

**Łukasz D. Sienkiewicz**

Wrocław University of Economics  
e-mail: lukasz.sienkiewicz@ue.wroc.pl

---

## COLLABORATION BETWEEN THE SCRUM AND THIRD PARTY SERVICES IN THE NETWORK ORGANIZATION<sup>1</sup>

---

**Abstract:** Many scientific publications demonstrate the Scrum as the best method for managing software development. Nevertheless, the original Scrum method is not sufficient for managing work in the network organization where stakeholders may know nothing about the Scrum or even Agile. In this paper, we use the 3C model to determine how the teamwork of third party services and other Scrum players looks. We believe that will help in finding a more satisfactory approach for managing software development in a strongly distributed environment (i.e. network organization).

**Key words:** 3C model, collaboration, cooperation, communication, coordination, Scrum, third party service, network organization, Agile, software developments.

### 1. Introduction

We believe that the success of the project in many cases depends on the proper selection of the software development method [Guntamukkala, Wen, Tam 2006]. This is very important especially in a Network Organization where a distributed environment brings the threat that other participants will not respect the Scrum rules.

In this paper, we analyze the Scrum and Kanban as Agile methods introduced in the network organization:

- We refer to the network organization and determine how the third party services collaborate in an Agile environment.
- We use the 3C model (i.e. Communication, Coordination, Cooperation) [Lucena et al. 2008] to classify collaboration, with an emphasis on the teamwork types between Scrum roles.
- We believe that the improvements proposed in this paper will help in the better understanding of the software development method in network organizations and will be the basis for more empirical research.

---

<sup>1</sup> Selected parts of this article were published under nonexclusive copyright in FEDCSIS'2011 proceedings [Sienkiewicz, Maciaszek 2011].

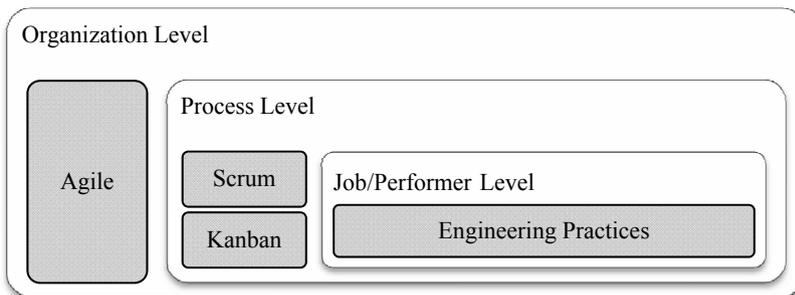
## 2. Network organization

We consider the network organization as a pattern of social relations over a set of persons, positions, groups, or organizations [Sailer 1978; Van Alstyne 1997] with long-term arrangements between related for-profit organizations (i.e. firms), in order to gain or sustain a competitive advantage [Jarillo 1988] and to facilitate goal-oriented processes and activities [Van Alstyne 1997].

From our point of view, it is very important to point out that participants of the network organization might be another network organization or organizations.

### 2.1. The relationship between Agile, Kanban and Scrum

In order to highlight the relationship between the impact of Agile, engineering practices, Kanban and Scrum in the network organization, we can use the “Three Level Framework” [Rummler, Brache 1995]. The framework takes the Scrum viewpoint and distinguishes between three types of layers introduced in [Sienkiewicz, Maciaszek 2011] and is presented in Figure 1.



**Figure 1.** Relationship between Agile, engineering practices, Kanban and Scrum in terms of “Three Level Framework”

Source: own elaboration based on [Rummler, Brache 1995].

By the Agile environment on the organization level we understand all the activities which are additional to Scrum or Kanban (e.g. HR, financial, accounting, capability management, etc.) and relate to the organizational point of view (i.e. market, competitive advantage, priorities, products and services) [Sienkiewicz, Maciaszek 2011].

We treat Kanban and Scrum as two separate software development methods implemented on the same level (i.e. the process level). Thus, we can expect that both are some kind of series of steps, rules and artifacts that are used by teams (e.g. Scrum Team, Kanban Team) to produce the product or service.

The third level (i.e. job/performer level) represents all undertakings and practices (e.g. pair programming, code review, demo sessions, continuous integration, TDD) essential to achieving the goals propagated from the process level.

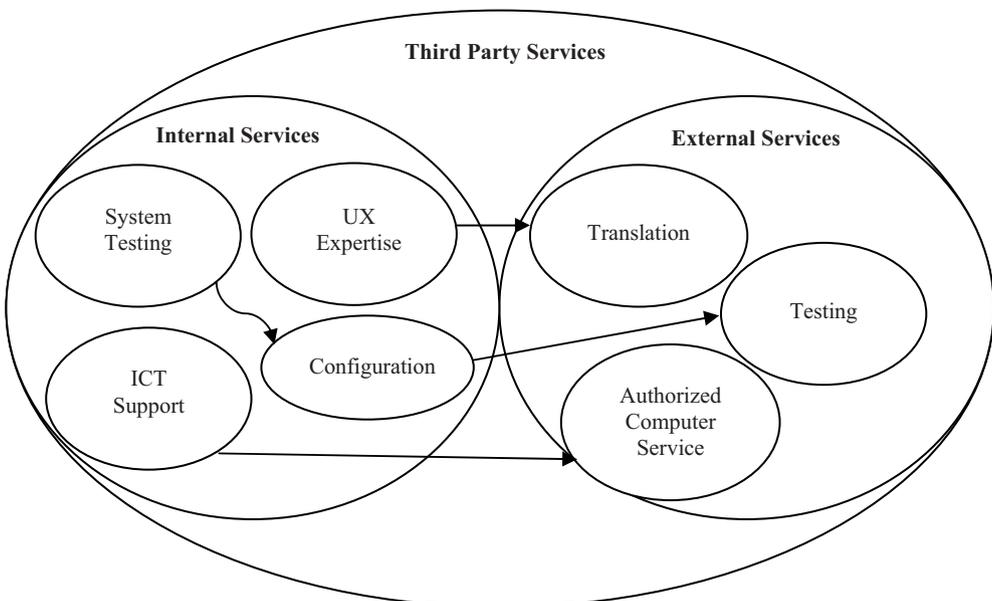
Considering relationships between Agile, engineering practices, Kanban and Scrum, a network organization follows the Agile principles and implements good practices from the Scrum and Kanban as a method for managing teamwork (i.e. job/performer level).

### 2.2. Third party services in the network organization

The network organization lies halfway between vertical integration and market disaggregation that together help in building packets of entities, in accordance with the nature of the provided services or relation between the vendor/supplier and receivers.

In the paper by Sienkiewicz and Maciaszek [2011] we propose to distinguish two types of third party services provided for software development in the network organization:

- internal services – time-consuming activities, additional to the entire project, usually provided by subunits of a large company (e.g. UX expertise, internal testing, ICT support, etc.),



**Figure 2.** Example of relationship between third party services in network organization

Source: own elaboration.

- external services – not covered by Internal Services, handled by other firms (e.g. authorized computer service, external testing service, translation of UI texts, etc.).

Figure 2 presents an example of a relationship between the third party services. It is important to highlight here that both internal and external services may involve other third party services, which in turn may adversely affect the software development process by a deterioration of quality, changes in the scope and increasing delays in the schedule.

The companies that see their units as separate profit or cost centres, may encourage the units to sell their services outside the company, and by operating within the market improve performance, better manage the prices and earn money for the whole organization. This kind of relation usually establishes a long-term relationship between suppliers and helps in building proper cooperation between service providers and other participants of the network organization (ref Section 3.3).

In the next section we will focus on teamwork in terms of core Scrum roles and third party services (i.e. organizations/groups within the network organization), in regard to the proposed mix of the Scrum-based model [Sienkiewicz, Maciaszek 2011].

### **3. Communication, Coordination, Cooperation and Collaboration – the main differences**

We distinguish the terms coordination, communication and cooperation and we use the 3C model (i.e. Communication, Coordination, Cooperation) [Lucena et al. 2008] to classify collaboration, with an emphasis on the group work implementation proposed by the authors of [Camarinha-Matos, Afsarmanesh 2006, 2008; Camarinha-Matos et al. 2009].

For a better understanding of the differences between these terms, we can use the following definitions proposed by [Camarinha-Matos, Afsarmanesh 2006, 2008; Camarinha-Matos et al. 2009] in the context of Scrum:

- Communication (alias networking) – is a way how information is exchanged and used for sharing facts, human experience and practices as well as feelings and rumors, for mutual benefit. It relates to how information is shared and how people understand each other [Denise 1999]. People can benefit from information exchange even without common goals and the common generation of value [Camarinha-Matos et al. 2009].
- Coordination – in addition to exchanging information, it achieves efficiency of aligning and altering activities, which influences the efficiency of the achieved results. Each entity might use different resources and methods to create values or achieve goals at an individual level. There is a correlation between coordination and results like in the traditional Scrum approach where the Scrum master is coordinating Scrum team work by facilitating the whole Scrum process. We deal

with coordination in the case of Scrum master work. The main responsibility of the Scrum master is to facilitate the work of the Scrum team, what in many cases means coordinating the Scrum team and others in following the Scrum rules.

- Cooperation – is a process that involves sharing resources and adjusting activities for achieving compatible results, which provide efficiency for the cooperating entities. The work divides between the involved entities, thus aggregated value is the sum outcomes generated by all participants in a nearly independent manner. A traditional Scrum software development method, based on the client-supplier relationship and the prescriptive roles is an example of the cooperative process, especially when we consider developing from scratch. This usually means that at the very beginning both the client and the supplier do not know what the result will be. Therefore, each participant performs its part of the work in cooperation with others (e.g. the product owner is defining the scope, the Scrum team delivers a demo based on it, etc.). Thus, as an outcome we have a sum of measurable results. Of course, this does not preclude the existence of a common plan, which usually is roughly defined by a single entity (i.e. the customer) and requires co-working, especially when the results of one are delivered to the others. We can say that the goals are compatible as long as the results of the participants *in a value chain lead to the end-product or service* [Camarinha-Matos, Afsarmanesh 2006]. Due to the fact that obtaining collaboration between third party services in a network organization is very hard, that kind of teamwork should be introduced and maintained between all third party service providers involved in software development.
- Collaboration – 4<sup>th</sup> item in 3C model which is described as a collaboration triangle [Camarinha-Matos, Afsarmanesh 2006], is the process where entities are sharing information, resources and responsibilities as well as designing and implementing activities, which are crucial to achieving common goals. This concept derives from the Latin word *collaborare* and means *to work together*, seen as a process of shared creation where a set of entities increase the capabilities of themselves and the others. For the outside observer, collaboration gives the image of a joint identity with the mutual engagement of all participants. We know that building mutual trust takes time but effort and dedication as well as solving problems together, help in building commitment of all the involved entities although value creation is an effort of a team rather than individual contributions. *A collaboration process happens for instance in concurrent engineering, when a team of experts jointly develop a new product. From this example it can be noticed that although some coordination is needed, collaboration, due to its joint creation facet, involves seeking divergent insights and spontaneity, and not simply a structured harmony* [Camarinha-Matos, Afsarmanesh 2006]. This kind of teamwork should be introduced between all stakeholders. Therefore, reaching a high level of collaboration is crucial for the customer representatives and the developing team. In our Scrum-based approach [Sienkiewicz, Maciaszek 2011]

the collaboration is a prerequisite for the product owner, Scrum Master and the Scrum team.

The introduced 3C model is a guide to understanding teamwork in application software development. The authors of the paper [Fuks et al. 2005] wrote: *While communicating, people negotiate and make decisions. While coordinating themselves, they deal with conflicts and organize their activities in a manner that prevents loss of communication and of cooperation efforts. Cooperation is the joint operation of members of the group in a shared space, seeking to execute tasks, generating and manipulating cooperation objects. The need for renegotiating and for making decisions about non-expected situations that appear during cooperation may demand a new round of communication, which will require coordination to reorganize the tasks to be executed during cooperation.*

#### 4. Related work

In literature, a few researchers have already studied the way of adapting the Agile practices in virtual organizations and virtual teams. Usually, Agile software development methods are introduced as a set of principles that need to clarify many different interpretations of the Agile Manifesto [Beck et al. 2001]. Although it is very difficult to determine what exactly is an Agile methodology and how it should be introduced in network organizations.

In our research we focus mostly on the main representatives of Agile software development methodology, and due to the lack of a method dedicated to managing the distributed software development in network organizations, we have selected one (i.e. the Scrum) as the best representative and the Kanban approach as an addition.

Of course, we agree with the Scrum advocates that following the Agile Manifesto principles is possible only because the Scrum defines precisely the essential roles, principles and artifacts, what makes the method very prescriptive for managing software development. In addition to the traditional Scrum, we propose to add a new role (i.e. third party service provider – *S*) and some extra rules for adapting the Scrum and third party services to the network organization.

By referring to the network organization and third party services, we determine how they interrelate with the Scrum and Agile environment. The publication by W. Cellary and W. Picard achieves the agility and pro-activity by introducing the model of Collaborative Network Organization that provided the stimulus for us for reflection about the third party services provider role that is simultaneously a part and a whole in the network organization.

H. Fuks et al. and C. Lucena et al. introduce an approach based on the 3C (i.e. Cooperation, Communication, Coordination) model to the development of collaborative systems [Lucena et al. 2008; Fuks et al. 2005]. The authors studied the 3C model *by means of a detailed analysis of each of its three elements, followed by a case study of learningware application and the methodology of a web-based course,*

*both designed based on this model.* From our point of view, the most interesting is the 3C collaboration model instantiated for teamwork. This interpretation of collaboration and its iterative nature seems to be adequate to our understanding of the dependencies between Scrum roles and other stakeholders (e.g. *The participants obtain feedback from their action and feedthrough from the action of their companions by means of awareness information related to the interaction among participants*” [Fuks et al. 2005].

The idea of collaboration has also been examined by L.M. Camarinha-Matos and H. Afsarmanesh in several papers [Camarinha-Matos, Afsarmanesh 2006, 2008; Camarinha-Matos et al. 2009]. They describe the concepts related to CNOs (i.e. Collaborative Network Organizations), that provide a high-level classification of collaborative networks, and present an application case in the manufacturing industry.

The fundamental assumption of those papers is that a substantial increase in the materializing networked collaborative business ecosystems requires a comprehensive holistic approach. In our further research, we will use the definition proposed by them (mentioned in Section 3) to extend the definition of a relation, between the layers in our Scrum-based model described in [Sienkiewicz, Maciaszek 2011].

## 5. Conclusions

In this paper we have used the 3C model to distinguish teamwork types between Scrum “players”. By extending our previous research from (Sienkiewicz & Maciaszek, 2011) we highlighted the differences between teamwork within each entities which take part in application software development.

Because there is no easy way to adapt the Scrum software development method to work in the Network Organization, we believe that our approach can serve to advance research and help in finding the best solution.

## References

- Beck K., Beedle M., Van Bennekum A., Cockburn A., Fowler M., Grenning J., Highsmith J., Hunt A., Jeffries R., Kern J., Marick B., Martin C. R., Mellor S., Schwaber K., Sutherland J., Thomas D., *Manifesto for Agile Software Development*, <http://agilemanifesto.org>, 2001.
- Camarinha-Matos L.M., Afsarmanesh H., Collaborative networks – value creation in a knowledge society, [in:] K. Wang, M.W.G.L. Kovacs, M. Fang (Eds.), *Knowledge Enterprise: Intelligent Strategies in Product Design, Manufacturing, and Management*, Vol. 207, Springer, Shanghai 2006, pp. 26–40.
- Camarinha-Matos L.M., Afsarmanesh H., *Classes of Collaborative Networks*, [in:] G.D. Putnik, M.M. Cunha (Eds.), *Encyclopedia of Networked and Virtual Organizations*, Vol. 1 A-F, IGI Global, New York 2008.
- Camarinha-Matos L.M., Afsarmanesh H., Galeano N., Molina A., Collaborative networked organizations – concepts and practice in manufacturing enterprises, *Computers & Industrial Engineering* 2009, Vol. 57, No. 1, pp. 46–60.

- Denis L., Collaboration vs. C-three (Cooperation, Coordination, and Communication), *Innovating* 1999, Vol. 7, No. 3.
- Fuks H., Raposo A., Gerosa M., Lucena C., Applying the 3C model to groupware development, *International Journal of Cooperative Information Systems* 2005, Vol. 14, No. 2–3, pp. 99–328.
- Guntamukkala V., Wen J., Tam M.J., An empirical study of selecting software development life cycle models, *Human Systems Management* 2006, Vol. 25, No. 4, pp. 268–278.
- Jarillo C., On strategic networks, *Strategic Management Journal* 1988, Vol. 9, pp. 31–41.
- Lucena C., Fuks H., Raposo A., Gerosa M., Pimentel M., The 3C Collaboration Model, [in:] N. Kock (Ed.), *Encyclopedia of E-Collaboration*, IGI Global, 2008, pp. 637–644.
- Rummler G.A., Brache A.P., *Improving Performance – How to Manage the White Space in the Organization Chart*, Jossey Bass, San Francisco 1995.
- Sailer L.D., Structural equivalence: Meaning and definition, computation and application, *Social Networks* 1978, Vol. 1, No. 1, pp. 73–90.
- Sienkiewicz L.D., Maciaszek L.A., Adapting Scrum for third party services and network organizations, *Proceedings of the Federated Conference on Computer Science and Information Systems, IEEE digital library*, Szczecin 2011, pp. 329–336.
- Van Alstyne M., The state of network organization: A survey in three frameworks, *Organizational Computing* 1997, Vol. 7, No. 3, pp. 88–151.

## KOLABORACJA POMIĘDZY SCRUMEM A USŁUGODAWCAMI ZEWNĘTRZNYMI I WEWNĘTRZNYMI W ORGANIZACJI SIECIOWEJ

**Streszczenie:** Wiele opracowań naukowych przedstawia Scrum jako najlepsze podejście do wytwarzania oprogramowania. Mimo to, metoda Scrum nie jest optymalnym podejściem do zarządzania pracą w organizacji sieciowej, gdzie interesariusze mogą nie wiedzieć nic o metodzie Scrum czy metodologii Agile. W publikacji tej użyliśmy modelu 3C w celu określenia typów współpracy pomiędzy uczestnikami procesu Scrum i innymi usługodawcami (zarówno wewnętrznymi, jak i zewnętrznymi). Wierzymy, że przedstawiona analiza pomoże znaleźć lepiej dopasowane podejście do zarządzania wytwarzaniem oprogramowania w bardzo rozproszonym środowisku (tj. organizacji sieciowej).

**Słowa kluczowe:** model 3C, kolaboracja, kooperacja, komunikacja, koordynacja, Scrum, usługodawcy wewnętrzni-zewnętrzni, organizacja sieciowa, Agile, wytwarzanie oprogramowania.