

The Art of Compromise: Scrum and Project Governance

by Mike Cohn • 0 Comments • originally published in ebiz on 2009-12-22

The concepts of agility and project governance are not fundamentally opposed. Each is an attempt to improve the finished product. Scrum strives to do this through close collaboration and the short inspect-and-adapt cycles of the timeboxed sprints. Project governance strives to do it by what we might call inspect-and-approve (or reject) checkpoints in which the product or project is compared to a set of desirable attributes.

However, while pursuing similar goals, Scrum and project governance take entirely different routes to achieving those goals. It is in these different routes where problems will arise in mixing the two. Fortunately, a few compromises on each side, combined with the following actions, can lead to a successful combination of agility and oversight.

Negotiate and set expectations up front. Undoubtedly, the first Scrum project to go through the governance process in your company will have challenges. There will almost certainly be some things they cannot do; for example, a Scrum team cannot provide a thorough design before getting permission to start coding, because design and coding will be done concurrently. The only solution to this is for the team to negotiate with the necessary governance groups in advance. The more support a team has for this and the higher up in the organization that this support reaches, the better. The team does not need to solicit a permanent change in governance policies. The change can be pitched as a one-time experiment.

Fit your reporting to current expectations. The project review boards or oversight committees that provide project governance have existing expectations for what information each project is to provide at each checkpoint. Don't fight these expectations. If they expect a Gantt chart, provide a Gantt chart. When you can, however, try to slowly shift expectations by providing additional, more agile-friendly information. If burndown charts are suitable to show, do so. Or perhaps you want to include a report showing the number of times the build server kicked off continuously integrated builds and the thousands (or perhaps tens or hundreds of thousands) of

test runs that were executed.

Invite them into your process. Scrum teams can supplement less-detailed formal governance checkpoints by inviting governance committee members to participate in the regular meetings they will hold. I like to extend the well-known technique of management by walking around into management by standing around. Encourage managers and executives involved in the governance of a project to attend the daily scrums, where they can stand and listen to what is occurring on the project. The same shift from documents to discussions that is created by working with user stories needs to occur with project reporting. Encourage people to visit the team or join its meetings to see for themselves what is being built.

Reference a success. Nothing convinces like success. Do whatever you can to get a first scrum project or two through lightened or reduced governance checkpoints. Then point to the success of those projects as evidence that future projects should also be allowed through.

It's one thing to look at agile software development in a test tube; it's another to experience it in the real world. In the test tube, agile methodologies like Scrum are easily adopted by all members, and the nasty realities of corporate politics, economics, and such cannot intrude. In the real world, though, all of these unpleasant issues do exist. It is rarely as simple as deciding to use Scrum and then being able to do so with no other constraints.

Because few organizations will go so far initially as to completely revamp their current approaches to project governance, teams need to work with their organization's non-agile governance. Adopting an attitude of compromise and taking the actions outlined above will go a long way towards easing those first Scrum projects through the gates.

Posted: December 22, 2009

Tagged: user stories, sprinting, teams, management, daily scrum, agile projects, agile software development, scrum project
