

Descaling your Scrum

Every now and then I must remove scale from my kettle. It accumulates with use and slows the heating of the water over time. It may change the flavour of the water a bit but doesn't otherwise make a whole lot of difference other than to create lime as a byproduct of making tea.

Business that scale their organisations are a lot like that. A successful product accumulates more and more staff in times when there is enough money to go around. Too many firms are pushed into descaling when hard times come. Angry Birds' Rovio laid off 16% of its staff in October 2014, leaning down to 600 employees. It's hard to not compare them to rival Supercell, which was realising five times as much revenue with only 132 staff.

"Scaling agile" is now all the rage, perhaps because the large late-adaptor companies are now coming on the Scrum scene, bringing a misconception that they need to paint Scrum onto their thousands of engineers. Such large companies rarely stand still but are ever looking to grow. And, of course, a new topic like "scaling" is a newly drilled well of consulting, training and, yes, even certification revenues. Why do companies scale up instead of lean down? Because they're afraid that the competition may outperform them. In the contemporary software market there is an unspoken principle that more features is better. Society has so much given up on quality that few consumers even bother to look at vendor's lists of known bugs in new releases. Society instead values excess.

Scaling the work force is capitalism's way to achieve socialism's goals. If an enterprise realises profits far beyond what is necessary to feed the mouths of its employees, one can always share the wealth by hiring more. Many of them end taking non-value-adding roles in the production process. Neil Harrison and I call these "deadbeat roles" in our *Organisational Patterns* book. Our research found that these roles dominated many large organisations. At least one of my Japanese clients calculates management bonuses in part on the basis of the number of people reporting to them. That is the antithesis of lean.

David Graeber, professor of anthropology at the London School of Economics, also infers that the *majority* of jobs in a modern economy are rubber-stamp jobs. He claims that they don't add real value, but just keep people employed in an economy that has become so efficient as to not require "full-time" workers. See "On The Phenomenon of Bullshit Jobs," *Strike Magazine*, 17 August 2013, <http://strikemag.org/bullshit-jobs/>.

Scaling staff makes sense for endeavours where work is perfectly partitionable. If 10 slaves can transport one stone for the pyramid, then 100 slaves can transport 10 stones. Coordinated work has a sweet spot beyond which adding more people adds more coordination cost than is repaid in horsepower. 20 people can't sort a deck of cards faster than 5 people can. Adding people to work on one system means more teams, and that means more handoffs between teams. Intellectual work in complex domains requires coordination that lowers the sweet spot — and the sweet spot is very low in software development.

Small is beautiful. Most organisations start small and grow to feed more mouths as they give into social pressures that believe that there is power and success in growth and scale — like my Japanese client. If you start small, stay small. You should be very afraid if a consultant offers to help you scale your teams.

I'll continue these thoughts in the next column.

DRAFT