

CSM Practicing Certification Renewal Assessment

Name: Craig Larman email: craig@craiglarman.com date: April 10, 2005

Scrum depends on the inspect and adapt mechanisms of process control to manage the complexity of projects. For inspection to work, everyone must know what is being made visible. To implement the Scrum process, such regulating mechanisms as defined roles, involvement versus commitment, time-boxes, and regular cycles are used.

1. Describe one project on which you have used Scrum over the last twelve months. Describe:

- Purpose - what business goal was the project intended to deliver?

Recently, simulation system for the design and verification of nuclear fuel rod assemblies for fission reactors.

Note, as an aside, I've been coaching teams in applying Scrum since 1999 (based on your PLOPD published material), first on a simulation system for oil reservoir modeling.

In my role as chief scientist for Valtech (a company of about 1400 in 8 countries) i've spread Scrum practices to all our offices/projects worldwide, including our Bangalore outsourcing division. I've also been promoting and coaching Scrum to myriad industry customers worldwide, in my role as process consultant and coach, since 1999.

- Length - what was the duration of the project?

6 months; i now visit semi-regularly in the role of process coach

- Cost - what were the budgeted and actual costs?

Cost was not tracked; not relevant to the project, as the developers are a fixed sunk cost, and this is the division's flagship product. However, the project involves about 15 fulltime developers, so I can estimate that the cost is in the range of \$800,000 USD.

- Value - what were the projected benefits and actual (if measured) actual benefits?

This is a re-write of a 20-30yr old FORTRAN system, the goal of which is ease of extension for new and improved simulation models, and to satisfy some immediate customers who are requiring an OO version of the system.

Their immediate customers have agreed to purchase based on the promise of a rewrite and delivery of the new system, and that agreement is worth tens of millions of dollars. So this is a make-or-break critical project.

Improved ease of extension/modification won't be quantified, but the team recognizes the current

system is near impossible to modify or maintain (or comprehend), and anything would be an improvement.

I will also mention that this project was recently audited by the nuclear regulators because it was the first 'agile' project. In addition to applying Scrum, the team adopted all of my recommendations for modeling, documenting, traceability, etc., based on my agile UP approach (see my book 'Applying UML and Patterns' – Larman2003) which extend far beyond Scrum, and are important in this life-critical domain. The auditor gave this project a very rare "no findings of any problems" (the highest rating) and called the project a "model of how to run and document a project, to be emulated by others."

- Size - how many people were on the project team(s), how were they organized into teams?

Approximately 15 developers. Two teams, in two cities, organized by architectural modules (Conway's Law).

- Teams - were the teams cross-functional and self-organizing?

yes

- Were the teams collocated in an open space?

Only one of the teams; one team is in a location in which an open space is a battle they can't win.

- Were the teams physically separated within one location, or located in more than one physical location?

two cities

- Initiation - how was the project initiated? How was the team trained to use the Scrum process?

Initiated by management and sales. Trained in Scrum by my coaching first during a short process workshop, followed by iteration-1 application with me coaching.

- Reporting - how did you report progress to management and the customers?

By iteration demos, during which a summary of features implemented is also summarized.

- Change - what difficulties were surfaced by Scrum that had to be resolved? How were these resolved?

Friction between the architect and other team members who were not interested in his architectural vision (because of their own ideas). The increased communication via daily Scrums, plus coaching and application of whiteboard agile modeling with UML to discuss the designs,

broke through the communication impasse.

Prioritization (“valuation”): Being forced to consider what’s most important to them and a small step, and doing it. In their case, a critical priority is verification of correctness – the new sim system must produce identical results to the old FORTRAN version.

- Management - what was the previous role of the ScrumMaster? Who took on the role of Product Owner? To what degree were they successful in fulfilling their roles?

After I kick-started things in the SM role, the ScrumMaster was played by a senior engineer. Product Owner is a senior nuclear physicist.

- Engineering - what software engineering practices or environment had to be changed?

everything . they were “FORTRAN physicists” with virtually no modern sw eng skills or team dev skills. e.g., source control, issue tracking, testing, integration, modeling, ...

This was a FORTRAN culture of physicists rather than software engineers, so they had every poor practice you can imagine.

- Stabilization - for how long did the software have to be stabilized before it could be released? How did you structure this stabilization process?

Stabilization, which is underway, occurs through continuous integration based on CruiseControl, combined with automated unit and acceptance tests.

- Success - to what degree was the project successful? To what degree was the Scrum process instrumental in the success of the project?

Underway... Scrum was instrumental to establishing a culture of teamwork and communication between what has traditionally been a _very_ individualistic cowboy culture of engineer-programmers who aren’t used to building software together.

- Scrum Process - to what degree was the Scrum process implemented "out of the box?" To what degree did you have to modify the Scrum process for this project? For each modification, how did you formulate the modification so that the basic inspect/adapt mechanisms continued to function? What parts of Scrum couldn't be implemented, or failed, and why?

Most of Scrum was applied, but also with agile UP (e.g., some of the artifacts) and some XP practices (e.g., CI, TDD). Scrum meetings across the 2 cities are done with video conferencing technology.

variants:

the team works on 2-week iterations

2. How do you cause the accuracy of Product Backlog estimates to improve? To what degree does their accuracy matter?

First, the need for accuracy is project-dependent. For product companies with an external event milestone release (e.g., a trade show) it is of greater interest.

Most of my teams keep track of actuals; some use these in later estimation activities to get a sense of variance and how to improve. More generally, I coach teams to apply wide-band Delphi combined with relative point estimates in addition to absolute effort estimates.

3. How do you cause the accuracy of what a team commits to for a Sprint to what the team actually delivers?

Several methods: First, before the detailed Sprint goals are written, the team does "scope clarification" analysis on the tentative wishlist, so there is less ambiguity as to goals and their scope. Also, in contrast to classic Scrum in which a large "batch" of things are chosen, I first have the team estimate their capacity (in IEH) and then pick off only *one* goal at a time and then do detailed task analysis and effort totals for one small goal. Only if there is capacity remaining, is another goal chosen, until goals and tasks consume all team capacity. Also, the Sprint Backlog's requirement list (to fulfill the Sprint tasks) is very clearly defined and written with lots of scope clarification, so there is less ambiguity. I also push the team during Sprint estimation to be very pessimistic in their ability to deliver, so that they don't over-promise. Concretely, we apply 'estimation poker' during detailed task estimates.

4. What metrics do you use to track the development process? Which metrics have been changed, removed, or newly implemented as a result of using Scrum?

I'm not sure if this question means *process* metrics versus *project* metrics (e.g., velocity). We keep no process-specific metrics (e.g., # of agile method practices applied in the last iteration; team member satisfaction with applying the practices). Review of the process itself and its improvement happens at a Sprint retrospective via conversation and whiteboarding activities.

5. What type of training, resources, or tools would best help you successfully employ Scrum in the future?

Education specifically for product marketing, to participate better in Scrum.

6. (Optional) Scrum and Extreme Programming are sometimes used together. What must be considered when this is done?