Sid Phillips

1/20/2022

Takeaways from the Gartner Report:

I find it very Interesting how much the report is upfront about the fact that microservice architecture is not built for everyone. Microservices, even when used, should be applied incrementally and selectively. However, in an environment that demands continuous delivery (CD), it’s easy to see why MSA is so appealing to many companies. CD in a monolithic environment can be dangerous, and every programmer knows why. New code can and will break old code. In industries where 100% uptime is essential to maximizing your profit, a tiny programming error could cost thousands of dollars. Once you’re able to identify that the system is down, it can still be very difficult to identify where the problem even occurred. This is more and more true the larger the systems get. The way that MSA helps to solve this problem is by having individual components in a system work independently of one-another, minimizing dependencies and maximizing agility.

Again, these benefits can seem incredibly appealing at first glance. However, the report also estimates that by 2022, 70% of organizations that attempt to switch to microservices will have found the effort too disruptive and will have switched to miniservices instead. I found this shocking. “How could so many businesses have decided to cut their losses with an architecture that provides so many benefits?” There seem to be three main reasons for this. The first is that while the architecture provides great scalability, it comes with an incredibly high upfront cost that smaller businesses are unable to cover. To switch to this model means that you’re radically changing up the way your workplace operates, which costs time and money. Employees who were used to the monolithic development cycle will need to take time to learn their new roles. If your teams consisted of only a few employees, very few people would be looking over and reviewing code that might need to go live before a tight deadline. Next issue I see is that it’s just not necessary for many businesses. Netflix, Twitter, and Amazon consume gigantic amounts of internet traffic matched only by two or three other companies in the world. While Twitter obviously has a need to expand specific parts of their platform, most companies don’t have this kind of insane traffic or need to expand parts of their service so quickly, and thus deem it unnecessary. Lastly, and perhaps most importantly, microservices increase complexity by a lot. Each individual application must be encapsulated and rely on outside dependencies as little as possible. This again eats into development time, which many smaller companies may be unable to afford.

It’s also interesting to think about the ways in which we have to be careful when talking about microservices. The report has a whole section listing off the things that microservices are not, and I’d bet that this has a lot to do with the “microservice-washing” that companies try to sell potential investors on. Perhaps the reason that so much confusion exists around the topic is because certain companies want that confusion.

The concept of miniservices is of course the most compelling of the microservice alternatives. Although it seems to be an implementation of MSA with much more relaxed constraints, this would be the obvious solution to many of the problems that come with the MSA. It’s a middle-ground of sorts, without so many of the complexity issues and cultural mindset problems from the MSA, but also without the complete independent working parts. If I were leading a smaller business, this approach would seem very appealing to me.

All in all, MSA is a radical shift in the way applications can be developed. It comes with upsides and downsides, but may well be the way many of us program for much of our adult lives, and thus is very important to get a grasp on now.