

BUILDING SOFTWARE USING “AGILE” AND “SCRUM” HAS TURNED TO BE A WASTE OF TIME AND MONEY FOR ALL CONCERNED!

[AGILE/SCRUM is literally the homeopathy of the developers community \(programming\)](#)

submitted ago by [roznak](#)

AGILE/SCRUM is literally the homeopathy of the developers community. It is PLACEBO!

Everyone is copying from everyone else because if the other 10 companies does it, then it must be good.

It is nothing more than: "10 steps successful millionaires do to become successful. Now buy this course for only \$1000"

The only people getting wealthy rich are the course teachers. Or in this case the SCRUM teachers that suck lots of money from courses for something that simply works as PLACEBO.

How can you test this?

- Take one team that works in SCRUM mode.
- Take one **good** developer and let him work alone.

Give both teams the exact same development job.

That one **good** developer guy will tap-dance around your SCRUM team and still have time to spare. And the bonus part is that the one developer will even have code that even a guy from school can understand because it is not abstracted to almost useless to understand code.

It is shocking to see how **good** developers are wasting productivity time on this mindless AGILE/SCRUM methodology that is push like a religion.

- [13 comments](#)

want to join the discussion? [login](#) or [register](#) in seconds.

sort by:

[New](#) [Bottom](#) [Intensity](#) [Old](#)

Sort: Top

[\[-\]](#) [aristotle07](#) 4 points (+4|-0) ago

I largely agree. Agile has been the bane of my existence. The problem with Agile is that there are so many things that look good on paper which is why they sell so much of it. But as a whole, my experience is that it stretches development time by 2-4 times the duration and increases cost because now you have a product manager, a scrum master, a scrum coach and the client advocate who does the work of what used to be one experienced PM.

A team is only good as its developers. Give me 1-2 good developers and I can deliver under budget and on time under any process even agile. Have a traditional makeup of 20% good and 80% bad, the developer is so busy doing stupid agile processes like pair programming, and showcase setup so each sprint you only get 30% productivity and everything goes into setting up for the stupid showcase. Just let 1-2 developers work with a full understanding of the requirements and allow them to speak directly to client for constant feedback and you have a successful project, every time.

I hate that inception and user stories drive the requirements and most people enforcing it doesn't even understand what's needed for a good developer to do good work.

- [permalink](#)

[\[-\] hotairmakespopcorn](#) 5 points (+5|-0) ago

Formal development processes exist BECAUSE so many developers are poor developers. For good developers, most formal processes only act as a hindrance. This is exactly why you're observing what you observe. As is the traditional 80/20 rule, 20% of your developers do 80% of the work. Which leaves us with what to do with the remaining 80% of the team. Yes, they can contribute, but they require some type of process to follow otherwise they'll get lost in the weeds every time. Sapping even more time away from the 20%. Management sees this as good because you get better utilization of the 80%. It may well result in a net win, if implemented correctly. But, of course, that's not how the 20% want to spend their time - mostly.

Agile is really about neutering the 20% to make the 80% appear and/or be more productive. It's a bribe to force the 20% to drag the 80% into the solution domain. For body count houses, this is great stuff. For good developers, it's a nightmare.

Let's face it, good PMs are harder to find than good developers. And good developers are as rare as unicorns. The result is that most projects get stuck in suckage-land, whereby the 20% are stuck fixing the issues caused by leadership, management, the PM, and the rest of the team. Which is usually compounded by the fact that everyone wants to work reactively rather than pro-actively. After all, who can possibly foresee that if you implement your solution with highly specialized patterns and that if the requirements change at all, it's going to mandate massive refactoring. But, hey, there's a reactive process for that, so it's all good.

Excuse me, I have to go finishing butting my head into the wall for another couple of hours.

- [permalink](#)
- [parent](#)

[\[-\] LittleMark](#) 1 points (+1|-0) ago

Agile is really about neutering the 20% to make the 80% appear and/or be more productive.

So, essentially it's Common Core all over again?

- [permalink](#)
- [parent](#)

[\[-\] oddjob](#) 3 points (+3|-0) ago

Just let 1-2 developers work with a full understanding of the requirements and **allow them to speak directly to client** for constant feedback and you have a successful project, every time

But, muh project scope. I can't trust developers to talk to normal people. /s

- [permalink](#)
- [parent](#)

[\[-\] roznak](#) [\[S\]](#) 2 points (+2|-0) ago

Pair programming that is another thing! Back in 2000 they also talked about pair programming. I tried it and it fails. It always makes the guy that is less developed feel dumb and ends up in depression. It causes stress in these developers which is a shame because some of them are very good at what they are doing, just less experienced.

- [permalink](#)
- [parent](#)

[\[-\] FuckFatReddit](#) 2 points (+2|-0) ago

I think you can have a bit of best of both worlds. Where I work we take the good parts of SCRUM and have left most of the shite behind. Bearing in mind we're only 3 full timers and a half timer so that leaves us more flexible.

Having said that I had a deadline coming up my ass and lost time getting my presentation ready!!!!

- [permalink](#)

[\[-\] BenKekJr](#) 1 points (+1|-0) ago

I absolutely agree, and I believe leftists / liberals love it. Because they need a master, without a master they feel like they have no direction. Others hide behind the secret sauce, they are bad/poor at their jobs. In the same way they say... I WANT THE GOVERNMENT TO SOLVE ALL MY PROBLEMS! They want Scrum/Agile to be the authority that solves all their problems.

It's so funny, accompanying AGILE/SCRUM at the company I work for is this "us" talk, where they treat Developers, QA, Devops/Admins like they are all equal. I'm a Developer for a reason, I want to write code. I'm not here to administrate your servers, or debug your dns resolution issue, or backup

your mail server. Now if I was at a startup, I get it, you do what the business needs you to do. But at a corporation, the Irony of saying "We are all equal, the team is all equal" is so ironic because it only works one way, no one ever writes code for me, and yet I find myself do everyone else's job. It's a system of cultural marxism.

Recently I pointed out that a particular engineer needed to do something (we'll call him Rob), literally because he had permissions / credentials.

My manager interrupted me and said... well it's not just Rob's job, "it's the teams job". The snowflake inclusion culture has hit software development / scrum, and all the developers have to move as slowly as the least competent one. I literally think Scrum was invented to destroy American productivity.

Snake Oil 100%.

Was told by a "scrum master" - if anything takes longer than 2 weeks it will never get done. Well what was Christopher Columbus thinking.

Scrum exists to deter from any great invention, to stifle the genius of any single individual, and to only apply obvious, mundane, and easy solutions to all problems.

- [permalink](#)

[\[-\] Gargilius](#) 1 points (+1|-0) ago

The whole thing is about micromanagement, that's what it boils down to; might work great to herd newbies and less experienced developers, but it is indeed a nightmare situation for experienced devs...

I have found this [blog entry from Michael O. Church](#) to be particularly interesting on the subject.

- [permalink](#)

[\[-\] BenKekJr](#) 1 points (+1|-0) ago

The **Violent Transparency** part of that post, is absolutely on target. What a concise way to describe it.

It's well known that creative people lose their creativity if asked to explain themselves while they are working. It's the same with software.

- [permalink](#)
- [parent](#)

[\[-\] ForgotMyName](#) 0 points (+0|-0) ago

A lot of what you're talking about has nothing to do with SCRUM or Agile. E.g. - Whether the code that a team creates is understandable and maintainable. If you don't understand why abstraction is valuable I question your understanding of development in general. I'm not saying this to belittle you but to inform others that are reading what you're writing and taking it as gospel.

SCRUM/Agile are just tools. Like any tool, they can be misused by morons and you'll end up with poor results. I've worked on teams small and large on SCRUM and waterfall projects alike and I have seen SCRUM be very successful when properly implemented.

As to your comparison contest - One person will always out-code a team if that team is composed of poor developers. Even if there are 50% rock stars and 50% idiots the team will still lose. The good devs will spend all of their time cleaning up after the bad devs. This is why some companies are extremely aggressive about hiring "the best". Bad developers aren't just slow, they are cancer. They actively hinder the rest of the team's progress.

TL;DR - If your life is crap it's because you are stuck with crap management and crap devs. Find a new place to work and pray that things are better there (spoiler: they probably won't be).

- [permalink](#)

[\[-\] AlwaysInService](#) 0 points (+0|-0) ago

I've never even heard of AGILE/SCRUM. What is that?

- [permalink](#)

[\[-\] DearestTrudeau](#) 0 points (+1|-1) ago

Agile is an attempt to turn Pajeets and similar non performers into interchangeable cogs. It has the advantage of reducing the impact of developers that actually know their shit, and forcing them into the peg hole that management has prepared. Lose-lose.

- [permalink](#)

[\[-\] TheDude2](#) 0 points (+0|-0) ago

If we could dump the shitty debts then we wouldn't need it.

- [permalink](#)