen those who proclaim themselves thought-leaders write that there's no origin to XP? WTF??*) - **at least know the basics because these 6 things spawned the movement toward more humane ways of working**, everything else (yes, my stuff too) is just noise designed to sell you workshops, books, useless certifications, and consulting services.

Hiring Agile People?

Thanks for waiting until the end. If you're interviewing an agile coach, scrum master, or certified agile thought leader for the enterprise, if they haven't heard of all 6 of these things, **throw them out immediately**. Ok, if they haven't heard of ANY of these, or can't cite why, and how, they came to be, THEN throw them out.

role?

Start with these 6 things. Read them before going into your interview. Ignore all the dimensions of [ACI's framework](#), except for the practitioner one...for now. There's plenty of time to learn about the others but get the basics right first.

By the way, there's no such thing as a Kanban Flowmaster, but I've seen enterprise organizations create official roles like that because Scrum has Scrum Masters, therefore, Kanban needs to have an equivalent.

Anyone who's learned anything starts with some foundational learning, whether it's identifying the notes on the piano, basic scales, and chords, or the basics of economics for their MBA, or Ohms law for electrical technicians.

Agile is no different. It was created for a reason.

Published By



Jason Little

[Follow](#)

If you don't know the fundamental history of how agile came to be, you're missing out.

36 comments

[Sign in](#) to leave your comment

Daniel Mezick

Founding Member and Advisory Board Member at Open Leadership Network

1y

Good job Jason. Great essay. Keep going man. Don't stop. And then there's this, previously written and (big surprise) 100% aligned with your great essay here: <http://newtechusa.net/get-it-in-writing/> "This problem plays out every single day in nearly every single Scrum implementation. Scrum is implemented; but they never agreed. BIG PROBLEM. "None of the people involved actually examined the Scrum Guide and accordingly, NONE OF THEM AGREED TO PLAY. Because not only did they NOT read the Scrum Guide, but even worse, no one asked them to AGREE to play by the rules. "In that scenario, the Scrum Master cannot (repeat) cannot be effective. "And the Scrum is going to be a total disaster." So: The #Agile money is very green. Is that a valid reason to be party to really crappy & totally ineffective Agile?

Like Reply 1 Reply

Jason Little

1y