



Scrum of Scrum Approach in Agile Development

Bharat Chandra Joshi*
Information Technology, YMCA,
India

Kapil Dev
Computer Science & Engineering,
India

Dr. Naresh Chauchan
Computer Science and Technology,
India

Abstract— Agile software development is a conceptual framework for software engineering that promotes development in iterations throughout the life-cycle of the project. Software developed during one unit of time is referred to as an iteration, which may last from one to four week. Scrum is part of the Agile movement. It is a lightweight process framework for agile development. The Scrum method is most often used to manage complex software and product development, using iterative and incremental practices. And results have been shown about significantly increase in productivity and reduces time to benefits relative to classic “waterfall” processes. But scrum method also have some problems and to overcome the problems a new approach is proposed that is Scrum of Scrum. A Scrum-of-Scrums is a way to scale a Scrum organization. It consists of various Scrum teams that use a Scrum-like process to align with each other.

Keywords— Scrum, Scrum of Scrum , Agile Methodology

I. INTRODUCTION

Software engineering is a step by step procedure to produce a good quality product at the end. It provides different ways to implement the software with the help of procedure which is called as software development methodology [1]. There are various software development methodology like waterfall model spiral and Agile. Agile Project Management is one of the revolutionary methods introduced for the practice of project development and management. In 1970, Dr. Winston Royce asserted that software should not be developed like an automobile on an assembly line, in which each piece is added in sequential phases[2]. Dr. Royce specifically objected to this approach due to the lack of communication between the specialized groups that complete each phase of work. But this research is focused on scrum which is a method itself in agile methodology.

Scrum methodology: Scrum was developed by Jeff Sutherland in 1993. Scrum is an iterative, incremental framework for projects and product or application development [1]. In scrum development takes place in cycles of work called sprints. These iterations are not more than 2 weeks and take place one after another. The sprints are time boxed—means that they end on the specific date in spite of the work completion. During the sprint there cannot be any change in the development process. At the end of each sprint we get a deliverable product. In scrum there is a team of 7 members consisting of Scrum master as a major role. The scrum master helps the team for any issues and help to achieve business value.

This paper presents the new approach in scrum of scrum methodology. How scrum of scrum methodology can be used in large projects consist of more than 100 team members.

II. RELATED WORK

According to Igor Riberio Lima Software development using agile methodologies is becoming a bigger reality in software development companies [3]. Agility brings quality to the software development and management process. To provide the quality value to the final software, one must have a well structured team that follows the methodology and uses correct strategies. The adoption of methodology such as Scrum may help to establish a bridge between quality and products that effectively solve problems of the business world.

Scrum brings an incremental and iterative development process for agile software development and management that stands on several points:

- i) Individuals and their iterations are more important than procedures and tools
- ii) Working software is more important than a complete documentation
- iii) Collaborating with clients is superior to contract negotiation
- iv) The ability to respond to change is more important than having a pre-established plan.

Teams are usually composed of four to seven members, which makes communication easier. An important adaptation is the inversion of the semantic of the product owner, for in their context he is a member of the lab allocated at the client. This change was made because of the difficulties associated with having a client in the lab.

Based on the analysis of the above implantation of Scrum for project development, we could see the change in software projects management and development, allowing for easier imagination of progress. The involvement and commitment of members of the team with the results increased, allow for more collaborative work.

He also realized that team members were motivated and open to the changing requirement, which facilitated the process adapting Scrum. But we also need to use metric to evaluate formally the progress achieved by using the agile scrum methodology.

According to "Scientist name"

There are several issues identified from survey results and interview sessions. These issues are directly effecting the SCRUM implementation product development and release process. These are the following issues:

- I) Quality item pileup: Due to agility team have some obligation to develop a deliverable product at the end of each sprint regardless of the scope in sprint planning. Because of this sometime team ignores the quality of product that can effect the customer. This is one of the biggest issue with the agile methodology SCRUM method.
- II) Module integration Issue: As the product releases in each iterations thus its become an arduous task to integrate the modules into a complete product. It also requires rigorous testing for quality assurance.

III. INTRODUCTION TO SCRUM OF SCRUM

A Scrum-of-Scrums is a way to scale a Scrum organization. It consists of various Scrum teams that use a Scrum-like process to align with each other. In Scrum of Scrums, the team is divided into two or more teams, upto the limit of nine people per team. Suppose a project consists of 1000 tasks (400 independent tasks, 600 dependent tasks). A sprint generally consists of dependent and independent tasks to give a shippable product at the end. Also 400 tasks in a sprint is not easy to handle by 9 to 10 people as a sprint cannot have more than 9 to 10 people due to daily scrum meeting. Thus a shippable product will not be achieved due to a sprint consisting of only independent tasks. So, parallel sprints to be required which are achieved by dividing the project into sub-projects. The subprojects are going to execute parallel and in each subproject there are several sprints (dependent within a subproject, but independent of other subprojects) which is going to execute sequentially. It is a technique to scale Scrum up to large groups (over a dozen people we can say), consisting of dividing the groups into Agile teams of 5-10. In daily scrum a member from the team designated as one member as "main person" to participate in a daily meeting with main person from other teams, called the Scrum of Scrums. Depending on the context, main person may be a technical contributors, or each team's Scrum Master, or even managers of each team. Scrum of Scrums proceeds as a normal daily meeting, with main person reporting completions, next steps and impediments on behalf of the teams they represent. Also known as a "meta Scrum".

IV. PROPOSED APPROACH

Agile only applies to small projects now a days. Some problems that make Agile adoption impractical

PROBLEM 1: Starting with the size of the team: According to Scrum, a team has seven people (plus or minus two). So, the question arise we use Scrum in a large project with a team limited to nine people? [4]

PROBLEM 2- How do we make a daily stand-up meeting with a team of 100 people? It is impossible! To keep to the 15-minute time-box, each person would have only 9 seconds to speak.

PROBLEM 3- The third problem is how to measure the team performance. we can measure the average productivity of a team of 7 people, but what about a team of 100? It would be very difficult to measure the performance of the team of this size.

PROBLEM4- Consider further the work of the Project officer. How can a single person sanitize, priorities and detail of a product which have a large backlog, and ready for sprint meeting and planning? To work with a backlog would be a full-time job.

SOLUTIONS TO THE PROBLEM:- The solutions is, Scrum of Scrums

In Scrum of Scrums, we divide the team into several team and each team has eight or nine person. Let us take an example so that we can easily understand, suppose we have a project and our team have 60 people. In this case, we now have 3 teams of 20 people each.

To divide our team is good strategy, but if all these groups work on a same product, we will face a lot of problem, some of these are: [5]

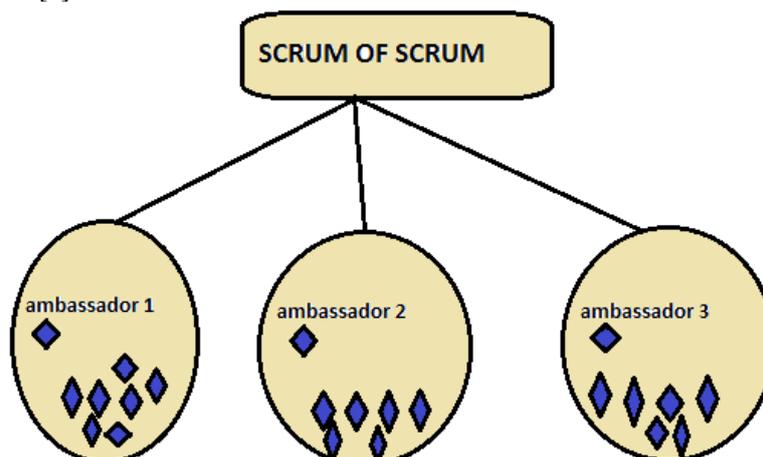


Figure 1

- i) Lack of a unified view of the product/service
 - ii) Redundancy of work (two teams implementing the same part of the scope)
 - iii) Communication failure
 - iv) Integration hell (how to integrate different parts of a product developed by different teams)
 - v) Dependencies between tasks of different teams
 - vi) Complex change management
- The first technique is the replication of important roles in each of the teams like as the PO, Scrum Master, and other technical leads.
 - Another technique is the alignment of iterations. Team composed of individual role based persons like a tester, developer and solution designer. Each team has a specific velocity, production capacity, and learning curve. In this case, aligning the start and the end of release iteration in a time-box makes the teams free to perform as much iteration as they need to complete the development of the product. The difference is that the teams are aligned in the project milestones; they must finish all the iteration together.
 - Another technique is to hold staggered daily meetings. It is like sequencing the daily meetings of different team with the scrum master as the common chair person so that representatives of each team can potentially participate for their status, and to identify dependencies and risks and to find specific skills.
 - Another technique, Scrum of Scrums is a generic model which could be applied to other levels of programs/project/portfolios, according to our need. For example we take a project of 50 people. In this case we would have 5 teams of 10 people, however 5 teams are more than enough the recommended number of 9. Then we would create another team taken to a higher level, creating a Scrum of Scrums of Scrums to monitor the teams at below level. Each team holds a meeting of fifteen minutes every day. One or two representatives from each team of scrum participate a higher level Scrum for coordinating overall teams. And for large teams, one or two representatives of higher level Scrum team attend an even higher level Scrum team.

SOLUTION APPROACH – the ideal solution is co-locating the team. Other solutions to solve the problem:

- Restructure the teams to form relatively smaller teams so daily scrum meetings can be completed in 10-15 minutes
- Future - divide feature teams into layer-wise or component-wise, considering the co-location of team members. Co-located team members can rapidly resolve inter-dependencies within the same team. Other inter-team dependencies can be raised and resolved either by the facilitator or an open office meeting on the same day. [6]
- Introduce one facilitator for every team so as to make on the spot resolution of quarry (the facilitator has development background business knowledge). This person may have to facilitate multiple teams, as the facilitator doesn't do actual development
- Facilitator re-orders the priority of defects in sprint on a daily basis
- Introduce daily open office meeting with architects and product owners to discuss big impact issues/show stopper issues. These are issues which could not be resolved by the facilitator. Anybody can join this meeting and raise a concern.
- Introduce a time-zone-wise testing server for sanity testing before promoting the daily build to the main testing server. It helps avoid high severity issues that can impact large numbers of people.

V. CONCLUSION

The goal of this research was to systemize and describe the existing stage of “the new agile process for remote projects” and propose suggestions for its modification. The first objective of the research was to systemize the practices, phases and other attributes of the process and to describe the new agile process for remote projects as precisely as possible. The second objective of the study was to identify issues causing negative effect on the process. Comparison with other software development processes and analysis of the project experience revealed several leaks and weak points of the process. The third objective was actual improvement of the process. According to the detected weak points several improvements were proposed. They took different forms such as extra roles on the process, additional practices or activities which were not performed (or performed on insufficient level before).

As it was mentioned above the research resulted introduction of several best practices and additional roles. It was found that knowledge transfer as initial and ongoing activity speeds up the development process. Traceable hierarchy practice helps to avoid involvement of indifferent team members in communication within interested party. As well it assists to reduction of misunderstanding and to improvement of transparency between the sites. Direct communication empowers the developers to contact with onsite team for any unclear requirements without wasting time. Daily meetings encourage the developers to communicate actively with others and keep track of the progress.

It was found that additional roles like proxy customer, (person acting as the customer for the development team) and cultural liaison (person performing the link between two sites levelling the cultural differences and language barriers) would bring benefits to the process. The proposed practices and additional roles are suggested based on key findings in the process description and the implemented project documentation. Ideas and approaches are based on the agile principles and the agile manifesto and taking into account the issues of distributed global software development

ACKNOWLEDGEMENT

I express my sincere and deep gratitude to my guide Mrs. Rashmi popli and Dr Naresh Chauchan Department of Computer Science & Engineering , YMCA university of science and technology, for their guidance, support and encouragement.

REFERENCES

- [1] The scrum paper available at [http://www.hbagency.com/scrum/tl_files/scrum_inc/documents/ScrumPapers.pdf]
- [2] The Agile movement avialble at [<http://agilemethodology.org/>].
- [3] Adapting and Using Scrum in a Software Research and Development Laboratory Igor Riberio by Igor Riberio Lima and Tiago de Castro Freire 2012.
- [4] Scrum: Developed and sustained by Ken Schwaber and Jeff Sutherland;scrum.org; February 2010.
- [5] Bruce Powel Douglass;ROPES:Rapid Object-Oriented Process for Embedded Systems;I-LogixInc;adapted from Doing Hard Time:Developing Real-Time Systems using UML, Objects, Frameworks, and Patterns Reading, MA: Addison-Wesley, 1999.
- [6] Eduardo Miranda; Combining Critical Chain Planning and Incremental Development in Software Projects; originally published as a part of 2004 PMI Global Congress Proceedings – Europe.