Traditional Project Management vs Scrum: Adapting Square Pegs to Round Holes

By Frank Rios

Traditional IT project managers have struggled to use the PMI methodology when it comes to software development for decades. Using the traditional project management methodology for software development is similar to trying to put a square peg into a round hole; you can force it, but it just does not fit as well as it should.

Over the past several years, the Agile methodology has really started to gain momentum. This is in large part due to the popularity of Scrum, even though Agile has been around for nearly two decades. Scrum is one of several frameworks that fall under the Agile umbrella. Some of the others include Extreme Programming (XP), Rational Unified Process (RUP), and Design for Six Sigma.

Control Theory

There are generally two different types of control theory. The first is the defined (or theoretical) process. This is what traditional project management follows; it's all about command and control. There are lots and lots of planning. You plan what you expect to happen, then enforce the plan; sometimes regardless of the conditions. Finally, this process makes use of change control. You will often find a change control board that oversees any change requests.

Scrum, on the other hand, employs what is known as the empirical process. In this process, you learn as you proceed. Instead of planning everything up front and planning on how to handle change, the empirical process states to "plan for change." As a matter of fact, the empirical process embraces change through inspection and adaptation; two of the three pillars that uphold every implementation of empirical process control.

The Three Pillars

Traditional project management for years has followed the "iron triangle"; time, cost and quality. These are still the three pillars every project manager must juggle. In an ideal world, projects would be delivered on time, under budget, and be of the utmost quality. In reality, this rarely happens. For software development projects, obtaining all three never happens.

A ScrumMaster also follows three pillars. Their pillars are transparency, inspection and adaptation. Transparency involves open communication with all members of the Scrum project team, and the ScrumMaster proudly displays their team's burn-down charts where everyone can see. They also review how well they did at during their sprint during what is known as a Sprint Review. Finally, adaptation involves making changes and improvements to tasks that can be improved.

Beginning the Transition

For many companies trying to make the transition to Agile, the first thing they must understand is that a good project manager will not necessarily make a good ScrumMaster. They are not directly interchangeable. Contrary to some thought, a good ScrumMaster does not have to have Project Management experience.

As a matter of practice, the best ScrumMaster's are generally very technical. Former SME's and technical leads make for a great ScrumMaster. This is because they can better empathise with developers, they understand the big difference between level of effort (LOE) and duration, and they can better help prioritise features along with the Product Owner.
Know Your Role

There are three primary roles in Scrum: the ScrumMaster, the Product Owner, and the Team. Oftentimes, you'll hear people being referred to as "chickens" or "pigs". People who make up any of the three primary roles are referred to as "pigs", while everyone else is referred to as "chickens". A "pig" is someone who is committed to the project, whereas a "chicken" is someone who simply involved.

The origin of these terms comes from the following story:

"A chicken and a pig are together when the chicken says, "Let's start a restaurant!" The pig thinks it over and says, "What would we call this restaurant?" The chicken says, "Ham n' Eggs!" The pig says, "No thanks, I'd be committed, but you'd only be involved!"

ScrumMaster

The ScrumMaster's primary job is to adhere to Scrum values, practices and rules. They are an advocate for Scrum and help it get accepted and adopted throughout the organisation. They also act as the figurative "shield". The ScrumMaster protects the Team from outside political noise and ensures nobody goes directly to any team member without following the proper chain-of-command. This allows the team to remain focused on the job at hand, and if any issue is a priority, the ScrumMaster and Product Owner will discuss it and prioritise it within the Product Backlog as appropriate.

Product Owner

The Product Owner's primary responsibility is to manage the Product Backlog. The Product Owner is a single person, not a committee. The collection of stakeholders can influence the Product Owner, but the Product Owner has the final say. The Product Owner sets the priority of each feature/request. For new Product Owners, the ScrumMaster will work closely to teach him or her how to do their job.

The Team

The Team is responsible for turning items on the Product Backlog into potentially shippable functionality every Sprint. The Scrum Team is cross-functional. In other words, they consist of people with one or more specialties; including, but not limited to quality control, development, database design, business analysis. The team is self-organising and self-managing. As such, everyone has the same title: Scrum Team Member.

The team size should be around seven 7 people, plus or minus 2. This size does not include the ScrumMaster and Product Owner (unless they are pigs who work on tasks included in the Sprint Backlog).

Transition Hurdles

Agile Methodology does not conform to PMI Methodology. This is absolutely the largest hurdle to overcome and where the internal conflict of project managers occurs; even more so for seasoned PMP's. To successfully complete the transition, the department must choose one or the other when it comes to Software Development. Failure to conform to the Agile principles will lead to a failed transition.

Dual Role or Two Different Resources

Any transition to Agile is in-and-of-itself a project. Therefore, a Project Manager should be chosen to lead this transition. Also, the ScrumMaster's life cycle is revolved around software development; which is only a subset of the entire project life cycle.

As any Project Manager is aware, being a Project Manager is a full-time job. Being a ScrumMaster is also a full-time job. The big question is can a single resource successfully perform both roles? The answer, like so many requirements developers are given is, "it depends." Some companies will try to fill both of these positions with a single resource due to budget constraints or other reasons. This is a perfectly acceptable reason, but not necessarily the best one. A Project Manager who is a
Certified ScrumMaster can perform this dual role, but this is discouraged.

**Deprogramming Project Managers**

Traditional plan-driven project managers must be deprogrammed before one they can become a successful agile project manager. President Eisenhower said it best when he said, "Planning is essential, plans are useless." That phrase sums up the biggest difference between Agile and PMI. Success is no longer measured by how well the triple-constraints are balanced; it is only measured by the Customer. Project scope is no longer the driver; scope is driven by time and budget. No longer is success measured by the completion of tasks and phase-gate reviews it is measured by the delivery of features and functions. Finally, learn to embrace change; love it, live for it.

**Working Together**

The Project Manager and ScrumMaster must be treated as peers if the project is to be successful. The Project Manager is in charge of the entire project, whereas the ScrumMaster is in charge of the software development portion of the project. It is very important that Management respect the difference.

During the actual software development, the project manager must let the ScrumMaster run Scrum using the Agile methodology, not the PMI methodology. As a Project Manager, the tricky part is "letting go" and trusting the ScrumMaster. This portion of the EXECUTE process group must be considered a "black box". The Project Manager is now considered a "chicken"; he or she can listen in, but carries no weight.

**Becoming Truly Agile**

Many companies feel they are Agile simply because they are doing iterative software development. However, doing Waterfall in an iterative fashion is definitely not Agile. Agile is much more than iterative development and rapid releases. The traditional PMI way of thinking cannot be the guide while implementing Agile; it will become an impediment. It is a whole new way of thinking; a whole new philosophy. To become truly agile, the entire IT department must go through a paradigm shift.

Frank Rios is founder and consultant at Protean IT Management, LLC. He is an experienced technical consultant, leader, and project manager in both the Public and Private sectors with a solid background in IT project management, application architecture and application development. His specialties are technical business and Agile project management, programme management oversight, and process improvement.

Through his combination of business and technical expertise, Frank has been able to provide added value to teams and companies alike, by increasing communication between business and technical departments, reducing ambiguous requirements, and increasing time-to-market.

His formal education includes two Master's degrees; a Master's in Computer Science (MCS) and a Master's in Business Administration (MBA) with a concentration in Project Management. His professional certifications include Project Management Professional (PMP), Six Sigma Green Belt (SSGB) and Certified ScrumMaster (CSM).

© Project Smart 2000-2010. All rights reserved.