Approaches how to scale agility

9 February 2014 | **Software Engineering** | [**Rainer Grau**](#)
Reading time: 7 minutes

Dance with the system
In his book Management 3.0 [1] Jurgen Appelo explains very clear and understandable that Software Development represents a complex system. He discusses what agile means and management approaches exist to "dance with the system" to use Jurgen's words. The three topics that especially delighted me are 1. empower teams, 2. align constraints and 3. grow structure.

I personally hope that management will understand the message behind Jurgen's book any time in the near future. The message is that the times are gone where management designs **the plan** how to compete successfully on the market and success becomes reality when all workers **strictly follow the plan**. Instead, managers have to "dance with the system". And the most important elements in a complex system of knowledge work are people.

Well, this is where the challenges arise. Theory about team building clearly states that the most productive size of a team is between 5 and 10 persons [2] working together in a context to achieve common and agreed objectives. The Scrum framework for example incorporates this as a basic principle. So, if we want to empower teams, grow structure and build up a high-power competitive organization of, let's say, 300 or 500 people trying to achieve common objectives, we need more than the very basic principles and management tools as given to us by Jurgen. More concrete blueprints on how grow structure could look like would help us. Based on these blueprints with its characteristics and constraints, we could develop ideas to grow structure in a specific context.

Different Visions how to grow Structure
This is where the crowd of agile and lean consultants arrives on the scene. To be frank: I am one of those. I personally divide this crowd of consultants into three rough categories (please do not take this too seriously):

- Visionaries: Consultants with a very strict opinion on how "grow structure" looks like. These consultants develop a specific blueprint. This blueprint represents structures and proceedings based on their experience over the years that showed success in many practical applications. They establish the "XYZ" framework representing this blueprint on how to grow structure. Smart as they are, they now market and sell the "XYZ" framework as a kind of a product. This is a good thing. It requires a lot of experience and hard work to create a consistent "XYZ" framework; to present it in a way so the value is understood by many others and to market it;

to create a community willing to pay money to license, learn and apply "XYZ"; to make customers happy with "XYZ". This is the return of many years investing into gaining experience.

- Followers: Consultants that are convinced that agile and lean is the right way to do things. But unluckily they never got the chance to establish their own "XYZ" framework. They often stick to one "XYZ" framework. The reasons they stick to one framework are manifold. Maybe they love to work with their clients; maybe they have to earn money for life, so time for learning beyond "XYZ" is restricted; maybe they are dogmatic and once they tasted how sweet "XYZ" fruit are all other fruit are sour; maybe any other reason. The point is: they follow "XYZ" that's why I call them followers. If you as a customer need exactly "XYZ" these followers are perfect as they understand "XYZ" within all aspects and often have serious experience in applying "XYZ" in a context. Nontheless – if a mixture of "XYZ" with "123" would be the better dance for your system, well you might have trouble with a follower – or two competing followers….

- Dedicated: Consultants that are convinced that agile and lean is the right way to do things, but not as an end in itself. The dedicated strive to improve organizations so they compete successfully on the market; deliver products that customer love; create work environment of trust and fun for their employees. This category of consultants never stops learning and improving. They take what is out there, challenge, combine, rearrange and "dance" with ideas, means and "XYZ/123" frameworks. The goal is to improve, lead and guide through the jungle of how to grow structure. These consultants are neither the visionaries nor followers. They are rather hunter-gatherer. I personally would assign me into this category.
Coming back to Jurgen and the grow structure thing: Visionaries develop "XYZ" frameworks how to grow structure. Followers focus on training, coaching and introducing of a specific "XYZ" framework. The dedicated investigate and try to find the ideal combination on how to grow structure in a specific context by using and adapting "XYZ/123" frameworks and methods that are on the market.

I am personally convinced that an organization that decided to transform towards agile requires all three types of consultants. The visionaries develop fundamental guidance; the followers excel in application of specific means, the dedicated care for the big picture and the optimal way to go.

Applying the agile Tool Box
In the end, to make certain: there is no silver bullet. By the way, the bullet allegory never convinced me at all. I rather like the tool allegory. There are many tools in the agile and lean tool box, smaller ones, larger ones, simple ones and complicated ones. The art of consultancy is to choose the right combination of tools. And even to adapt every single tool like a casing tong can be adapted to the diameter of a pipe. Me, as a dedicated consultant, I am eager to

know as many tools as possible and to understand how to adapt and combine tools to suite a very specific application scenario. That's what I did for quite some years in the past – together with a good team of like-minds in permanent discussions and exchange and with many challenges facing at customer site. With this I extended my tool box and learned how to use it – and I'm still learning.

The Course "Agility at corporate level"
Now the time has come to give some of my experience back to the community and discuss what is in that tool box. I engage in communities, teach at universities and attend workshops. At Zühlke we developed a public course for growing structure in agile organizations based on our collective knowledge. We named the course "Agility on Corporate Level" as the course name "grow structure" would not get the desired perception. This course discusses four basic "XYZ" frameworks, developed under the lead of four visionaries that are widely used approaches to grow structure in large organizations. These are: SAFe ([3] by Dean Leffingwell), Agility Path ([4] by Ken Schwaber), LeSS ([5] by Craig Larman) and DAD ([6] by Scott Ambler). The focus of this course is to discuss their basic principles, structures and methods. With this the attendees know characteristics of the frameworks, are enabled to draw decision for their own organization and start to grow and improve. A potential way to start could be taking some SAFe elements, enrich them by LESS and start going the Agility Path. Yes, there's no 'silver bullet'!

I would be happy to discuss with you and share what I have learned with what you learned and improve…

[1]: Jurgen Appelo; Management 3.0: Leading Agile Developers, Developing Agile Leaders; Addison-Wesley Professional; 1 edition (January 7, 2011); ISBN-13: 978-0321712479

[2]: Gerhard Comelli und Lutz von Rosenstiel; Führung durch Motivation (3rd Edition 2003); Publisher: Vahlen; ISBN 3-8006-2927-5

[3]: Agile Software Requirements: Lean Requirements Practices for Teams, Programs, and the Enterprise; Dean Leffingwell; Addison-Wesley Professional; 1 edition (January 6, 2011); ISBN-13: 978-0321635846

[4]: scrum.org website; Author: Ken Schwaber; https://www.scrum.org/Agility-Path

[5]: Craig Larman, Bas Vodde; Scaling Lean & Agile Development: Thinking and Organizational Tools for Large-Scale Scrum; Pearson Education, 2008; ISBN-13: 978-0321617149

[6]: Mark Lines, Scott W. Ambler; Disciplined Agile Delivery: A Practitioner's Guide to Agile Software Delivery in the Enterprise; IBM Press, 2012; ISBN-13: 978-0132810135