

The rise of knowledge workers

- In the past, a lot of times, you could learn everything you needed to know by moving up the corporate ladder. Today, there are many ways that modern product delivery makes that much difficult. So it's important to think of the agile mindset not just as a new thing but as a reaction to difficult challenges with typical product delivery. In the 1990s, many project managers recognized that they needed a way to deliver products in a new way. What they found is that many of the people working in the organization had changed. **These different employees started to specialize** and it was difficult to find a generalist who understood all the different specialties. So a database engineer might know a little bit about software development but not enough to manage a team of developers. This led to a serious challenge in management. It put project managers in a difficult position. They were responsible for delivering the product but had very little knowledge on how to make key technical decisions. Software products had also become much more complicated, so it forced many more people to specialize. It was harder to just be a computer scientist who built an entire product. Instead, now you had software developers, database engineers, testers, and even infrastructure specialists. Think of it this way. I grew up outside of Detroit and when I was a kid, most of what you needed to work in a factory was a lunchpail and a desire to learn. So you might start out as a line worker twisting bolts, then over time you might get promoted to a line manager. Finally, you might get a job in the office and manage hundreds of employees. By the time you worked in the office, you knew most everything there was to know about building a product. You probably started building them yourself and then worked your way up into management. The managers then had the knowledge of the workers they managed. Now, compare that to a modern software product manager. Here, you might manage a team with many different specialists, each person probably spent decades building up their own expertise. There is no way that one project manager could know all the different decisions that it takes to build a software product. They couldn't be a database engineer, software developer, and infrastructure specialist. Even if they did, they'd never be able to keep up with the changing technology. That's because there is a big difference between managing workers in a factory and managing knowledge workers developing software products. Knowledge workers are employed for what they know. That makes it very difficult to manage them with a traditional hierarchy. You can't direct a team of people who collectively know more about the product than anyone else in management. So you have managers becoming less directive and more supportive. One of the key things that you see with agile teams is this push to be self organized. That's because it's very difficult to manage the team with a top-down approach. So you shouldn't have one manager who's directing the team to deliver the product. There is no line manager like you have in a factory. **Instead, you focus on having**

collective decision making. This is a very difficult change for most organizations. Many managers like to tell their employees what needs to be done. And then they go off and get it done. They might not be ready for a group of specialized knowledge workers coming up with their own plan on how to deliver the product. Later, you'll see the importance of this servant leader style management with a role called the scrum master. But this role is just part of a larger trend in self organization. A lot of the agile mindset focuses on giving these knowledge workers the flexibility they need to succeed.