

CSM Practicing Certification Renewal Assessment

Name: Paul Goddard email: paul.goddard@bt.com date: 9th March 2006

(working with Geoff Watts in BT, looking to co-teach with Ken in April 2006)

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?

Deliver a suite of web applications to serve a network operations team, as a single source of network information.

- Length - what was the duration of the project?

12 months

- Cost - what were the budgeted and actual costs?

Budget = £450K, Actual = £400K

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

Project benefits in reducing the amount of time taken to resolve a network enquiry to the control centre by rationalizing the number of systems needed to be accessed to gain information, allowed a quicker turn around on major customer enquiries.

The network control centre teams had access to a toolkit of applications which allowed a lot of other legacy systems to be switched off. This gave them quicker access to network information from a single logon through a web browser.

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

Initially there was a development team of 12 and a customer team of 8. When SCRUM was adopted this became one team of 15. One of the customers took responsibility on product owner roles to aid prioritization.

- 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 original project team structure consisted of separate customer and development teams but after we adopted scrum the team became one cross-functional team. A majority of developers were collocated, and had the benefit of scrum rooms with white boards. Although our customers were geographically dispersed we all attended our sprint planning meetings, and would regularly meet mid sprint to work collaboratively. Online web collaboration tools and audio conferences were used extensively to assist this.

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

The project was initiated 9 months prior to my arrival, and several developers had been working in isolation and producing stove-piped solutions which could not interact with each other. Consequently, a lot of the low risk, low value solutions had been developed first but having little value to the customers and users. SCRUM was introduced to both developers and customers as a means of bringing those solutions back together as a single product and ensure the highest priority features were developed first.

After the initial introduction to SCRUM and agile practices the team agreed to a trial for a month to see the benefits. Once priority from our customer was clarified this created real value in deliveries in the first 30 days, which gave us more encouragement to carry on.

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

Customers were part of planning sessions and retrospectives, and were invited to listen on daily scrums. This increased their visibility of progress which then negated the need for weekly checkpoint reports which had become an overhead. This was replaced by a weekly call with myself as Scrum Master as a more formal update. Customers could also view the sprint backlog to gauge our progress and pass on their own reports to their senior managers.

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

A clear lack of priority of the required features. Due to a number of stakeholders requiring so many features, the team had a lack of direction and consequently a lack of understanding about the overall goal/theme. These were clarified with a whiteboard session in Brussels with the product owner, who clearly explained most valuable features, and this then gave the team an agreed target.

We also struggled with a not having a product owner to start with. This was resolved in Brussels by our overall product owner and budget holder, and his empowered customer proxy was appointed to help us.

- 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 actually the development delivery manager before I took on the role of ScrumMaster. The product owner role was the budget holder and due to his lack of availability throughout the project, one of his direct reports acted as empowered customer proxy. This gave the team a clearer vision of our customers overall goal, and which features were his highest priority and delivered most value.

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

The development up until we adopted scrum had evolved into isolated stove-piped solutions, so we had to bring the developers together and work as a closer team to design a more cohesive system. Pairing was promoted to spread domain knowledge across the team and also produce better quality code. The team were also introduced to time-boxed planning sessions, as opposed to submitting wildly ambitious project plan documents.

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

We had 2 sprints before we deployed our software to the network control centre teams, but after that we were able to release the highest priority features every 30 days thereafter.

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

When we adopted Scrum the project was being developed in stove-pipes and in most cases the simplest lowest value features had been developed but never deployed. Adopting Scrum changed our focus and allowed us to make big impact deliveries fast. The customer was overjoyed with the changes we made in the first 3 sprints, and this reinforced the whole teams belief in the benefits Scrum brought to the project.

- 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 very much wanted to use Scrum 'out of the box' but several barriers were in our way. We did fundamentally change to a time-boxed planning approach which was met with some resistance from our customers, mainly as it was so difficult for them to form a consensus on priorities. Also, we didn't have the luxury of a co-located team so we had to make use of audio and web conferencing instead. Developers were also uneasy about estimating, but their confidence and accuracy increased with time.

However, once we found a rhythm and people saw the benefits of features being produced every 30 days were we able to celebrate the success of adopting Scrum and continue to evolve with it.

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

Initially the teams estimates were either wildly exaggerated or extremely optimistic. But this was a lack of confidence and the false understanding that they had to fit everything on the backlog into their 30 day sprint. Once the process was more understood and developers had some experience of estimating, these estimates became more accurate. Identifying similarities in product backlog items (size and complexity) also allowed the teams skills of estimating to increase.

The accuracy of these estimates was not as important as the culture of estimating itself. The team had been driven by dates picked by management for a long time, and the change in mindset of estimating their own velocity was the greatest benefit.

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

Estimating will improve, sprint after sprint. But initially, we knew our sprint commitment would probably not match our actual delivery. As such, we had to set expectations to our customers and we tried to do that by integrating them as part of the team. The increase in collaboration exposed the issues immediately, and the customers understood those barriers at their inception. As our accuracy for commitment improved, the level of confidence with our customers improved with it.

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?

A multitude of complex, incomprehensible project plans were eventually replaced by a single excel spreadsheet, containing the product backlog items as well as a sprint backlog for the current sprint. The team would update this daily from central repository as well as communicating progress on the daily scrums. From that we could see the amount of work remaining in the sprint and deduce our progress on a daily basis.

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

I think the hardest thing to achieve is the change in mindset. Increasing communication and collaboration was the hardest hurdle to overcome. I think those areas need to be tackled first, and then introduction to product backlogs (with templates) and estimating backlog items are very much a 'trial and error' practice.

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