The Scrum Idea

By Richard Stanley, Apex Systems Java/Open Source Practice Director

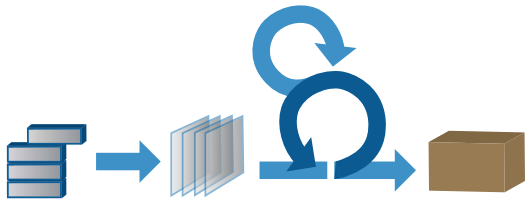
Today's technical arena is the wild wild west of new technologies and Open Source evolution. While projects can be completed in less and less time thanks to the constant evolution of tools; development is fraught with the dangers of over-designing, under-budgeting, and illogical planning. Various sources of data show that as little as 17% of software projects come in on time or under budget, with 68% of those projects still failing. What can we do to ensure that projects do not fail, and find a way toward the constant delivery of value-added product in a reasonable timeline?

Scrum, at its' very core, is an attempt to bring the idea of complicated design and development in to a world where long-term guess work does not make sense. Scrum is not a magic wand that can be woven over a company and expect results to be exacting or immediate. In particular, the movement to Scrum starts with the adaptation of a specific philosophy. This philosophy brings emphasis on the reality of failure by asking a company to realize that "complicated projects are hard" and that "one cannot realistically plan for or project for something so complicated." As a result, Scrum is a transformation of a company's way of thinking.

"In short, Scrum offers companies the ability to tackle the ever changing industry by choosing to adapt with it, rather than plan ahead. "

In order to more properly work through complicated design, Scrum asks that a company accept the following:

1. As technology projects are complicated, they cannot have a set budget or hard date
2. Software is, and always will be, a constant process
3. Engineers, and not management, are uniquely qualified to decide how best to solve problems
4. All development time should be divided in to increments where the process adds SOME value
5. The business should only be concerned with the order in which it makes the most sense to address business needs, not the technical approach to those problems
6. Scrum is "all or nothing"; part Scrum is not Scrum (though it MAY be agile)



Most companies that do "Scrum" have a hard time adapting these ideas, and so find little or no success with Scrum. As a result, only 8% of companies claiming to be using the Scrum method are actually doing it as is prescribed.

The implementation of Scrum is based around time-boxed events[1]. This means there are certain events which happen at certain times and with a very specific amount of time in with which to be completed. This concept keeps the overall process moving forward and ensures that at the end of a "Sprint," there is some kind of deliverable which adds value to the company. The various time-boxed events are tools used to ensure that everyone stays on task, but gets a chance to speak. It is essential for Scrum to be successful, that since we do not hold a team to a specific number of hours, that we instead can continuously justify the existence of the development process to the stakeholders via the value added.

In short, Scrum offers companies the ability to tackle the ever changing industry by choosing to adapt with it, rather than plan ahead. If we know that the environment, customer's needs, and technologies available in one year would be the same that they are now, we may not need Scrum. However, the reality is that with complicated projects, under-estimation can be more damaging than steady and logical progress in the right direction.

[1]For Definitions of Time-Boxed Events, Roles and other specifics, please see the Official Scrum guide at Scrum.org

About Richard Stanley

As the Java/Open Source Practice Director for Apex Systems, Richard provides high level architecture analysis and consultation on all Java/Open Source technologies. He partners with technology managers to help identify specific technical needs and determine the best strategy for success. In addition, Richard helps train Apex internal employees about Java/Open Source technologies.

Richard is a Certified ScrumMaster and has written CRM and LoyaltyScript applications and a virtual POS system prototype that was later used as a basis for the company's current iOS and Android applications. He has worked in a variety of industries throughout his career, including higher education, government, healthcare, finance and software. Richard has been in IT since 1999 when he worked as an MIS Database Administrator/Programmer for MCIWorldcom. He has a BSE in Computer Science and BS of Computer Information Systems from Northern Arizona University and a Bachelor of Science from Aoyama Gakuin University in Japan.

