A Model of Self-Organizing Collaboration

Rumen Andreev

Institute of Computer and Communication Systems, Bulgarian Academy of Sciences, Acad. G. Bonchev str. Bl. 2, 1113 Sofia, Bulgaria, e-mail: rumen@iccs.isdip.bas.bg

Abstract Collaboration joins together persons (active objects) in some activity. The paper concentrates on the activity, since it is the collaboration basis. The basic theory used in computer science for activity analysis is the activity theory that considers activity as a substantial part of the human interaction with the objective reality (environment). In conformance with this theory, the action presents activity substance and operation – activity (action) realization. It is analyzed cooperation as collaboration realized in operation context.
The activity theory considers a kind of activity that does not reveal it as an independent, autonomous thing (self-organizing activity). This paper indicates the main characteristics of self-organizing activity that are basis for content-, context-independent modeling of autonomous activity. The presented model uses formal constructions of the mathematical logic as autonomous frameworks. The self-organizing activity is foundation for modeling of self-organizing collaboration that results in a model describing a collaborative self-organizing system. The latter is framework for a real process and bases on a group of active objects associated by a shared need.

Keywords: Self-organizing activity, cooperative self-organizing system, modeling, reasoning, real process

1. Introduction

Collaboration indicates a form of participation of persons or active objects in some activity. In accordance with this form, the persons (active objects) are joined together in an activity, i.e. the activity is in state of being joint activity that ensures partnership. The collaboration has two ingredients, since it is a mixture of a group of persons (active objects) and an activity. Our consideration concentrates on the most important ingredient, which is the basis of collaboration - activity. In the computer science, the activity analysis is realized in the light of activity theory.

In view of the activity theory, the activity is substantial part of the human interaction with the objective reality (environment) i.e. the activity is determined through a system that balances [1]. The most fundamental principle of this theory states that the human mind that is one of the main features of human beings is a special component of human interaction with the environment. According to the second principle of this theory, the activity is object-oriented. The object could be not only a material thing of the human environment, but it is considered as a desired object (goal), motive or objective, as well. Hence, the activity is a substantial part of a system, which components human beings and the objects of environment interact. The system-relevant analysis of activity is due to its objective character and reveals that the activity takes part in keeping an (eco)system in state of balance. The system is a framework, in which the activity exists.

Another basic principle of the activity theory is the "hierarchical structure of activity" that concerns the activity ontology. The activity exists on three levels with regard to its nature - its existence results of a cause (object). At the highest level the activity is considered as result of motive - object that causes human beings to act. On the second level the activity exists as action - goal-directed activity. On the lowest level the actions are realized through operations that are determined by the actual conditions of activity. The operations convert actions into facts. This is a psychologically relevant consideration of activity.

According to a psychologically irrelevant presentation of activity existence, the activity description on the highest level is result of system-relevant analysis that presents its objective character: The activity is a phenomenon of the physical world which existence is explained by the principle of causality. The human that is one of the system components does not cause the activity. It serves for helping bring about the activity, i.e. it is activity factor – agent. As an agent the human being can be both *reactive* and *proactive*. The activity theory adopts the idea that human beings are proactive agents, since the reaction (activity of a reactive agent) is automatic, i.e. unconscious. A proactive agent is able to realize and adopt goals and to take the initiative. It is obviously that on this level the activity analysis and representation can be psychologically irrelevant (objective) while on the other levels they are psychologically relevant [2]. The psychologically relevant analysis reveals the human relevant substance of activity.

The psychologically relevant action is virtual activity, which substance is presented by the line of action. The latter depends on the goal and relates to the way of action happening (course of action). The course fixes the form of action existence (operation form). In view of the physical existence of an action, the activity is considered as operation. When a man operates, we say that he is in action, i.e. he implements its course that serves as purpose of human operation. The purpose causes successful operation (functioning) of human being or executive system. It is considered as objective cause of operation and is well known as *objective*. The operation needs of a plan (organization) of action implementation. Its development bases on the course of action and resources helping bring about an action. The course of action and resources are two operation factors that determine its

context. The former serves as a plan of operation organization. There are two types of resources that guarantee operation: resources that carry out operation (*operation performer*, agent) and resources that support operation performer in doing the action. The purpose (objective) and resources are two interacting factors that are in the base of operation and present an integrated context.

Usually the collaboration is known as cooperation, since it is considered with respect to operation. There are three interacting elements that support the cooperation reality: set of agents, shared resources and shared objective. They determine the framework, which supports an engineer in construction of cooperative forms (Fig. 1). The agents take parts in various cooperative actions, in which they have different roles [3]. They are the substance of cooperation. An agent can either be a human or a computer-based component, which is an active process supported by a computer system [4].

This framework supports the development of various methodologies for cooperation design and realization. The cooperation transforms the set of agents into a group of agents. There are two factors that help the composition of a group: shared resources and a shared view of communicating agents on a subject or shared objective of working agents [5]. The shared resources are contained in environment that is influenced by organizational approaches and coordination techniques. This organized environment is the *background* of cooperation. The shared view is the cooperation *foreground*, i.e. the domain, in which the agents cooperate [6].

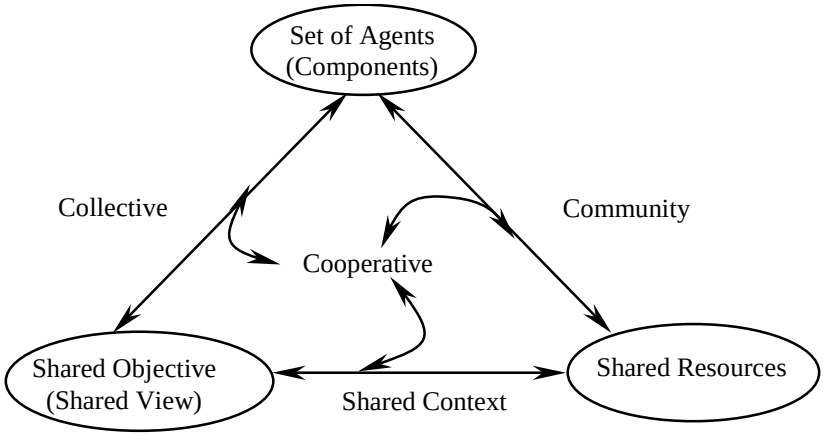

Figure 1. Engineering framework for cooperation

The three cooperation factors act on each other. The shared resources helping transformation of a set of agents into *community* are *community factor*. The community is a cooperative form. The cooperative use of shared resources is a typical community-wide behavior of the members of community, which is considered as a whole. The shared objective is *collective factor*. It helps the union of agents in cooperative form known as *collective*. This is a group of agents that looks like a

whole. The shared objective guarantees collective behavior of agents in an organized system. The shared objective and shared resources compose shared context that determines the cooperative operation of agents. They ensure two types of groupware. The group construction is analyzed not only with respect to cooperation, but with regards to communication, as well. The groups regarded in communication assist the cooperation, since the information is a necessary factor for operation management [7]. The shared objective is a factor for organization of resource environment and the shared resources are factor (means) for achievement of a shared objective. The interaction between these cooperation factors ensures an integrated shared context.

The cooperation background can be computer-supported environment (computerized environment), in which the resources are usually distributed. Environment consisted of resources of the same kind is *homogeneous environment* that guarantees uniform access to the resources. If the latter are of different kind, the environment is *heterogeneous*. The unification of the access to the resources of this environment requires its homogenization that can be achieved by using of grid technology [8]. The cooperative use of resources is provided by services for coordination, shared sessions and support of synchronous and asynchronous access [9, 10]. The shared objective that is the logical part of shared context organizes the agents that are workflow participants in a collective system, i.e. coordinates their work [11]. A common project can be regarded as shared objective [12].

The shared view can be used for arrangement of computerized environment, which becomes setting for cooperative work of agents of a group that is regarded as multi-agent system [13]. The shared view is a conceptual model that is factor of interoperability supporting cooperation [14, 15]. In Internet-based environment the common view ensures the building of Semantic Web that ensures cooperation through interoperability [16]. The framework for construction of cooperative forms integrates the main cooperation factors: agents, computerized environment (grid) and shared objective. In this way, it gives an approach to solution of the problem of agents/grid integration [17].

Self-organizing collaboration requires a self-organizing basis, i.e. a self-organizing activity. The self-organizing activity is the substance of autonomous activity, which main characteristic is independence. A thing that is complete in itself (a whole) is an autonomous thing. The self-organizing activity has the form of unity, since it results of arrangement of its parts to form a complete whole. The activity theory explains a kind of activity that is not self-organizing, since it has the following characteristics:

- The activity is substantial part of an integrated system: The integration is due to interaction that ensures system balance. The activity takes part in this interaction, but does not organize it, i.e. it has not unity form;
- As a necessary consequence of an external cause presented by a motive, the activity evidently subjects to the law of causality and its existence *depends on* an external object – the activity is not independent;

- The proactive agent and object that causes the activity are determining elements (*determinants*) of activity and present its context: The *context-dependent* determination of activity is a systematic view on it that does not describes the activity as autonomous thing - independent object with quality of wholeness. It is necessary to distinguish the completeness of an activity from its wholeness that is characteristic of an integrated system.

The next section defines self-organizing activity. On this basis it presents a model of collaborative self-organizing system. The third section describes an engineer view on the content-, context-independent model of self-organizing collaboration.

2. Model of collaborative self-organizing system

The basis of self-organizing collaboration is self-organizing (autonomous) activity. This kind of activity has the following basic characteristics:

- The law of causality is a principle of self-organizing activity: It is in the nature of self-organizing activity
- An autonomous activity results of an intrinsic physical need of an object (human being) and belongs to it, as its capability: As this object holds the activity, it is *active object* that causes an activity and helps for its happening;
- In view of the activity independence, the activity of an active object must ensures the satisfaction of its intrinsic need with the help of another object that must meet this need: This *necessary object* guarantees the activity completion, i.e. it is a part of activity;
- This kind of activity involves in itself all objects that are necessary for its existence, integrates them as parts of a complete whole and arranges them in harmonious relation: The result is the composition of a unity.

These characteristics are necessary for nature-based modeling of self-organizing activity that ensures content-, context- independent representation. The result of this way of modeling is a formal model.

The presented kind of activity materializes the law of causality. Its substance is an integrated system consisting of two interacting components - active object and necessary object. This system guarantees activity wholeness. In self-organizing activity the necessary object does not only correspond to the active object, but corresponds with it, as well. The two objects are in harmonious relation, since the necessary object meets an intrinsic need of the active object. Harmony and equality are simultaneously the most important characteristics of autonomous activity.

According to the nature of self-organizing activity, this kind of activity can be presented by implication, since it present a formal material form that involves two objects that are simultaneously in harmonious relation and in balance. This consideration of implication is in conformance with its definition given by the mathematical logic through a truth table [18]. Using the definition of equivalency,

the implication (autonomous activity) has the following presentation, in which a_object is active object and n_object is necessary object

$$a\_object \Rightarrow n\_object \equiv (a\_object \Leftrightarrow n\_object) \vee (\neg a\_object \wedge n\_object) .$$

This statement reveals that the implication is union of equivalency and harmonious relation ($\neg$a_object $\wedge$ n_object), where *n_object* satisfies a need of active object ($\neg$a_object). The equivalency *a_object* $\Leftrightarrow$ *n_object* presents an integrated system that bases on the balance due to the interaction between an active object and necessary object. As the integrated system (equivalency) and harmonious relation are the basis of a self-organizing activity, which substance coincides with the definition of unity [19], it is obviously that the implication indicates an autonomous activity, i.e. it is in conformance with unity. Therefore, the formal model of self-organizing activity is *active object* $\Rightarrow$ *necessary object*.

Since the self-organizing activity supports self-organizing collaboration, the model of autonomous activity is a model of the framework for self-organizing collaboration. It represents the outer form of self-organizing collaboration, in which active objects act together. It is accepted that two active objects are partners in a self-organizing activity. This self-organizing collaboration is presented by the following expression

$$\text{active object}_1 \wedge \text{active object}_2 \Rightarrow \text{shared necessary object.}$$

This model of self-organizing collaboration states that the necessary object satisfies a need, which the active object_1 shares with the active object_2, i.e. it is a shared necessary object (sh_n_object). The self-organizing collaboration is due to *a shared need*.

As result of nature-based (formal) modeling, the model of self-organizing collaboration presents its essential characteristics:

1. Self-organizing collaboration exists in a collaborative self-organizing system. The active objects and necessary object are components of a collective self-organizing system, since they are in symmetrical relation guaranteed by the autonomous activity, in which they take part. The symmetry presented by its qualities of harmony and balance is a characteristic of self-organizing activity.

2. Collaborative self-organizing system is framework for a real process. The self-organizing collaboration has several representations. The upper model is its basic representation that could be presented in the following way

$$\text{active object1} \Rightarrow (\text{active object2} \Rightarrow \text{shared necessary object}).$$

This expression states that a complex autonomous activity can exists in the collaborative self-organizing system. The complex autonomous activity is a real process, since it presents a continuous succession of activities. The continuity is a necessary consequence of the wholeness of a object. As the autonomous activity

ensures the framework for self-organizing collaboration, the collaborative self-organizing system is a complete whole, as well.

3. Self-organizing collaboration is induced by naturally associated active objects that are in state of being united with an object, which helps them to satisfy their shared need and to integrate them. The following statement presents another alternative expression of self-organizing collaboration

$$a\_object_1 \wedge a\_object_2 \Rightarrow sh\_n\_object \equiv (\neg\, a\_object_1 \vee \neg\, a\_object_1) \vee sh\_n\_object$$

This expression is inspired by the equivalence between $(p \Rightarrow q)$ and $(\neg p \vee q)$ settled by the mathematical logic. It states that the active objects taking part in collaboration are naturally associated by a shared need, i.e. they are in a group. They are of different types. The shared necessary object integrates them in a whole. They wholeness of the group of active objects is preserved in the framework of self-organizing collaboration.

3 An engineer view on the model of self-organizing collaboration

There is a dualistic view on the representation of self-organizing collaboration that is a special kind of collaboration and is supported and shaped by collaborative self-organizing system. The latter is a material form, in which the collaboration exists. It determines unconditionally an outer form that ensures the happening of self-organizing collaboration. Hence, the collaborative self-organizing system is *a priori form* of self-organizing collaboration and it is a foundation for achievement of a real self-organizing collaboration. In view of an engineer, the presented model is very general description of autonomous collaboration. It is due to the fact that this model does not concentrate on the substance of self-organizing collaboration. The system-relevant analysis of self-organizing activity gives only the outer form (shape), in which it exists and ensures its formal (general) description.

In view of an engineer, the formal description of self-organizing collaboration is useless. An engineer shows interest in the substance of an object that he designs and realizes. For this purpose, it requires a detailed object description that presents the inner form of the object. An inner form that gives the nature of object existence is necessary for object specification. The detailed description does not only ensure a detailed representation of engineer view on object existence, but guarantees object realization, as well.

The satisfaction of engineer requirements can be achieved by a deductive approach to consideration of the model of self-organizing collaboration. It requires more definite presentation of collaboration substance on the levels of action and operation. It has to be in conformance with the limitations of the collaborative self-organizing system. On the level of action it is presented the particular substance of self-organizing collaboration. On the lower level of operation the col-

laboration presentation is more concrete, since the operation description gives specific knowledge that is necessary for the performance of action.

On the action level, the collaborative self-organizing system has to undergo the following changes: the shared necessary object is treated as shared goal; the active objects are regarded as human beings; the activities of active objects are considered as separate processes, i.e. the whole process is divided in two. To keep the harmony of self-organizing collaboration, the first of the two processes has to produce a satisfactory result, i.e. a result suitable for use of the second process. The second process has to produce a desired result. It is evident that the processes have to be efficient. This is the requirement to the psychologically relevant presentation of self-organizing collaboration on this level. With respect to the operation level, the two main processes are presented as series of actions that are realized through operations. To satisfy the requirement for realization of efficient processes, it is necessary to control the operations. In view of the control, there are the following paradigms of operation performance: *arrangement-conducted operation, operation in regulation* and *adaptation-based operation.*

To be in line with computer science, the self-organizing collaboration could be represented on the action level in the following way

$$\text{Provider} \wedge \text{User} \Rightarrow \text{shared goal}.$$

In this model the user and provider are the active objects that are associated by a shared need presented by a shared goal. This model is suitable for representation of an education system, in which the teacher is considered as provider and the student as user. [20]. This statement is useful for representation of a software development system, as well. Here, the software engineer is a provider. The provider can be a producer or servant.

In the following presentation of the upper model of self-organizing collaboration, the provider carries out a producer role

$$\text{Provider} \Rightarrow (\text{User} \Rightarrow \text{shared goal}).$$

In this form the model represents the following real meaning (substance) of the analyzed self-organizing collaboration:

- User regulates the producer process and has to achieve a desired result: The producer has to supply the user process with a satisfactory product;
- The provider activity (production process) supports the collaboration and reveals the capabilities of the collaborative self-organizing system: This model of the system is known as process-oriented, since it supports two processes;
- The symmetry of the collaborative self-organizing system is ensured through a harmonious relation between the provider and user: The former has to satisfy the user requirements.

Another form of action-based model of collaborative self-organizing system adopts the idea that the provider is a servant

User $\Rightarrow$ (Provider $\Rightarrow$ shared goal).

This form of the psychologically relevant model of self-organizing collaboration brings out in the foreground, the following essential characteristics of the system:

- The provider is a servant: it is treated as a means in the collaborative process;
- The model presents a service-oriented approach to the realization of collaborative self-organizing system;
- The symmetry is ensured by interaction between the user and provider: In this case the symmetry is presented by its quality of balance;

If the provider is simultaneously a servant and producer, its process has to produce a desired result (not satisfactory result).

4 Conclusions

The analysis of self-organizing collaboration concentrates not on its substance or realization, but on its outer form that coincides with the framework of autonomous collaboration. This approach results in a content-, context-independent representation of collaboration. Since the formal constructions of the mathematical logic are suitable for formal modeling of self- organizing activity and collaboration, the latter are expressed by the five well-known operations (constructions) of the mathematical logic: equivalency, implication, disjunction, conjunction and negation. They are taken up not as structures that serve for presentation of subject substance, but as frameworks (outer form of a whole thing) presenting the material form of self-organizing collaboration.

The different ways of presentation of implication that represent self-organizing activity support our reasoning on self-organizing collaboration. It results in the following findings: self-organizing collaboration exists in the border of an autonomous system; a collaborative self-organizing system is a framework for a real process; naturally associated active objects induce self-organizing collaboration. The usage of the operations of the mathematical logic for presentation of various autonomous frameworks needs of a new interpretation of the basic logic symbols A and $\neg A$ and the basic constructions, in which they take part.

References

1. Kaptelinin, V.: Activity theory: Implications for human computer interactions. In: Brower-Janse, M., Harrington, Th. (eds) Human-Machine Communications for Educational Systems Design, NATO ASI Series F Vol. 129, pp. 5-15. Springer-Verlag (1994)
2. Vicente, K.: Wanted: psychologically relevant, device- and event-independent work analysis techniques. Interact. With Comput. Vol. 11, pp. 237-254 (1999)
3. Tamai, T.: Objects and roles: modeling based on the dualistic view. Inf. and Softw. Technol. Vol. 41, pp. 1005-1010 (1999)
4. Wegner, H., Hupe, P., Matthes, Fl.: A process-oriented and content-based perspective on software components. Inf. Syst. Vol. 25 (2), pp 135-156 (2000)
5. Larsson, T.I., Vainio-Larsson, A.A.: Software producers as software users. In: Gilmore, D.J., Winder, R.L., Detienne, Fr. (eds) User-Centered Requirements for Software Engineering Environments, NATO ASI Series F Vol. 123, pp. 285-306. Springer-Verlag (1994)
6. Lu, St. C-Y, Cai, J.: STARS: A socio-technical framework for integrating design knowledge over the Internet. IEEE Internet Comput. Vol. 4 (5), pp. 54-62 (2000)
7. Guerrero, L.A., Fuller, D.A.: A pattern system for the development of collaborative applications. Inf. and Softw. Technol. Vol. 43, pp. 457-467 (2001)
8. Roure, D., Baker, M.A., Jennings, N.R., Shadbolt, N.R.: The evolution of the grid. In: Fox, G., Hey, A.J.G. (eds) Grid Computing-Making the Global Infrastructure a Reality pp. 65-100, John Wiley and Sons (2003)
9. Lopez, P.G., Skarmeta, A.F.G.: ANTS framework for cooperative work environments. IEEE Comput. Vol. 36 (3), pp. 56- 62 (2003)
10. Singh, M.P.: Conceptual modeling for multiagent systems: Applying interaction-oriented programming. In: Chen, P.P., Akoka, J. (eds) Conceptual Modeling, LNCS Vol. 1565, pp.195-210, Springer-Verlag (1999)
11. Gellersen, H.-W.: Support of user interface design aspects in a framework for distributed cooperative applications. In: Taylor, R.N., Coutaz, J. (eds) Software Engineering and Human-Computer Interaction, LNCS Vol. 896, pp. 196-210, Springer-Verlag (1995)
12. Perez, M., Rojas, T.: Evaluation of workflow-type software products: a case study. Inf. and Softw. Technol. Vol. 42, pp. 489-503 (2000)
13. Liedekerke, M.H., Avouris, N.M.: Debugging multi-agent systems. Inf. and Softw. Technol. Vol. 37 (2), pp. 103-112 (1995)
14. Szykman, S., Fenves, St.J., Keirouz, W., Shooter, St.B.: A foundation for interoperability in next-generation product development systems. Comput.-Aided Des. Vol. 33, pp. 545-559 (2001)
15. Chen, P.P., Thalheim, B., Wong, L.Y.: Future directions of conceptual modeling. In: Chen, P.P., Akoka, J. (eds) Coceptual Modeling, LNCS Vol. 1565, pp. 287-301, Springer-Verlag (1999)
16. Aroyo, L., Dolog, P., Houben, G.-J., Kravcik, M., Naeve, A., Nilsson, M., Wild, Fr.: Interoperability in personalized adaptive learning. Educ. Technol. and Soc. Vol. 9 (2), pp. 4-18 (2006)
17. Foster, I., Jennings, N.R., Kesselman, C.: Brain meets brawn: Why grid and agents need each other. In: Proceedings of AAMAS'04, ACM (2004)
18. Stall, R.R.: Sets, Logic and Axiomatic Theories. Freemand & Company, London (1965)
19 Hornby, A.S., Cowie, A.P., Gimson, A.C.: Oxford Advanced Learner's Dictionary of Current English (Third Edition). Oxford University Press, Oxford (1987)
20. Andreev, R.D., Troyanova, N.V.: E-learning design: An integrated agent-grid service architecture. In: Proceedings of IEEE John Vincent Atanasoff 2006 International Symposium on Modern Computing, pp. 208-213, IEEE Computer Society (2006)