A Systems Thinking Model for Open Source Software Development in Social Media

Moyen Mohammad Mustaquim
Uppsala University
Department of Informatics and Media
Ekonomikum (plan 3), Kyrkogårdsg 10
Box 513, 751 20 Uppsala, Sweden
Telephone number: +46 18 471 1010
Moyen.Mustaquim@im.uu.se

ABSTRACT

In this paper a social media model, based on systems thinking methodology is proposed to understand the behavior of the open source software development community working in social media.

The proposed model is focused on relational influences of two different systems- social media and the open source community. This model can be useful for taking decisions which are complicated and where solutions are not apparent.

Based on the proposed model, an efficient way of working in open source development community within social media can be found, while social media and open source development- these two systems can be analyzed analogously for further improvement.

Categories and Subject Descriptors

H.3.4 [Social networking]: Model construction and analysis, analyzing a particular community's character stand on a model, model based decision making in social networking.

General Terms

Design

Keywords

Social media modeling, social networking, systems thinking, opens source development.

1. INTRODUCTION

By the existing social media boom on Internet, sharing acquaintance has become further collective and easy accessible, throughout the globe; hence giving a new opportunity to share knowledge from diverse platforms which can be used as a key tool to contribute any effort for open source software development.

While traditional science tends to do observation, identification,

description, experimental investigation, and theoretical explanation of any phenomena to gain new knowledge through experience, systems science can be said to be an interdisciplinary field that studies the nature of complex systems. Systems science aims to develop interdisciplinary foundations which can be applicable in different fields of science for example biology, engineering, social science and so on.

In the field of system science, the systems thinking methodology is practiced before the traditional approach, in order to get a better understandings of a problem or for creating a model of a specific system. The proposed model in this paper uses a systems thinking approach to understand the combined character of two systems. One system is the social network itself and the other system is the open source development community. A model of this type is needed, as it is useful to envisage and find out optimal ways of solving problem by the open source development community under social network. Also this model can be used to identify, and counteract, possible negative effects of the two systems working in parallel.

2. OPEN SOURCE DEVELOPMENT

While the phenomenon of open source software development has recently been given a lot of media attention, the basic concept is actually much older in its origins. It has been estimated that about half of the operating systems for CTSS, an early time-sharing system at MIT, were developed by the users of the system [13] and from the very beginning of computer science, it has been a common practice among researchers, students, and engineers to share source code [6] [11]. One of the motivations for launching the ARPANET project in 1960s was the belief that by connecting different computing sites, communities of computer programmers could more efficiently share their programs and knowledge [7] [9] [10].

As social media sites grow in popularity, so does the desire for smaller niche-based networks that cater to smaller groups of peoples. For instance we can look at the rise of social networks for chess players or even for people who like to sleep in the morning. As the open source software development community is expanding, there is a prominent possibility of using social media tools for them to share their knowledge. The entire community involved in a particular interest (open source software development) can be pooled regardless of differences in time, geographical location or access to knowledge. An attention-

grabbing objective is, if open source software developers become the designers of a social media tool, what will be the consequences and what might be the issues between these two different systems. The model proposed here is intended to be helpful for decision making in such situations if any sensitive issues within systems arise.

3. SYSTEMS THINKING

3.1 Why Systems Thinking

Systems thinking is needed to bridge the borders between different disciplines, as a common language for knowledge exchange. It has importance in predicting the influence of one thing on another without basing this prediction on a specific object. Systems thinking are useful for prediction and can help us in decision making situations. As believed in systems thinking, a small part of any system can be understood by the relationships with other small parts of that system, along with the relationship with the parts of any other system; hence systems thinking focus on the influence rather than the cause. How one system will work and what the impact will be on other systems; can be predicted and reasoned about by means of systems thinking. Of course it is important to understand what a system means in respective context. The term “system” can mean different things in different contexts, situations and scenarios. When reasoning about complex problems that involve many actors, systems thinking can help actors to see the 'big picture' and not just their own part of it. There might be problems whose solutions are not obvious. And there might be recurring problems or problems that have been made worse by past attempts to fix them. Systems thinking have proven its value in such situations [12].

Traditional analysis focuses on separating out the individual pieces of what is being studied, while systems science in contrast focuses on how the things being studied interacts with the other constituents of the system. Thus instead of isolating smaller parts of the system being studied, systems science works by expanding its view to take into account a larger number of interactions. The result is dramatically different than what is obtained from traditional forms of analysis, particularly so if the system studied is considered as dynamically complex [6].

In system science there can be issues that affects or are affected by the surrounding environment. Such issues, where the environment can be stable or dynamic [1], systems thinking are very handy while dealing with such issues. However the target of the systems thinking remains somehow the same regardless of what kind of system it is while trying to predict an issue influenced by another system [6].

3.2 Emerging Properties and Synergy

Emerging properties are considered to be the unexpected behaviors that stem from interaction between the components of a system and between the components and its environment [12]. It is one of the significant challenges of complex system engineering. As a system can be a combination of sub systems that are highly integrated to achieve the overall goal of the system, the emerging properties hence can create problems while interacting among the components of the sub system; onward down or up in the hierarchy. Although emerging properties sometimes give positive effects, for instance user produced knowledge that can help in the system designing process, but it also can be harmful.

The nature of emergent properties can lead the system to be more than the sum of its component parts. Therefore emerging properties have considerable significance for the design and engineering in many system and applications.

Although the role is often unappreciated, synergy can be considered as one of the core concepts of systems sciences. It can be defined as a combined action of different entities of the system where the output or final outcome is favored over the difference of individual action of any entity. The term emergent behavior rises from the definition of synergy where the whole system's behavior is unpredictable comparing with the behavior of the parts of the system taken independently. Hence in systems thinking, it is a critical issue as the working together of two or more things (subsystems) to produce an effect greater than the sum of their individual effects is synergy, which is important for decision making and for predicting the system's performance over another system. There are different implications of the synergism hypothesis in system science and the goal is to come up with an advantageous outcome where the sum is greater than the individual parts [5].

3.3 Hard and Soft Systems Thinking

Hard systems thinking (HST) is an approach to real-world problems in which an objective or end-to-be-achieved can be taken as given [2]. Then, to meet or achieve the objective, a system is engineered. The distinguishing characteristic of HST is the belief that all real-world problems can be formulated in the following way:

That, there is a desired state, S_1 , and a present state, S_0 , and there are alternative ways of getting from S_0 to S_1 . Problem solving according to this view consists of defining S_1 and S_0 and selecting the best means or ways of reducing the difference between them. In systems engineering ($S_1 - S_0$) defines the need, or the objective to be attained, and systems analysis provides an ordered way of selecting the best among the alternative systems which could fulfill that need or objective [2].

Situations involving actions of humans could be claimed to be complex by definition. Complex situations have ill-defined or unknown ends or objectives, and are thus likely to call for another approach than hard systems thinking. Soft systems thinking (SST), or interpretive systems approach [8], first looks at “what”, and secondly considers “how” [1]. We consider the “what” to be essential for dealing with complex situations as there are most often no given goals. The “what” looks at the situation and identifies relevant system as well as possible ends.

Checkland [2] suggests that while HST belongs to the paradigm of optimization, SST belongs to the paradigm of learning. This is interesting in that the goal-seeking purpose has been moved down on the priority list, in order to make space for learning about the situation at hand and possible goals. The learning involves identification of the system including structure and boundaries in order to get a holistic view of the situation. Once the system is in place, it should be possible to identify possible goals.

SST is not a replacement of HST, but rather a complement [12]. Once the objective is known, HST could be the suitable approach.

4. PROPOSED MODEL

The proposed model has two different configurations. The first structure provides a viable model view of accomplishing a development task within the open source community. The second structure shows how to find imperfections while two systems are working together with each other, while the open source developer system is considered to be a sub system of the social media system.

Any specific platform of social media (existing or to be designed) can be used to employ this model and see the differences between performances, with or without model based structure. For example an application can be designed based on this model which can be used to bring open source software developers under one umbrella and being users of such application in a social media platform, they share their knowledge for a specific development task, which will be optimal, time and cost effective while the structure of this model is followed during application design.

4.1 Open Source Development Structure on Systems Thinking Model

This structure of the model consists of three main components- the principal user's task, involvements of other users and finishing the task. Hard systems thinking and soft systems thinking methodology are used to construct this model. The model is shown in Figure 1 and each part is described in the following sub sections.

4.1.1 User's Task

The principal user in the community is the one who starts a particular blueprint or development of a piece of software. To do so, the user involves in learning new things as long as no difficulties are faced. Once the user finds complexity, he then tries to brush up his own skills to solve the problem himself. In these phases the user's attempt of solving problem is goal oriented, hence typical hard systems thinking methodology is applied. Once user feels complicatedness, it is then time according to the model (Figure 1) to hand over the problem to other users engaged through social media platform.

4.1.2 Other User's Engagement

The open source software development community in social media may consist of users with different knowledge level. All users cannot solve a particular problem. And also it is not the idea for everyone to get involved to solve one problem. The idea of sharing knowledge suffers in redundancy then. The model describes three principal tasks while other users get involved into the problem sent by the primary user. It is important that they understand the problem first. And once they want to integrate with other users for sharing their knowledge, they need to know the optimal way of communication with others. Again, optimal way varies from situation to situation hence they would like to know what the typical optimal way is in their specific case. As seen on figure 1, soft systems thinking methodology is used to find the answers of these questions. Once soft systems thinking is successfully applied and user finds their answers as required by them, they can go for achieving the goal, that is the next step,

using hard systems thinking methodology (Figure 1). It is important to understand that soft systems thinking methodology was first used to narrow the goals of a specific user so that the goal can be achieved in a fewer steps, using hard systems thinking methodology in the model.

4.1.3 Finishing a Task

While multiple users are reaching their own specific goals their ideas are combined and the knowledge is sent back to the primary user who then can use this knowledge to finish the task (Figure 1).

It is important to notice that the social media being an independent system acts differently without the use of such model. While systems thinking model is used in a particular sub system (open source developer group, in this case) of the social media system, the tasks are broken up in a top down manner for the ease of completion.

Figure 1. Systems thinking model for open source development in social network.

4.2 Manipulation of Open Source Community with Social Media by the Model

As mentioned earlier, systems thinking approach is suitable to figure out the effects of one system with its surrounding system, it is interesting to see the behavior of social media and open source community- two being independent system working together and one within other system. The systems thinking model for finding out the issues, which can be tuned to overcome problems which arises in development phase is shown in Figure 2.

Using both hard and soft systems thinking methodology, this model is designed to find any unexpected behavior or synergy of the systems and is described in the following two sub sections.

4.2.1 Unexpected Behaviors of System

Soft systems thinking approach helps in finding emerging properties while hard systems thinking attempts to solve it (Figure 2). In this particular case, user involvement in developing software might turn out to be something else which was not meant

to be. Each individual user is considered to be a sub-system of the open source development community and due to emergent properties inside the open source community system; the expected goal might not be achieved. These problems might affect the existing social network system even though there were no emerging properties in that system.

Also synergy is another issue to be considered. Subsystems' combinatorial outcome and final outcome that is the goal, has huge influence on the type of two systems this model is based on. For example user's individual work can be favored over the overall goal of the development, leading towards bad synergy which is important to detect and overcome. If users in the subsystem (open source developer) create more negative synergy it can have adverse effect on social media system too. Soft systems thinking methodology is therefore helpful to find emergent properties of these systems according to the model (Figure 2).

Of course there are good synergies also, for instance performance of few users in solving specific tasks, can be useful if adopted in the system and these also can be found using soft systems methodology. For instance, this type of good synergy leads to faster development in this case. If the open source developers become the gradual developer of a social media themselves, it is a challenge of software engineering to handle both good and bad synergies technically since personal skill gets prioritized over attaining the principal goal hence leading good and bad synergy in two systems. Top down investigation of this model in such circumstances helps in decision making, at the same time as the reverse engineering practice of the model can as well be very functional in state of affairs like this.

4.2.2 Overcoming the Error

Once there are errors detected such as the possibility of emergent properties or bad synergies, hard systems thinking can be used to overcome such situation. Hard systems thinking methodology where the goal is known is best to apply in such situation not only to overcome the error but also to use the good synergies in a system, found using soft methodology (Figure2).

Figure 2. Error handling in social networking.

5. CONCLUSION

While social media opens a door of enormous opportunity in terms of sharing interest of specified knowledge, it is important that the sub systems working under social media follow a specific model to make the most out of social media's power. The proposed systems thinking model for open source software development community in social media can be used to create a tool which can let the people within such community work together under social media; thus creating the best possible output against the effort and time implied. Based on such model, social networking structure can be altered for the enhancement of both sub systems under social media and the social media itself.

6. REFERENCES

- [1] Aronson D. Overview of Systems Thinking. Communications. 1996.
- [2] Checkland, P. (1981). Systems thinking, systems practice. Chichester, Wiley.
- [3] Checkland, P. (1985). "From Optimizing to Learning: A Development of Systems Thinking for the 1990s." The Journal of the Operational Research Society **36**(9): 757-767.
- [4] Checkland, P. and J. Scholes (1990). Soft systems methodology in action. Chichester, Wiley.
- [5] Corning, Peter A. (1996). "Synergy and the Systems Sciences"
- [6] D. Caminer, J.Aris, P. Hermon, F. Land, 1996. *User driven innovation: The World's First Business Computer*, New York, McGraw-Hill.
- [7] E:E David, Jr, R.M Fano, 1965. "Some thoughts about the social implications of accessible computing" excerpts reprinted in IEEE Annals of the History of Computing, volume 14, number 2, pp. 26-39.
- [8] Emmeche, C., S. KØppe and F. Stjernfelt (1997). "EXPLAINING EMERGENCE: TOWARDS AN ONTOLOGY OF LEVELS." Journal for General Philosophy of Science **28**(1): 83-117
- [9] J. Abbate, 1999. *Inventing the Internet*. Cambridge, Massachusetts MIT Press.
- [10] J. Naughton, 2000 *A Brief History of the Future: From Radio Days to Internet Years in a Lifetime*. Woodstock, N.Y Overlook Press.
- [11] K.Kuwabara, 2000, "Linux: a bazaar the edge of chaos" *First Monday*, volume 5, number 3 (March), http://www.firstmonday.org/issues/issue5_3/kuwabara/
- [12] Marchal, J. H. (1975). "On the Concept of a System." Philosophy of Science **42**(4): 448-468.
- [13] R.M Fano, 1967. "The computer utility and the community" *IEEE International Convention Record*, pp. 30-34, excerpts printed in IEEE Annals of the History of Computing, volume 14, number 2, pp.39-41.