CSM Practicing Certification Renewal Assessment

Name: Jay Conne email: Jay@jconne.com date: January 23, 2006
Certification: Boston, MA June 14, 2003

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?
  The project's purpose is to improve the clarity and accuracy of high volume, automated customer communication about credit decisions. In addition, it will provide an enterprise-wide consistency and provide a new level of maintainability by reducing two active classes of components by factors of 6 and 10.
- Length - what was the duration of the project?
  11 months external drop-dead date, with an internal target of 8 months to cover contingencies.
- Cost - what were the budgeted and actual costs?
  I do not have access to this and it was not a constraint.
- Value - what were the projected benefits and actual benefits (if measured) ?
  This project is required to meet external regulatory demands. In addition, the internal value is to have customers and prospective customers better understand the credit decisions and the reasons for those decisions. Prior communication of this kind with customers frequently had serious quality problems including the use of internal company jargon. This project should improve the company's credibility and decrease customer service calls. As stated in the Purpose above, this should dramatically increase the maintainability of the application by dramatically reducing the complexity.
- Size - how many people were on the project team(s), how were they organized into teams?
  Before the first iteration, the team was split to address two relatively independent aspects of the project. The first team had about six core members, including two dedicated off-shore coders, and was able to start iterating immediately on 2-week intervals. The second team was much larger and varied from 10 to 20 as the company was negotiating approval on their approach. One senior subject matter expert was on both teams and was the Product Owner. A traditional Project Manager was largely consumed with logistics – coordinating external resources while trying to also act as an Apprentice Coach (ScrumMaster).
- Teams - were the teams cross-functional and self-organizing? Were the teams collocated in an open space? Were the teams physically separated within one location, or located in more than one physical location?
  The two teams shared one relatively large (~625 sq.ft.) team room and had some dedicated off-shore coding resources. Each team had a rectangle of tables set away from the walls with utilities in the center of each table group. Each team had 20 feet of whiteboard on opposite sides of the room. We had also considered getting some cubicles that were just outside the team room to allow for locked storage of personal items and for some phone privacy for calls on non-team business. This was never implemented.
- Initiation - how was the project initiated? How was the team trained to use the Scrum process?
  The teams were trained together with 3.5 days of "Discovery".
  - The first two days were PROCESS DISCOVERY, which introduced Lean/Agile/Scum principles, roles and processes with interleaved presentations, group discussions and more structured exercises. The exercises concretized concepts of push/pull, batch

size and flow, self organization, the Scrum process and what 'done' means. We also introduced some XP concepts such as Pair Programming, Test Driven Development, and Continuous Integration & Testing. Part of one day was set aside to visit other Agile Team rooms where members of those teams described their approaches and answered questions.

- The third day was PROJECT DISCOVERY, which included the Project Sponsor, a senior executive, challenging the team(s) to openly embrace the change from waterfall to Agile methodology. She also described the critical importance of this project to the firm meeting their standards for customer satisfaction and for preserving their legal status under regulation. Then the group worked as a whole or in smaller groups to develop some global project artifacts including: Vision statement; Mission statement; Objectives / Outputs / Outcomes (O^3); Scope: In / Not-in / Who (for Not-ins); and Project Sliders ala Rob Tomsett for project constraints.
- The fourth day was only a half-day of TEAM DISCOVERY, which included a group process to determine: initial team norms; core hours; Stand-up (Daily Scrum) time, what 'done' means, and more. Also the team decided on the division of the original team into two:
    - The first team was prepared to immediately start iterating. They selected a 2-week iteration cycle that fit the current release cycle for this application.
    - The second team was not ready to iterate. Under leadership of the Product Owner / SME, they addressed the approval process and further analysis to establish a prioritized Product Backlog for the second part.

- Reporting - how did you report progress to management and the customers?
    - Continuous reporting to those in the room and to visitors was radiated from the walls and magnetic white-boards.
        - The artifacts of the Project Discovery and Team Discovery were posted high on the walls above the white-boards to keep context visible.
        - The white-boards had the burn-down chart and movable cards for stories, conditions of satisfaction and associated tasks. An interesting innovation we copied from another team room was to use the colored magnets to hold the cards and color code the state of the tasks – not-started, in-process, test-me, and done.
    - The Project Manager was responsible for weekly reporting of time and progress to meet the company's software development methodology requirements. Those requirements were modified to be more accommodating to Agile project concepts including iterations, incremental-releases, velocity determination and Product Backlog.
    - The Project Sponsor was so committed that she scheduled a half-hour meeting, twice a week after the normal work hours to hear any issues for this and another Agile project.
    - Iteration reporting to the Product Owner and other stakeholders were the Scrum standards of iteration (Sprint) Reviews and Iteration Retrospectives captured with a 2-column flip-chart for OK and DELTA (Change) – a simple focus that is posted on the team room wall.
- Change - what difficulties were surfaced by Scrum that had to be resolved? How were these resolved?

  The first iteration failed to deliver due to a dependency on an external organization that misunderstood a requirement. It was an opportunity to celebrate the lesson learned and to correct the mistake. The second team was not able to get approval for their process by the time I left this project. After I left, I learned that they were having trouble maintaining core hours for much of team that was being pulled to other demands.
- 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?
    - I was dropped in from outside the organization as a Consultant – Master Coach (ScrumMaster). Transition was mitigated by pairing me with a 'buddy' Consultant –

Master Coach who had a year of experience on this site. This 'buddy' and other consulting coaches shared the materials they had evolved over their time on-site. Lack of access to the customer's very secure, web-based collaboration environment and PM hostility made success in this role impossible. In spite of that, we got the team trained and an initial 2-week iteration started in the second week of the project that resulted in a delighted customer.

- Initially the Product Owner was committed to only a few hours a week with the team. This was raised as an issue with the Sponsor before the Discovery week and she changed his priorities to have him with the team almost all of the week. This was not sustainable with the true demands on him. After I left the project, I learned that it continued to be an issue. Also, this Product Owner was skeptical about the Lean-Agile approach and then became a strong advocate in the Discovery week.
- A Project Manager was pre-ordained to be an Apprentice Coach. At the start she expressed considerable hostility toward the designated Product Owner and expressed the opinion that the consulting coach would not be needed after the first iteration when she would take over. Her assumption was that the Coach role required only learning some tasks, not new or deep insights. This person was clearly not suitable for the role in values or temperament. This was a mistake that was not repeated on future projects I initiated for this firm. In those projects, the sponsors and champions agreed to not have a designated apprentice at the start. A qualified apprentice candidate would emerge and be a choice the team and Coach could discover. Each of those other Project Managers were interviewed and told of the mistakes we had made with the pre-ordained approach; the expectations for the role and the alternatives that were open to them, especially that it was not a career ending decision as some had thought.
- When I left the project another Master Coach and senior Practice Manager took over to work the political and Project Manager issues.
- The Project Sponsor was fully committed to the Scrum discipline and wanted to be informed of any issues. In charging the team during Project Discovery, she asked all present to come to her with any concerns or issues. After I left the project, others were not prepared to satisfy this Sponsor's request.

- Engineering - what software engineering practices or environment had to be changed?
  The technical environment for the first team was previously well established with a two-week release schedule for sustaining engineering. For the Agile context with off-shore programmers, the team had the off-shore code units returned with unit tests written by those programmers. While not formally 'Test Driven Development', it permitted the team to aggregate the unit tests into a regression testing discipline, verify the unit test code and the success of the tests.
- Stabilization - for how long did the software have to be stabilized before it could be released? How did you structure this stabilization process?
  This project was an expansion of an existing sustaining engineering process. The plan was to have a two week iteration to develop a potentially shippable increment and then have a subsequent 2-week cycle for the standard release process to run its course. The first iteration was slipped another 2-week cycle to correct the discovered error
- Success - to what degree was the project successful? To what degree was the Scrum process instrumental in the success of the project?
  The first iteration was able to identify and commit to an initial increment that corrected over 98% of the production volume while only being a tiny fraction of the code changes for the other 2% of the volume. Although it was put into production 2 weeks later than planned, the Sponsor was ecstatic. This choice of scope was a direct result of the Scrum principle of identifying a manageable bite of highest value backlog for an iteration.
- 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?
We had 2-week iterations that matched the existing release cycle. The reporting requirements were much heavier than the Scrum model. This was handled by having a Project Manager own those tasks as was their responsibility. The sponsor and team were open to adopting Scrum to the extent possible.

2. How do you cause the accuracy of Product Backlog estimates to improve? To what degree does their accuracy matter?
Starting the first sprint with a best guess estimate is as good as it can get at that stage. Over time, the team members get more insightful about their capabilities, the project and their chosen technology. However, estimating accurately is of relatively small import compared to learning what velocity a particular team can sustain on a particular project and technology. This is how Scrum satisfies the CMM goals of predictability. It can get no better.

3. How do you cause the accuracy of what a team commits to for a Sprint to what the team actually delivers?
Focus on three principles: Commitment to Reality, Continuous Learning, and Maintaining Trust. The first two yield the third. Hold people to a daily review of their specific commitments and maintaining that with the burn-down chart updated one or more times a day. Each iteration retrospective provides an opportunity to evaluate and adjust their process.

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?
The Scrum standard burn-down chart and story/task board are used to track day-to-day progress. The Product Backlog was tracked in a spreadsheet maintained by the Product Owner. The customer modified their standard software development reporting requirements to try to accommodate the Scrum model with the project / release / iteration hierarchy.

5. What type of training, resources, or tools would best help you successfully employ Scrum in the future?
As a professional in technical training for many years, I am working with other Lean/Agile/Scrum coaches to develop a modular collection of training components structured for train-the-trainer. I will use a model that has been successfully used by my training development and delivery teams for many decades. This will be structured for continuous expansion and improvement with shared IP rights and attributing to the max.

6. (Optional) Scrum and Extreme Programming are sometimes used together. What must be considered when this is done?
Scrum and XP address independent dimensions of project management and software development. Scrum focuses on the quality and frequency of communication between team members and between the team and other stakeholders. XP focuses on two goals:
- deliver only features that match what the customer is willing to pay for; and
- a discipline that produces more error-free code – build in quality (quality is free).

The intersection of these two disciplines can be ‘conditions of satisfaction’ for a story that are expressed as executable code. In a more ideal future, we will be able to specify, code, test and document with a single, simple, consistent, readable and executable syntax and semantics. That is a goal that I have been supporting with my involvement in new language development in and around MIT. Its all about making the process better fit the way we think and communicate.