Twitter Thread by Andrew Harmel-Law ■

Andrew Harmel-Law ■
@al94781

Software architecture is in crisis, and the way to fix it is a hefty dose of anarchy.

Some lay the blame for this on @boicy with the whole microservices thing.

(Admittedly, @nicolefv, @jezhumble and @realgenekim didn't help when they statistically proved that he might have been onto something with all that de-coupling and team-alignment…)

However I don't blame him at all.

I think he saved us; bringing us back to the path of value-delivery and independent services, but now with added independent teams.

But one thing is clear. Microservices need more architecture, not less (as do other forms of #Accelerate-style software organisation).

(See https://t.co/B2hWmXhIqe if you need convincing)

I mean, all those pesky slices we need to carve up our monoliths (or were they big balls of mud?) That's a significant amount of work right there…

The truth is, the vast majority of attempts at - and ensuing aftermaths of - such slicing and continuous delivery have highlighted problems for almost all organisations, even the ones who were doing great architecture: How to scale their architects.

One of the biggest problems? Suddenly architects (and I include myself in this group) needed to be in far too many places at once, doing all that "architecture".

And so to cope, we architects either kept doing our job and became bottlenecks, or admitted defeat / got circumnavigated, or worse still went back to code and created “frameworks” which helped teams stick to the true path.

Shudder

None of these approaches have worked. And they never will.

And consequently many, many #microservices adoptions failed; with #microservices themselves getting an undeserved bad name in the process.

But microservices aren’t a curse on software delivery - and they ought to be a blessing.

What we need is a workable way to approach them, and in the process realise the associated benefits of both team autonomy and improvements in system architecture.

In the remainder of this thread I’m going to introduce the idea of an #AnarchisticArchitecture.

I’ll describe what it is and you could do it. Hopefully you’ll see how it offers the best (only?) way out of this mess.

Let’s remind ourselves first what “#anarchy” means: It’s the absence of government. The absence of authority. https://t.co/iJAZQJIXgg

Straight away that means how we are used to doing architecture, via all-powerful architects taking all the decisions, is going to have to stop.

> \u201cYou may think in describing anarchism as a theory of organisation I am propounding a deliberate paradox: \u2018anarchy\u2019 you may consider to be, by definition, the _opposite_ of organisation. In fact, however, \u2018anarchy\u2019 means the absence of government, the absence of _authority_. ..."
>
> — Andrew Harmel-Law \U0001f3e1 (@al94781) November 29, 2020

But decisions still need to get made - that’s what architecture is.

(As @Grady_Booch has said “architecture represents the set of significant design decisions that shape the form and the function of a system, where significant is measured by cost of change.”)

@martinfowler agrees “software architecture is those decisions which are both important and hard to change”.

So the first test is can an #AnarchisticArchitecture deliver on this?

The answer is "Yes".