

Scrum in larger organizations

We had the opportunity to have a chat with Jeff Sutherland when he was in Sweden during the Øresund Agile conference. We discussed the implementation of Scrum in larger organizations, and his latest experiences of using Scrum in an industry where off-shore outsourcing is a major trend. Jeff had many interesting things to say, and we have split the discussion in two parts, of which this is part I – Scrum in larger organizations.

When did you first try Scrum for larger projects?

In 1996 I was hired by IDX-systems as their VP of product development, and eventually their CTO. I started at that company with about 300 developers, and by year 2000 we had almost 600. So we started immediately by converting the 300 developers that were in the company – all of them – to Scrum in 1996.

And how were the Scrum teams organized in this company?

IDX had several business units with different product sets, and all the products had to work together in large enterprises. The larger product units had about 100-150 developers, and the smaller ones were 30-60 developers.

One of the interesting things I am finding in companies now – is that about 60 developers tend to be a product unit. If the product is bigger it can typically be broken down into pieces that work well, with about 30-60 developers in each development team. And that can be very nicely be broken down into a set of Scrums that is run by a Scrum of Scrums which consists typically of the leaders of every Scrum team.

And, according to your experience, how many Scrums would there typically be in a product unit with 60 developers?

If you divide that by seven you get a little bit less than nine, so it would probably be seven or eight Scrums.

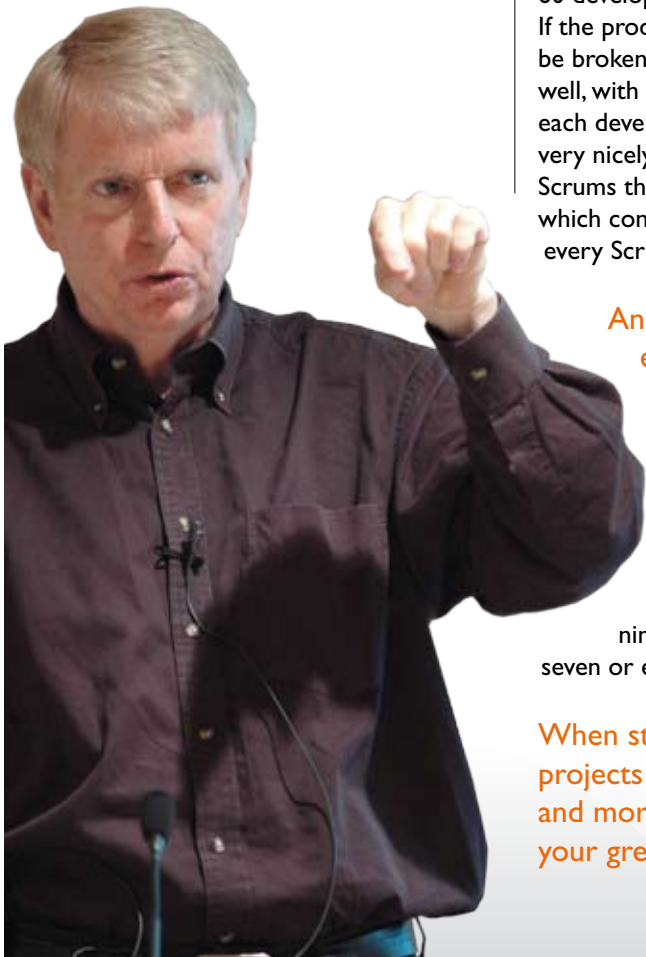
When starting with these larger projects with Scrums of several and more Scrums – which were your greatest challenges?

We really did not have any significant problem in extending it into multiple teams. The problems we had were the typical problems when starting up a single Scrum – having people to understand how it works and following the process. Having the daily meetings and get the reports out, so the whole team could see the state of the project every day.

In retrospect at IDX we had very detailed data provided by external consultants on the productivity of every team.

Scrum was designed to produce teams that get five to ten times of industry average.

I engaged Capers Jones, who is one of the leading productivity analysts in the United States. I had his company to come in and do a function point analysis every release for all products. So we knew how many function points there were added incrementally by every team. So we got a clear velocity that was completely independent of any team, and was comparable across teams. IDX on the average increased their productivity with a factor of about 2.4 using Scrum – 240 percent. But Scrum was designed to produce teams that get five to ten times of industry average. I was always interested in finding a few teams – which we did have – that hit that hyperproductive state. And I was always trying to figure out why the Scrums that only doubled productivity – why they couldn't do more. At IDX in retrospect, I had a lot of difficulty getting them to break the teams down small enough. They had



a culture that historically had had thirty person teams – and I managed to get these teams down to fifteen or less, but I could not get them all consistently down to seven. And industry data show that seven people can finish the same project earlier than eleven people – so that I think significantly impacted productivity. You know, the goal of Scrum is to achieve Toyota-level performance, that is four times the level of your competitors in terms of throughput, and twelve times the quality. And the fact that we only doubled the productivity and could not get four times I believe was directly related to the team size.

So, one of the problems when implementing Scrum in larger organizations is obviously that it is hard to get the size of the teams small enough. Are there other challenges that an organization should look out for when changing to Scrum do you think?

Well, in order for Scrum to succeed, people need to shift their mindset to an Agile mindset. And, with Scrum, there's a continuous quality improvement process, that is designed to on a daily basis surface problems, or impediments. And a primary role of the Scrum Master who is leading the Scrum team is to prioritize and resolve those impediments. Some of these impediments are a result of other parts of the company disrupting the team. So, one of the jobs of the Scrum Master is to keep the rest of the company from disrupting the team and getting management support to resolve the impediments. And if the entire management team does not know what Scrum is all about, and why we do things the way we do it is hard to get them to help. If the management continually stays in an old mindset, they will not achieve the level of performance that they need, and in the worst cases it makes the Scrum implementation impossible.

So another great risk is that management stays in an old mindset?

Yes, and we say that the Scrum Master has to continually contend with several key issues. One is what we call the “waterfall mentality” which is a linear planning process so that the management thinks that you can create a Gantt-chart that plans out the project for the next three years, and you'll systematically follow that Gantt-chart step-by-step. But what we know, in all companies, is that on the average half of the requirements change during the development process, and many unpredictable things come up. So that the dependencies that were outlined in that initial project plan turned out to be wrong almost within the first month. The beauty with the Agile approach is that you adapt to that. But if management holds this linear plan – “You must do this because you said you were going to do exactly this six months ago” – they will destroy the implementation.

“...so that everybody knows were the project is all the time, and exactly what the problems are.”

A second thing that is really important in Scrum is that one of the goals is to make all information available to everybody, so that everybody knows were the project is all the time, and exactly what the problems are.

And there are always people who will be impacted by this free availability of information. Particularly if they are trying to protect their own organization and trying to sub-optimize the corporation by building their internal empire. As soon as the information surfaces that reveals that that is causing problems with global throughput, that organization – and particularly the management of that organization – can often try to crush the Scrum-implementation. And even if they don't try to crush it they are not helpful. They try to keep information secret. They try to avoid any comparison that would make their less productive teams look bad and they are manoeuvring against a good implementation of an agile process.

You have typically these problems with an old mindset versus the new. Toyota has found out the same thing – they have a consulting group, and

they've shown that they can come into any company and introduce lean practices, and virtually double any parameter that is related to productivity or quality in six months. They come in and they work for six months and then they leave, and those productivity metrics degrade – because the mindset tends to revert back to the old mindset. That is another example of the problems of implementing agile or lean processes in a company. If it's a small team this is not such a big problem, but if it is the whole company or a large set of teams it is a much bigger cultural change that you have to work with.

So, to conclude, two of the major challenges you will have to deal with when you implement Scrum in a large organization is (1) to change the mindset in the organization in general and on management-level in particular, and (2) to make the size of the Scrum Teams small enough, preferably 7-9 people in each team.

Yes, that's about it.

Jeff Sutherland is the inventor of the Scrum development process, which was first used at Easel Corporation in the early 90's. He has been VP of Engineering and/or CTO for nine software product companies, developing Scrum in four of them and introducing today's standard Scrum methodology to five of them.

Visit his blog at: <http://jeffsutherland.com>



Jeff Sutherland and Gustav Bergman, Softhouse, at Jens Østergaard's place at Nyteboda, Sweden.