Scrum Alliance®

Scrum of Scrums: Running Agile on Large Projects

Leandro Faria Stefanini, 5 June 2013

How to use Agile practices on large projects? There is an idea that Agile only applies to small projects, perhaps for the simplest of the routines, events, and artifacts of the frameworks.

The fears are valid. We can mention several issues that would make Agile adoption impractical:

- Starting with the size of the team: According to Scrum, a team has seven people (plus or minus two). So, how do we use Scrum in a large project with a team limited to nine people?
- Not considering the restriction of the number of people, how do we make a daily stand-up meeting with a team of 100 people? Impossible! To keep to the 15-minute time-box, each person would have 9 seconds to speak. Not to mention the need for an auditorium and a Kanban board of the size of a whole wall.
- From the point of view of planning and control, things get too complicated. Because we have a large team, a lot will be implemented during a sprint. Even if we have a longer sprint, say two months, we would be spending a lot of time estimating the stories, not to mention the mess it would be to have so many people estimating stories at the same time. Also remember that the longer the sprint time-box, the higher the risk of the sprint losing its sense and its objective halfway through.
- Think also about how to measure the team performance. Yes, we can measure the average productivity of a team of 7 people, but what about a team of 100? It would be almost impossible to forecast the productivity of a team of this size, as the productive capacity and the learning curve are very different between different groups of people.
- Consider further the work of the PO. How can a single person detail, prioritize, and sanitize a product backlog so large, and be ready on time for the sprint planning meetings? Working with the backlog would be a full-time job.

Knowing the problems, now let's go to the solutions, These are, by the way, very simple.

The name of the concept is almost self-explanatory: Scrum of Scrums.

In Scrum of Scrums, we basically divide the team into 2 or more teams, respecting the limit of 9 people per team. To make this example easier to understand, let us assume that we have a project with a team of 30 people. In this case, we now have 3 teams of 10 people each.

TEAM ALPHA
TEAM BETA
TEAM GAMA

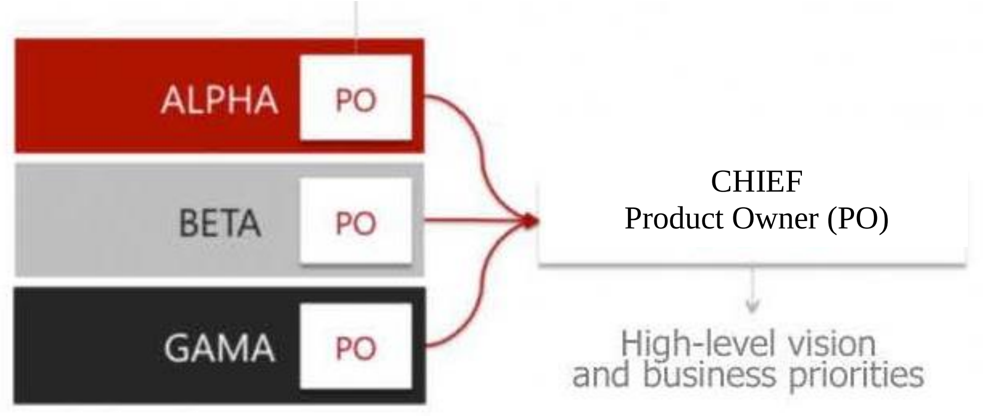
OK, dividing a team into parts is a good strategy, but if we consider that all these groups will be working on the same product, we will still certainly have a lot of problems, among them:

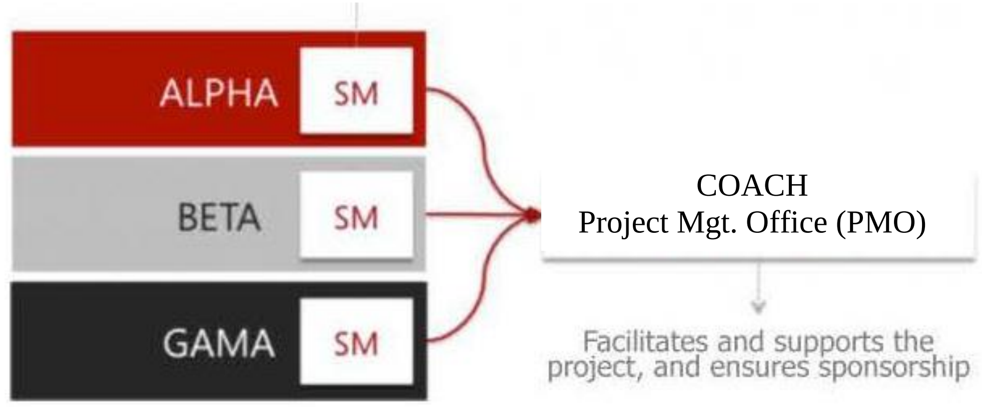
- Lack of a unified view of the product/service

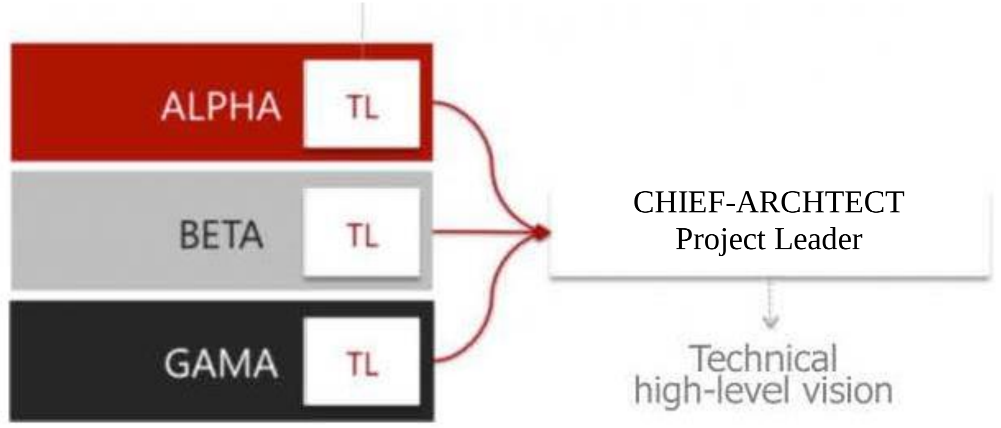
- Redundancy of work (two teams implementing the same part of the scope)
- Communication failure
- Integration hell (how to integrate different parts of a product developed by different teams)
- Dependencies between tasks of different teams
- Complex change management

We could surely list a dozen more problems. And I start by saying that there is no 100 percent-effective solution to any of them. What I will describe below are some techniques that allow us to deal with these problems, though doing so certainly demands special skills of team leadership, in particular that of communication.

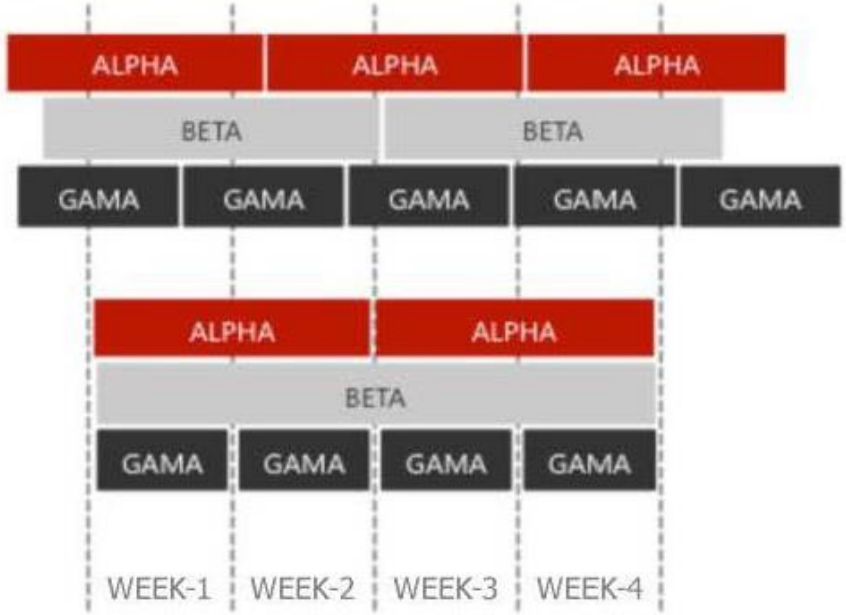
The first technique (and it's somewhat obvious) is the replication of key roles in each of the teams, such as the PO, ScrumMaster, and technical lead.



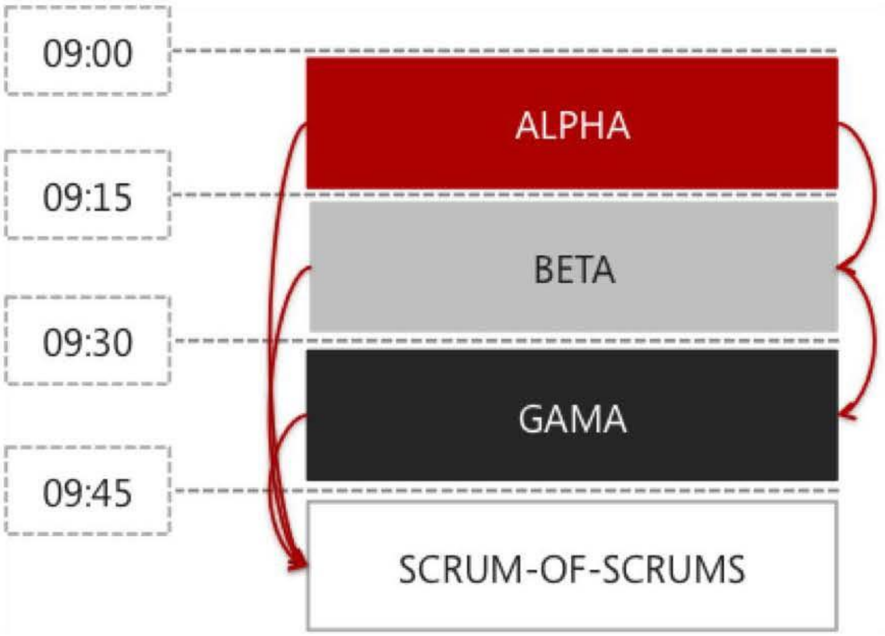

ALPHA
SM
BETA
SM
GAMA
SM
COACH
Project Mgt. Office (PMO)
Facilitates and supports the
project, and ensures sponsorship



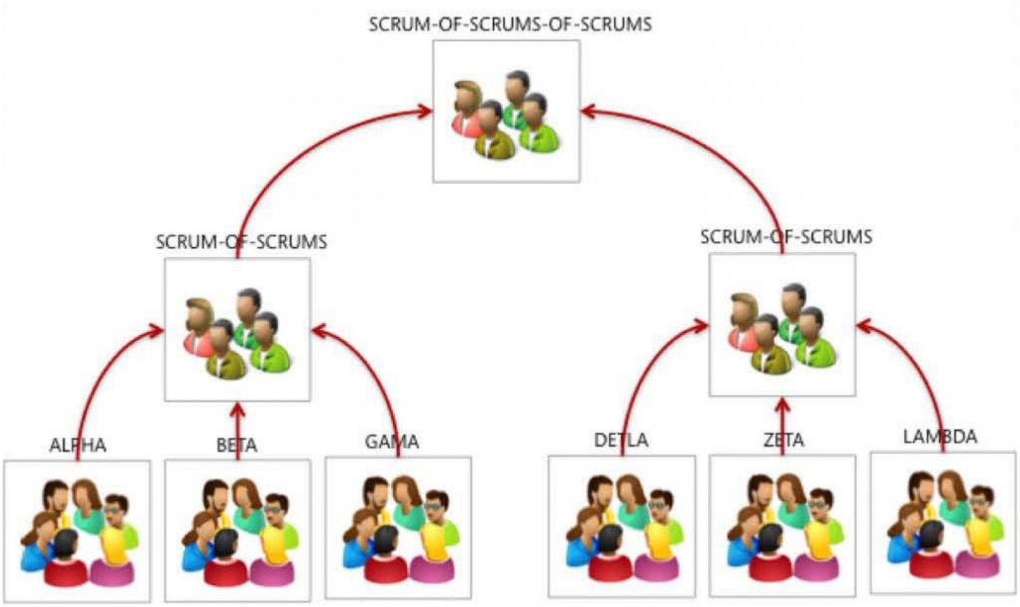
Another technique that makes a lot of sense is the alignment of iterations. Teams are composed of individuals, and they are different. We know that each team has a specific velocity, production capacity, and learning curve. This is normal. Nevertheless, it makes all the difference to have the teams aligned in terms of planning. In this case, aligning the start and the end of a release iteration in a time-box makes the teams free to perform as many iterations as they need. The difference is that the teams are aligned in the project milestones, finishing all the iterations together and making it possible to, for example, plan the deployment of a new version of the product on a specific date.



Another technique is to hold staggered daily meetings. This is nothing more than sequencing the daily meetings so that representatives of each team can potentially participate in the other teams' daily meetings, to identify dependencies and risks and to find specific skills. The most interesting fact is that we also have a daily Scrum of Scrums, in which the technical leaders of each team meet up to follow up the project with a full perspective.



Another interesting fact is to realize that the technique is scalable. I mean, Scrum of Scrums is a generic model that can be applied to other levels of projects/programs/portfolios, according to need. For example, imagine a project of 100 people. In this case, we would have 10 teams of 10 people; however, 10 teams are more than the recommended number of 9. Then we would create another team taken to a higher level, creating a Scrum of Scrums of Scrums!



Remember that, although we have not discussed it here, event planning, review, and retrospective can also be adapted to fit into a Scrum of Scrums model.

Opinions represent those of the author and not of Scrum Alliance. The sharing of member-contributed content on this site does not imply endorsement of specific Scrum methods or practices beyond those taught by Scrum Alliance Certified Trainers and Coaches.

https://www.scrumalliance.org/community/articles/2013/june/scrum-of-scrums-running-agile-on-large-projects

Comments

Xi (Amy) Wu, CSM, 6/25/2013 2:12:03 PM
This approach is very logic and sensible. However I wonder how it works in reality?

Randall Schmidt, CSP,CSM, 3/11/2015 1:02:14 PM
Works very well