

# Communication Issues in Agile Software Development

Salvador Esquivel<sup>1</sup>

<sup>1</sup> Facultad de Informática,  
Universidad Nacional de La Plata.  
salvadoresquivel@hotmail.com

**Abstract.** Scrum framework is a growing trend in software industry to companies that are looking for development agile ways. In their early days, this methodology required that the working team members were established in a unique room, because it is necessary to have great communication and working together. However, it is a common increasingly practice that the teams are in geographically dispersed places, which means that it is necessary to adapt and/or look for the way that this methodology is suitable in these contexts. There are many experiences in the industry where we can see communication issues due to this kind work. In this paper we will briefly explain the Scrum framework definition, the Global Software Development (GSD) context, and the practices used in different case studies to solve the issues when applying Scrum in GSD.

**Keywords:** Scrum, Global software Development, Communication issues.

## 1 Introduction

Scrum framework is a growing trend in software industry to companies that are looking for development agile ways. In their early days, this methodology required that the working team members were established in a unique room, working face to face [1], because it is necessary to have great communication and working together.

However, it is a common increasingly practice that companies applies GSD in their software factories, and still require to continue getting the benefits achieved by agile methodologies, such as reducing the "time to market" and greater flexibility in projects, so they need to look for the way that this methodology is suitable in these contexts.

There are many experiences in the industry where we can see communication issues due to this kind work, and the different alternative solutions that were implemented, leading to analyze the results obtained with this way of increasingly expanded work worldwide.

The scope of the document includes the review of those case studies that describe the problems of communication and the different results to solve them while using

Scrum framework within the context of "Global Software Development" (GSD), and will focus in obtaining the different experiences in the industry in this context.

The remainder of this paper is organized as follows:

Chapter 2: the Scrum framework, characteristics and concepts: roles, events and deliverables.

Chapter 3: the different aspects of GSD.

Chapter 4: the different possibilities in which the Scrum methodology is used in projects with GSD structure are described, and the results of the analysis of the documentation for these cases studies are obtained.

Chapter 5: preliminary conclusions as a result of the description of the previous chapter are formulated and research or work that emerges as consequence of this study is mentioned.

## 1.1 Research method

In a first step, an Internet research was done taking into account the resulting information from the words "Scrum", "Agile" and "Global Software Development" using [www.google.com](http://www.google.com) search engine.

For each of all the papers that were obtained, all their references were recursively searched using <https://scholar.google.com/> search engine.

Then, those articles where Scrum was not in the body of them were discarded.

The remained articles were reviewed, and classified considering the purpose of them in:

"Case studies"

"Description of detected issues"

"Framework for investigation – Existing Literature Review"

From these articles, those that referred specifically to show communication issues in the case experience were selected, reaching the amount of 17 papers that are used to review the experience of the industry to the issue we are evaluating.

## 2 Scrum

According to the Scrum Guide [2], Scrum is a "framework within which people can address complex adaptive problems, while productively and creatively delivering products of the highest possible value.

Scrum is not a process or a technique for building products; rather, it is a framework within which you can employ various processes and techniques. Scrum makes clear the relative efficacy of your product management and development practices so that you can improve."

It was promoted as a way of reducing time to market, increasing productivity, improving quality and gaining cost effectiveness and efficiency, and has gained significant popularity because of a promise to handle requirements volatility throughout the development life cycle, promotion of extensive collaboration between customers and developers, and support for frequent delivery of a product [3].

The following roles are defined in Scrum:

The *Product Owner* (PO): PO is the customer representative in the Development team. He is responsible for defining the priorities for implementing the various defined requirements, responsible for these definitions, and answers the different questions about them to the team. He is responsible for approving the product increases shown in the sprint review meeting.

The *Scrum Master* (SM): SM is responsible for ensuring the proper implementation of Scrum processes. He is there to serve to resolve all the obstacles that may arise under the project. This role is the closest to the project manager one in cascade method, but with a completely different profile, not taking any decisions or assignments. This role change is one of the most difficult to take when transitioning from cascade development scheme to Scrum one.

*Team*: It is responsible for the development of the increment of the product. It is self-organized; the team in the planning meeting is who defines responsibilities, estimates and which the outcome of the sprint will be, following the priorities defined by the PO. It refines the requirement definitions with the SM and PO in the Product Backlog Refinement meetings. During execution of the sprint, it meets daily to synchronize efforts and raises all the impediments to SM. It is necessary to team members to have an active participation in the project and become owners of it to be successful, and this is a complex requirement to achieve, considering the natural introvert characteristics of most professionals in software engineering, where they often wait to have their tasks assigned by the manager to perform them.

The defined events in Scrum are:

*Sprint Planning*: It is divided in two steps: In the first one, the Team analyzes and selects the requirements that will be developed in the sprint. The Team negotiates with the PO which items will be committed to be delivered by the end of the sprint; the “Sprint Goal” is defined. In the second step, the Team produces the list of the necessary tasks to develop the selected requirements and the members self- assigns them. This way, the “sprint backlog” is defined as a result of this process, which is the base to be used in the sprint daily meetings.

*Daily meeting*: It is a meeting to synchronize efforts that lasts at least fifteen minutes, where each member of the Team responds to the following questions:

¿What did I do from the last meeting?

¿What am I going to do until the next meeting?

¿Do I see any impediment that prevents me or the Development Team from meeting the Sprint Goal?

*Sprint Review*: At the end of the sprint, the Team shows the developed software increment. It could have a preliminary presentation, which is valid when the attendants need to have some kind of introduction to the feature, to understand what is going to be shown.

Then, the participants give suggestions and improvements that the PO takes into account to add or not in the pending product backlog to prioritize in future sprints.

Finally, the PO approves or rejects what the Team has developed.

*Sprint Retrospective*: The Team analyzes the way of working and the different issues in order to increase the productivity.

*Product Backlog Refinement*: The Team and the PO analyze the product backlog items that will be developed in future sprints, with the objective of having them clearly defined by the time the sprint planning meeting.

The Scrum artifacts are:

*Product Backlog:* It is a prioritized list of features defined by the PO. It is reviewed in the Product Backlog Refinement meeting and it is used as input in the sprint planning meeting, where, depending of the priority and the needed volume of work, the Team defines a part of this list as the objective of the sprint.

*Sprint Backlog:* It is the subset of the product backlog that the Team commits to develop in the sprint, along with all the needed tasks to manage. It is created in the sprint planning meeting, and the Team is responsible of it. It is reviewed in the daily meetings, and it is possible to add new tasks as soon as they are discovered, and also to eliminate those that are no longer necessary.

*Sprint Increment:* The different software components developed during the sprint that are potentially ready to deliver and that are shown in the sprint review meeting in a demo.

### 3 Global Software Development

According to E. Carmel [4], “the list of features that distinguish global software teams from normal (nonglobal) software teams is short and precise:

- Distance (the distance of developers from each other and from their customers or end-users):

Distance impacts the communication between designer and customer, between two developers, and between development teams and their remote managers. Communication constrains become increasingly significant for software development teams, whether the team’s sites are in the same metropolitan area, within the same country, or cross-ocean.

Distance affects all sorts of coordination and control.

Distance forces most communications into electronic pipelines of various widths and colors. These pipelines are not as rich as face-to-face communication.

- Time zone differences:

These differences exacerbate the communication problem. Almost all the communication is channeled through various asynchronous technologies, such as e-mail or formalized work flow arrangements.

- Local culture (including language, national traditions, customs and norms of behavior).

Perhaps it is the most confusing intra-team feature. It encompasses national and ethnic traditions, customs, norms of culture, as well as language. Cross-culture teams have more potential for productivity as well as more potential for problems, related to that of more homogeneous cultural groups. Problems may stem from mistrust, miscommunications, and lack of cohesion.”

Accordingly, in [5] it is stated that virtual teams :

- Reduce costs by cutting travel expenses and time, creating new "e-economies" of scale, and designing better digital processes.
- Reduce the development cycle evolving from process in series to in parallel, establishing better communications and generating more widespread confidence.

- Increase motivation, allowing more diverse participation, stimulating creativity and encouraging processes and products synergy of new business development
- Promotes learning by capturing knowledge in the natural course of work , and by sharing best practices

Dean Leffingwell [6] states that in scale, all development is distributed development; the opportunity to communicate informally decreases with distance : people located 100 meters or more, has only 5 % chance to talk to each other. By increasing the size of the team, working in the same field it is not a practical reality for most companies. While companies with lots of developers are in a unique city, the size of the organization is such that people are not in the same workspace, the same floor, the same building or the same campus.

In the case of multinational corporations, with teams located in multiple countries, increasing trend of outsourcing, the problem gets more severe.

Related to home office, in [7] it states that the difference from other forms of distributed models , the group is individually and fully distributed , while in other cases small groups are integrated in the same physical place. An advantage of home office is as following: decreasing population of large cities, economizing high cost office space, and giving a chance to the people who are not able to work outside. Nevertheless, the disadvantage of home office includes problems such as burden from less communication, and an exclusion from society. In the software development field, Home Office environments became popular with the growth of Open Source Communities, where people contribute in their free time, from different locations, to build even powerful systems.

#### 4 Scrum in GSD

Most of the success of Scrum is that the team members are located in the same geographic area [8], and the interaction between team members in GSD is becoming difficult. The agile community advocates the importance of the proximity and the relationship between team members [9], which presumes that the implementation of these concepts will be complicated.

The practice of Scrum in these environments is becoming increasingly extensive, even several people argue [9] that the careful use of agile practices in globalized projects and distributed can provide a number of benefits in terms of communication, improved productivity, project management, trust, team motivation , project visibility.

The following models of distributed Scrum teams are proposed:[5]

- Isolated: teams are isolated scattered in different geographies without contact with each other
- Distributed Scrum of Scrum: the teams are isolated and integrated through regular Scrum of Scrum meetings
- Fully distributed Scrum: the teams have their members scattered across geographies.

In this third division we might consider two different aspects:

- A first consideration where, although team members are well dispersed, those who are in a particular place are located in the same office, i.e. attending a common space to work.

- A second consideration where their members, totally or partially, are working remotely from their homes ("home office"), without any other contact except the "virtual", or via the Internet, telephone, etc.

In these contexts, we see the different experiences related to communication problems documented in the case studies of the articles analyzed and the actions taken to solve them:

#### 4.1 Issues

In one of the projects that are described in [10], the meetings should perform using telephone and video conference, forcing to one team to be late evening/night in the office, because there is not enough broadband at members' home.

The lack of face to face communication is a recurring problem in different cases : [10], [11], [12], [9], [13], [14], [15], [1], [16], [17], [21]. Even there is a great difficulty in knowing who is talking as they cannot see the faces [18], [19].

To [8], the reduction of informal contact could lead to lack of awareness of the criticality of the different tasks.

To [15], relationship problems between team members are evident, and they make the mistake of not spreading all the information to the remote Team with the purpose of simplifying, leading to low morale and sense of frustration. It is also detected that when increasing the number of members of the teams, communication becomes more complex, and the setting up of teams if they are not multidisciplinary, is making impossible for them to take ownership of the project.

To [20], video conference is not always available, the conference calls are not good due to the lack of face-to-face communication. The telephone system doesn't work properly.

To [22], even informal communication is difficult, they find different expectations and assumptions related to how to implement Scrum due to the cultural barriers and the language.

In [23], is reported that phone discussions between two sub teams don't work, and takes a lot of time communicate and interact.

To [24], daily meetings become a problem.

In [23] also, they say that conference calls are not understood due to language, so they were replaced by chat and e-mail.

Finally, to [22], they raised expectation and language issues in management meetings.

**Table 1.** Summary issues:

Issue	# of cases
not enough broadband at members' home	1
Lack of face to face communication	11
Not knowing who is talking	2
Lack of awareness of critically of tasks	1
Not spreading all the information to remote team	1
Video conference tool not available	1
Informal communication difficult	1

Conference calls not understood/don't work	1
Language issues	2
Daily meetings become a problem	1
Relationship problems	1
Increasing the number of members	1

## 4.2 Solutions

The ways to try to solve these issues are diverse:

To [10], it is necessary to make the team travel from one site to another and vice versa, to develop a whole sprint, trying to have local meetings whenever possible; on the other hand they defined short sprints, having review meetings all together, but splitting retrospective ones, and then having coaches meetings.

Similarly,[16] tried to generate personal relationships , moving offshore developers to onshore, and vice versa, temporarily.

To [9], the dynamic of the Scrum events, daily meetings reviews, retrospectives, refining, help or force to reduce this problem, while increasing the visibility, the communication between the stakeholders, the project coordination.

In [13], they try teams to work in the same physical space whenever possible, even to the extreme that if there are no offices, meet in coffee shops, homes or meeting rooms. In any case, it is essential to use technology for web meetings.

In[14], they use videoconference rooms for daily meetings, and developed a digital board (add on of Jira).

In [15], ambassadors were implemented to improve the communication between the different places, generate trust and let transfer knowledge.

The essential communication problem in [1] is in terms of understanding the requirements, so they decided to perform design and requirement workshops.

In [20], they use a tool called web demo, also they use instant messaging.

It is defined to reduce the frequency of daily meetings to three times a week in [21], with the possibility of doing them at home due to the time differences, and trying to have hour overlapping those days.

They decide to implement the co-SM role in [22], as well as an increment of flights between the sites; additionally, they replaced the onshore architect with an offshore one that was moved onshore during the project to improve the communication. For the second case of the same paper, people onshore are more time offshore and they replace the local coordinator with the on site manager.

In [23], the proposal is to delegate SM role in a person from India that travels to Norway for one sprint. They cancel daily meetings. They divide the requirements in modules to reduce the interdependencies. They begin specifying and documenting the product backlog, and they leave the details to the Team to decide.

In [24], they simply decide to use communication and conference tools while having SM controlling and managing the meetings.

And in [22], it was decided to perform an initial training related to Agile language and methodology to managers.

**Table 2.** Summary solutions

Issue	# of cases
Members travel from one site to another	2
Short sprints	1
Dynamic of Scrum events	1
Make members work in same place	1
Use video conference room or communication tools	2
Ambassadors	1
Reduce frequency of daily meetings	1
Co-SM role	1
Training	1
Delegate SM role in offshore site	1
Reduce interdependencies between sites	1
Design and requirements workshops	1

## 5 Conclusions

- Scrum framework immediately shows communication problems that are typical in these contexts.
- Scrum provides the procedural tools to detect and solve them, for example through retrospectives meetings, while projects are active.
- The ways of resolution are extremely varied and depend on the different contexts in which projects operate.
- Several drawbacks often try to be solved through the travel of team members between different locations, either "onshore" to "offshore" or vice versa, strongly constrained by the project budget and result is not always possible.
- Need for communication requires the availability of a technological infrastructure to provide web conference services, good sound quality teleconferencing, environments with tools to share information, which do not have or are limited in many opportunities, especially when working with equipment in emerging countries in Asia or Latin America.
- Most cases are referred to an "onshore" group working with another "offshore" team.
- There are no cases where part or all of the members of each team are in the form of "home office", where entirely all informal face to face communication is absent.
- Many papers claim that the framework itself solves or alleviates the inherent GSD problems.

As a result of this study, the author considers the topic of Scrum in the form of total "home office" as an important line of research, commonly called "virtual teams", where there is no personal face to face interaction between development team members.



## 06 References

1. Paasivaara M, Heikkilä VT, Lassenius C. Experiences in scaling the product owner role in large-scale globally distributed scrum. In: Global Software Engineering (ICGSE), 2012 IEEE Seventh International Conference on. IEEE; p. 174–8. (2012)
2. K. Schwaber and J. Sutherland, “The Scrum Guide - Scrum-Guide-US.pdf.”, <http://www.scrumguides.org/docs/scrumguide/v1/Scrum-Guide-US.pdf#zoom=100>.
3. Hossain E, Babar MA, Verner J. Towards a framework for using agile approaches in global software development. In: Product-Focused Software Process Improvement. Springer; p. 126–40.(2009)
4. Carmel E.: Global Software Teams. Collaborating Across Borders and Time Zones. Prentice Hall PTR; 269 p. (1999)
5. Lipnack J, Stamps J.: Virtual teams: people working across boundaries with technology. 2nd ed. New York: Wiley; 317 p.(2000)
6. Leffingwell D.: Scaling software agility: best practices for large enterprises. Upper Saddle River, NJ: Addison-Wesley; 349 p. (The Agile software development series).(2007)
7. Luz M, Gazineu D, Teófilo M.: Challenges on adopting scrum for distributed teams in home office environments. World Acad Sci Eng Technol.;59:308–11.(2009)
8. Bannerman PL, Hossain E, Jeffery R.: Scrum practice mitigation of global software development coordination challenges: A distinctive advantage? In: System Science (HICSS), 2012 45th Hawaii International Conference on. IEEE; p. 5309–18.(2012)
9. Hossain E, Babar MA, Verner J.: How Can Agile Practices Minimize Global Software Development Co-ordination Risks? In: O’Connor RV, Baddoo N, Gallego JC, Muslera RR, Smolander K, Messnarz R, editors. Software Process Improvement. Springer Berlin Heidelberg; p. 81–92. (Communications in Computer and Information Science). (2009)
10. Drummond B, Unson JF.: Yahoo! Distributed Agile: Notes from the world over. In: Agile, AGILE’08 Conference. IEEE; 2008. p. 315–21.(2008)
11. Cristal M, Wildt D, Prikladnicki R.: Usage of Scrum practices within a global company. In: Global Software Engineering, 2008 ICGSE 2008 IEEE International Conference on. IEEE; p. 222–6.(2008)

12. Jensen B, Zilmer A.: Cross-continent development using Scrum and XP. In: Extreme Programming and Agile Processes in Software Engineering. Springer; p. 146–53.(2003)
13. Karsten P, Cannizzo F.: The creation of a distributed agile team. In: Agile Processes in Software Engineering and Extreme Programming. Springer; p. 235–9. (2007)
14. Tietz V, Mönch A.: Facing Fake-to-Fake: Lessons Learned from Distributed Scrum. (2015)
15. Hogan B.: Lessons learned from an extremely distributed project. In: Agile Conference, 2006. IEEE; p. 6 – pp.(2006)
16. Summers M.: Insights into an agile adventure with offshore partners. In: Agile, 2008 AGILE'08 Conference. IEEE; p. 333–8.(2008)
17. Kilpala M, Kärkkäinen T.: Distributed scrum when turning into maintenance: A single case study. In: Second International Conference on Computer Science and Information Technology (COSIT 2015) [Internet]. [cited 2016 Feb 3]. p. 55–67. Available from: <http://airccj.org/CSCP/vol5/csit53706.pdf> (2015)
18. Paasivaara M, Durasiewicz S, Lassenius C.: Using scrum in a globally distributed project: a case study. *Softw Process Improv Pract.* 13(6):527–44.(2008)
19. Paasivaara M, Durasiewicz S, Lassenius C.: Using scrum in distributed agile development: A multiple case study. In: Global Software Engineering, 2009 ICGSE 2009 Fourth IEEE International Conference on [Internet]. IEEE; [cited 2016 Feb 3]. p. 195–204. Available from: [http://ieeexplore.ieee.org/xpls/abs\\_all.jsp?arnumber=5196933](http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5196933) (2009)
20. Cho J.: Distributed Scrum for large-scale and mission-critical projects. *AMCIS 2007 Proc.* ;235.(2007)
21. Vax M, Michaud S.: Distributed Agile: Growing a practice together. In: Agile, 2008 AGILE'08 Conference [Internet]. IEEE; [cited 2016 Feb 3]. p. 310–4. Available from: [http://ieeexplore.ieee.org/xpls/abs\\_all.jsp?arnumber=4599497](http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4599497)(2008)
22. Sadun C.: Scrum and Global Delivery: Pitfalls and Lessons Learned. In: *Agility Across Time and Space* [Internet]. Springer; [cited 2016 Feb 3]. p. 71–89. Available from: [http://link.springer.com/chapter/10.1007/978-3-642-12442-6\\_5](http://link.springer.com/chapter/10.1007/978-3-642-12442-6_5) (2010)
23. Hole S, Moe NB.: A case study of coordination in distributed agile software development. In: *Software process improvement* [Internet]. Springer; [cited 2016 Feb 3]. p. 189–200. Available from: [http://link.springer.com/chapter/10.1007/978-3-540-85936-9\\_17](http://link.springer.com/chapter/10.1007/978-3-540-85936-9_17) (2008)
24. Chalegre V gínia C, Santos WB, de Souza LO, Muñoz HJ., Romero de Lemos Meir a S.: Estudo de Caso da Utilização de Scrum no Desenvolvimento Distribuído de Software. In *PUC - RS*; p. 129–36.(2010)