

CSM Practicing Certification Renewal Assessment

Name: Roger Brown email: roger.w.brown@valley.net date: 3/1/06
Certification: Boston 5/10/05

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?

Our company is one of the two main producers of digital map data. We sell to partners in internet mapping, automotive and personal navigation and governmental GIS work. The purpose of the project was to create an API and repository for our partners to use to reporting errors and omissions in our data. They will create applications to pass these reports to us from their end users. The project consists of a set of web services to provide mapping user interface functionality, a data collection service, a data accumulation service, data analysis agents and a sample web site client with a suggested user interface.

- Length - what was the duration of the project?

The project is in its fifth month. It is scheduled to go into service in two more months.

- Cost - what were the budgeted and actual costs?

The budget was \$200,000 for development. The actual hardware costs are \$150,000. Personnel cost for development is 4 Engineering FTE for 7 months.

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

This is not a for-sale product so there has been no clear dollar value assigned. The value will come from a reduction in manual processing of customer reports, improved product value and extra sales due to this new service to our partners. Many partners have asked specifically for this capability as an additional benefit to their end users, leading to higher sales for them and higher license revenues for us.

Our project had an unexpected added value of serving as a vanguard Scrum project in an organization that was ripe for a process change.

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

The project is a single scrum team consisting of 4 engineers and a part-time functional tester. The team's functional manager has also participated as Scrum Master for 3 sprints. We have had two product owners for different phases. I have participated as both an engineer for 7 sprints and as Scrum Master for 3 sprints.

- 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 core team consists of 3 senior software engineers and an experienced engineer with functional testing experience. We had two part-time members join two sprints to provide

special expertise or manual labor. We work in a shared room with a single shared private cube adjacent. We work in rotating pairs. In terms of functions and self-management, we have made it a point to trade skills across several technologies – PHP for the UI, Java for the middle tier and MySQL and Oracle for the databases. We make it a point to reduce specialization and code ownership. As ScrumMaster in a small team, I was also a team member. For those sprints, I made it a point to take less critical tasks in disciplines that were unfamiliar to me – QA and deployment. We chose tasks and partners on a daily basis, deliberately sharing the knowledge and mixing up the programming pairs.

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

This was an entirely new project taken on in a new company by a team of people who had worked together to some extent in a previous company. We were allowed to choose our own technology, methods and process. We had done Scrum pilots at the previous company and so had a basic understanding of the process. I and the functional manager (who is not a team member) are both Certified Scrum Masters.

We started the project with a short “practice” sprint to achieve two goals: become familiar with the Scrum rubric and to assess two possible technologies for the project. We learned a lot about estimating, managing the daily meeting and personal interaction in this first try. We then did an official “foundations” sprint in which we implemented a large part of our chosen architecture with minimal functionality. This gave us a chance to increase our understanding of the process. As this sprint was also a highly visible event in the company, we made it a point to produce the full set of project products for maximal impact in the first demo – working software illustrating our design for a distributed topology, API documentation, an As-Built document describing our efforts, unit test suites and a functional test suite.

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

We report outward in these ways.

- Our project is publicized to any interested party on an internal Wiki site. We publish all documents resulting from a sprint as well as the daily burndown chart.
- Our functional manager has a weekly call with the executive sponsor.
- We do a product demo for the product owner (an internal employee), future product manager and any interested upper managers at the end of each sprint.
- We have published an internal report of “Scrum Lessons” for upper management as part of our Agile Adoption efforts in the organization.
- I have conducted presentations on Scrum and TDD for other engineering teams using our team experience as one example.
- We have had conference calls with our one pilot partner and done one demo for them. We also provided an externally visible preview site for them to use and generate feedback in Sprint 4 of 6.
- We have a constant task for each sprint to produce an “As Built” document describing what we did for both historical purposes and for reference by anyone who is monitoring our progress. These are also a vehicle for knowledge sharing as team membership changes.

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

We were a new team on a new project using methods that are new to the company. The difficulties that appeared were fairly predictable. Only some were due to Scrum:

- We have encountered a budget-induced blockage in the acquisition of necessary hardware despite having produced the required "spec" well in advance of our needs. The current Scrum Master (and group manager) is working on this problem.
- We are introducing Scrum into the organization and have had a very good buy-in from upper management. Two other high-profile projects are using Scrum and I am about to start a new one. We have encountered resistance and skepticism at the middle management level. We saw the same thing at our last company. Our approach is one of education and visibility. I have started in a role of Agile Coach and have begun making presentations and classes to other teams and potential Product Owners. This will continue.

- 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?

We have had 3 people try out the Scrum Master role:

- The team's functional manager has mastered some of the scrums. He is a CSM and has previous experience as a technical team manager, though he does not have a technical background.
- I have been SM for 3 Sprints. I have also managed development teams in the past but am not currently in a management role. I am a veteran software engineer.
- One of the team engineers was SM for one sprint with help from our two CSMs. He has just been certified as well.

Here are some observations about our Scrum Master experience:

- There is a conflict between the roles of functional manager and Scrum Master by virtue of the different relationships between the manager and the team members. The functional manager has more authority to affect the productivity of a team through traditional organization channels (one-on-one discussions, career development activities, performance reviews, etc.) As Scrum Master the manager needs to step back a bit to allow for team self-management while still making adjustments in personal motivation and team membership. I think it would be difficult to manage both relationships at the same time.
- I played the role of both team member and Scrum Master in this small team. In the sprints where I was the SM, there was some conflict between my need to inspect and adapt the process and my input to technical decisions. There was a natural tendency for team members to grant some authority to the current Scrum Master. I had to be careful to not use that granted authority to influence technical decisions. I resolved this fairly successfully by recusing myself from big decisions and by performing the less appealing functional roles in those sprints (deployment, documentation, functional testing).
- The functional manager had more contact with the outside world for unblocking issues and information sharing. When I was in the SM role, I had to channel that through him. When he was SM, he could do it directly.

- There was some desire expressed by the team for a “permanent” Scrum Master to have more continuity in connections with the various product owners across sprints.

We have had two different product owners. In two sprints devoted to infrastructure, our functional manager played the PO role. The first product owner was a sales manager who handled the account for our first pilot partner. She was excellent in the role and learned it very quickly and very well. The second product owner is more internally focused and has done fine so far in helping us set priorities and fill out the backlog. He, however, has not been able to allocate the amount of time that we would like to answer questions and keep up momentum in the planning process.

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

This team migrated together from a 100% waterfall environment where we had been advocating and piloting agile methods. In the new company we had carte-blanche to do as we chose. The set of practices we implemented includes Scrum, Test Driven Development, Co-location and Pair Programming. The change was from our dabbling in agile methods in a waterfall regime to a complete embrace of the methods that we thought best. Our approach was novel in the new company and has been instrumental in promoting the adoption of these methods on other projects, even becoming a part of newly identified corporate “best practices”.

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

Our project is fresh from the start and we follow the agile tenants of continuous integration and a deployable product at the end of each sprint. To that end, we are very thorough in ensuring the scalability and robustness of the product. We all have experience in this kind of system so we have made testing an integral part of the development process, including unit testing, functional testing, stress and quality testing and integration testing. We have had specific development tasks to create tools for creating test data and running tests so that we can answer questions about scalability. We have sized the implementation to handle estimated normal and peak usage. We anticipate no need for a separate stabilization phase.

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

The project has been very successful so far in multiple ways. We have achieved all of our stated goals to date and continue into new aspects of the project architecture. Our design allowed us to put the front end components into service while working on the back end components. We have managed to demonstrate successful rapid development (compared to historical company methods) with a high quality result. We have managed to use this highly visible success to influence several other strategic projects in the company and establish Scrum as a “best practice” recommended by the corporate technology group during an important time of global integration between two formally distinct development centers.

- 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 deliberately started with text-book scrum except that our team was smaller than the “ideal” size. The circumstances were unique and we could not control that aspect. We tried both 30 day and 15 day sprints and have decided that the shorter sprint fits better with our work style. I suspect that our small team size may relate in some way but cannot quantify it. In our one 30 day sprint, we found ourselves regrouping in the middle to remember what we meant by the tasks defined two weeks earlier.
- We had a lot of confusion in our first attempts at estimating. Since we work in pairs, we could not agree on our units of effort – were they person hours or pair hours? We settled on person hours but then later decided that they are just generic “work units” that look like person hours. After 6 sprints we now know how many of these units can be accomplished in a sprint. We also tend to end each estimation meeting by standing back, taking a deep breath and asking “do we think this array of tasks on the board looks reasonable for the sprint?”. We have made adjustments on the basis of that more general assessment.
- We evolved a task board that works really well for us. We even argue about its role as a motivator. Is it a good reminder and director of work each day? (yes for all). Is it a tyrant that is forever looking over your shoulder reminding you of your commitment to complete all tasks? (yes, for some). We do, however, stick to the concept of a sustainable pace. I describe the task board at the end of this report because it works really well.
- We debated whether the daily meeting should be sit-down or stand-up and finally settled on stand-up. People are less likely to run on when standing. We usually follow stand-up with a sit down discussion of whatever needs to be covered. We are not strict about the start time since we are all in the same room and work is never stalled waiting for the meetings. It is always before noon, though.
- We have had some trouble for lack of personal space. We work in a small bullpen. Even though there are normally only 4 people in the room, 3 of us have expressed a desire for occasional private space. We have one room for that but it evolved into a server and refrigerator room. We are building a new space and have asked for at least one private office to share.
- We were unsure about handling new tasks that appeared during the sprint. In some cases these were things we simply missed in planning. If they were blocking other tasks, then we accepted them into the sprint. In other cases they were components of larger tasks so we discussed whether to decompose and retire the original task or just reduce its estimate. We settled on the latter.
- We were not rigorous about which day the sprint ends in some cases. This was to accommodate the availability of the product owner or customer for a demo and to compensate for unscheduled absences. With a small team this was necessary and we felt that a day or two added to the sprint was not a major violation. Our last sprint actually finished two days early to suit an external schedule requirement. Team members adjusted their non-sprint activities to accommodate.

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

This came with learning. Team membership has stayed constant through 7 sprints so we have developed confidence in our ability to estimate. While we cannot articulate exactly what our work

units are, we have a very good idea of how many we can accomplish as a team in a sprint. We have been quite consistent in our sprint velocity.

We have made one major improvement in our process. For the earlier sprints we often had a number of large questions to answer regarding technologies that were new to us. We tended to describe these as “investigation” tasks and attempted to use a “time box” number rather than an estimate. We soon discovered that these tasks remained fuzzy and we never really limited the time spent on them. In recent sprints we have replaced this fuzziness with more time spent on planning day to break the fuzzy tasks down into smaller pieces that we can define better and estimate better. This has given us much better results.

I think that the accuracy matters quite a bit in making the team feel comfortable with what they have committed to. While we have achieved all of our sprint goals, including some stretch goals, the morale is better when we come in at or under an estimate that it does when someone falls into a “black hole” that exceeds the estimate. We use our other agile tools to handle the black holes – a rule to call for help if you are working alone or to take a break and/or change partners if you are pairing.

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

We do it in reverse. We have been very good at estimating. We have all remarked on this shared dynamic of doing a task-by-task estimate followed by a total work, gut level assessment of whether or not we can do it all. In no sprint have we felt compelled to work overtime to meet our sprint goals. At the same time, we have often felt that we are working at full speed for long periods to keep the task board in motion.

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?

We use the burndown chart, a simple spreadsheet table and graph. I post it in the room before each standup meeting. If the velocity looks unusual, I call attention to it and discuss. We have had some cases of high velocity when the discussion is “this is great, be wary of some pitfall ahead”. When the curve starts to flatten out we discuss the possible reasons, to see if we need to make any changes in our work patterns (usually to increase the pairing time) or consult someone outside the group with expertise we are lacking. Often the flattening is due to the identification of new tasks. In that case we discuss how vital the new task is for this sprint, what it might impact by pushing something else out or sometimes by making some other task easier.

We experimented with variations of the burndown chart to show the effect of new tasks added during the sprint. We considered backdating the estimates of these new tasks to the beginning of the sprint to show total estimated effort. We considered backdating by a single day any task that appeared and was completed in a single day. We have, in the end, kept it pretty simple and up the Scrum Master’s discretion on how best to handle these cases. In my retrospectives and as an input to planning day, I have compiled a list of initial estimates and “peak” estimates (the total estimate on day one plus the initial estimate for any added tasks) for comparison to our total at the end of each sprint plan. It has been helpful in assessing our sense of whether or not we can accomplish all that we have estimated.

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

I am working toward a career as an Agile Coach, so I need to get advance SM training and Product Owner training to have a fuller understanding of the interaction of roles.. We have many people asking for SM training so I need to keep one step ahead of them. We have many potential product owners who need to be trained. We are a global company. I have a little experience at

mastering a scrum with a bi-costal team and would like help in learning good ways to work with transatlantic teams in the near future. I am also interested in understanding how to manage interpersonal team relationships as ScrumMaster without conflicting with the duties of the functional manager in that realm.

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

We do not do full XP but are very impressed at the synergies of collocation and pair programming. We also do Test Driven Development and put a heavy emphasis on testing in general. On a small team, we would not want the added overhead of a more XP formal process but we are definitely sold on these elements of XP. If we had some reason to be more formal in an XP direction, I would want to understand that process better to evaluate what negative impact it might have along with the advertised gains. I just don't know enough about it as a total framework at this point.

Addendum

I want to describe our task board and how we use it. We have found this to be an unexpectedly delightful tool in our environment. We use a cork board and index cards. We start the sprint with a white card for each task. Each card has a task number that relates to a number on the burndown spreadsheet. Each card also has a task title. As the task is worked, the card moves on the board and each move is recorded on the card with a date and a revised estimate. Notes about the task may also be written on the card.

The board is divided into three columns marked Not Started, In Progress, and Done. It is also divided into two rows. The top, short row is the "In Box". When a card is changed in any way (moved or an estimate is revised), the card is put into the In Box of the appropriate column. The ScrumMaster is the only one who can move a card out of the In Box. This allows me to see what tasks have activity to be recorded on the burndown before moving the card.

We start with white cards. If a new task is identified that we are sure should be in the sprint, it is written on a different color card. We change the color each sprint just for variety. If a task is identified that may not be appropriate for the sprint, its card is a third color. New tasks go in the In Box of the Not Started column.

At a glance, we can see how many tasks are not started, how many are being worked and how many are finished. We can also see how many new tasks have been identified. Since our cork board is small, we use yarn for the grid lines. This allows us to adjust the column sizes as the cards move from left to right.

