

MODELING STRUCTURED NON-MONOLITHIC COLLABORATION PROCESSES

Willy Picard

Department of Information Technology
The Poznań University of Economics
ul. Mansfelda 4
60-854 Poznań, Poland
<picard@kti.ae.poznan.pl>

Existing systems supporting collaboration processes typically implement a single, fixed collaboration protocol, and collaboration process takes place inside a single group. In this paper, we present a model which provides support for multiple collaboration protocols for non-monolithic collaboration processes, i.e. collaboration processes in which collaboration is spread among many groups. In the presented model, collaboration protocols include communicative, "acting", and social aspects of collaboration processes, and the introduction of group actions provides support for group dynamics. Conditions that collaboration protocols have to fulfill to be valid are also presented.

1. INTRODUCTION

From prehistoric tribes to trade unions, group structure has always been at the heart of human activities. Grouping their competences, humans are able to achieve great projects, from pyramids to railroad infrastructure construction. The keyword for group activities is *collaboration*. Collaboration is the process of sharing competences to achieve a common goal.

To a recent past, the collaboration process was limited by the requirement of a single location. People involved in a collaboration process needed to meet to exchange information. In reality, people are generally spread on large geographical area. Meetings are difficult to organize, because of schedule incompatibilities, and costly in terms of time and money.

Telecommunication networks provide a partial solution to the former problem. Telecommunication networks let collaborators be spread over various locations. The use of telephone allows collaborators to exchange information via voice communication. Documents can be exchanged via fax in a graphical format. Local area networks (LAN) are the basis of electronic information exchange inside enterprises, while wide area networks (WAN) - in between enterprises.

With the rise of telecommunication networks, collaboration models that rationalize the collaboration process have been developed. Most of them are document oriented, i.e. the fundamental object of the collaboration process is one or more documents. In enterprises' intranets, collaboration tools are currently widely used for sharing files, for group scheduling or for document collaborative writing.

Traditionally, research in electronic support for collaboration has concentrated on

collaboration processes confined inside a single group. Few attention has been accorded to the case of *non-monolithic* collaboration processes, i.e. processes in which the collaborative activities are spread dynamically among potentially many groups. The term “non-monolithic” is taken from the negotiation vocabulary (see [10], pp. 4-5, 389-406), where a non-monolithic negotiation process is a negotiation process in which some parties do not behave as a unitary decision entity, i.e. a party consisting of many persons with various perceptions and goals.

In the field of computer support for collaborative work (CSCW), some works have addressed the issue of the group data organization in a dynamic way [3], the issue of non-monolithic collaborative document edition [8]. These works are usually poorly formalized and focused on very limited applications. In the field of electronic negotiations, some works addressed the issue of negotiation protocols [1][2][4][5][7][13]. According to [6], a negotiation protocol is “a formal model, often represented by a set of rules, which govern software processing, decision-making and communication tasks, and imposes restrictions on activities through the specification of permissible inputs and actions”. One may consider a negotiation protocol as a collaboration protocol. Works in the field electronic negotiations are usually limited to monolithic negotiations, or address a single user's point of view and do not provide support for group collaboration. To our best knowledge, the issue of support for both structured and non-monolithic collaboration processes has never been addressed.

As stated in [14], “there can be no separate document and communication management for effectively supporting electronic negotiations”. In the case of non-monolithic negotiations, a third aspect has to be taken into account: group dynamics. By extrapolating the case of cooperation from the case of e-negotiations, we argue that three elements have to be taken into account for an efficient support for non-monolithic cooperation: action management, communication management, and group dynamics. These three elements are required for computer support systems for non-monolithic cooperation processes as:

- the “acting” element is required to model the actions that the collaborators trigger during the cooperation process,
- the communication management element is required to model the meta-data about the cooperation processes,
- the group dynamics element is required to model social structures in which the cooperation takes place, i.e. cooperation groups.

The three elements presented above are not conclusive. Other components of a collaborative system, e.g computer support for proposition offer evaluation in the case of e-negotiations, may be included in the proposed model. However, it should be noticed that the three elements presented above are inherently related with the non-monolithic aspect of the cooperation processes, while other components are usually related with a collaborator's perceptions/goals concerning the cooperation process.

In this paper, we present a model which provides support for multiple collaboration protocols for non-monolithic collaboration processes. In Section 2, a model for collaboration protocols integrating communicative, “acting”, and social aspects is presented, then group actions required to provide support for group dynamics are introduced. In Section 3, the proposed model is formalized. In Section 4, the question of protocol validity is addressed. Section 5 concludes the paper.

2. STRUCTURING NON-MONOLITHIC COLLABORATION PROCESSES

In non-monolithic collaborative processes, collaboration always occurs inside a group. Even when a single collaborator works alone, it may be considered as a group consisting of only herself/himself. Therefore, it may be stated that *a group is a non-empty set of collaborators*. An other aspect of this kind of collaboration is that collaborators are collaborating via message exchange. As we would like to structure non-monolithic collaboration processes, we have to address two issues: first, a mechanism to structure collaboration inside a given group has to be proposed, which means that message exchange has to be structured, second, group dynamics have to be addressed.

2.1 Collaboration Protocols

Three elements may be distinguished in collaborative processes: a meta-data aspect, an “acting” aspect, and a social aspect.

Meta-data is a major component of collaboration as collaborators need to exchange information to achieve their common goal [15][11]. The acting aspect of collaboration concerns the fact that collaborators not only exchange information to reach their common goal, but also act to achieve it. Finally, the social aspect of collaborative processes, captured via the concept of role, concerns relationships among collaborators, the perceptions they have of others collaborators.

The concept of *social behavioral element* captures all three aspects – meta-data, acting, and social – of collaborative processes.

Social Behavioral Element A social behavioral element is a triplet (User_Role, Meta-data_Type, Action).

- The User_Role addresses the social aspect.
- The Meta-data_Type addresses the meta-data aspect. The introduction of meta-data types allows to limit ambiguousness of communication [12].
- The Action addresses the acting aspect.

In the proposed model, collaboration processes result from exchange of social behavioral elements among collaborators. Collaborators are exchanging social behavioral elements, sending typed meta-data and acting, in a given role. Exchange of social behavioral elements causes the evolution of the group in which collaborators are working: each sent social behavioral element causes a transition of the group from a past state to a new state.

Transition A transition is a triplet (SocialBehavioralElement, SourceState, DestinationState).

It is now possible to define collaboration protocols, which may be used to structure collaboration processes.

Collaboration protocol A collaboration protocol consists of a set of transitions, a start state, and a set of terminating states.

One may notice that a protocol is a variant of finite state machines. A finite state machine (FSM) is usually defined as “a model of computation consisting of a set of states, a start state, an input alphabet, and a transition function that maps input symbols and current states to a next state”. The set of states of the FSM can be easily deduced from the set of transitions of the protocol. The start state occurs in both the FSM and the protocol. The input alphabet of the FSM is the set of social behavioral elements which appear in all transitions of the protocols. Finally, the transition function of the FSM is defined by the set of transitions of the protocol. The only difference between FSMs and collaboration protocols is the existence of terminating states for protocols.

A collaboration protocol is a template definition for a set of collaboration processes. Using an analogy with object-oriented programming, one may say that a collaboration protocol is to a protocol instance what a class is to an object. In a given group, a given protocol instance regulates collaboration among group members/collaborators.

Protocol instance A protocol instance is a tuple (Protocol, CurrentState, UserToRoleMapping). The UserToRoleMapping is a function which associates a UserRole with a given user.

2.2 Group Dynamics

In non-monolithic collaborative processes, groups evolve: a collaborator may join or leave an existing group, a group may split in two or more groups, two or more groups may merge into a single group. Group dynamics may be modeled by a set of group actions. The following group actions have been identified:

- create action: creates a new group;
- join action: adds an author to the set of collaborators of an existing group;
- merge action: creates a new group consisting of the union of the set of collaborators of at least two groups;
- end action: deletes an existing group;
- leave action: removes a collaborator from the set of collaborators of an existing group;
- split action: creates at least two groups from an existing group and the union of the sets of collaborators of the created groups equals the set of collaborators of the existing group.

Group actions are illustrated on Figure 1. Dots represent collaborators while circles represent groups. One may notice that, as shown on Figure 1 for the split and merge actions, a given collaborator may participate at a given time in many groups.

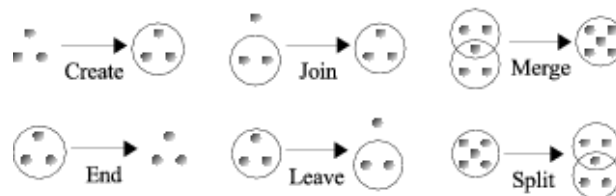


Figure 1. Group actions

3. FORMAL MODEL

Let's first define the concepts of behavioral element and roles, which are the fundamental bricks of group protocols in the proposed model. The concept of *behavioral element* is required to model relationships between actions and meta-data types.

Behavioral element A behavioral element is defined as a couple (Action, Meta_data_type)

The relationship between actions and meta-data types is a *n-to-n* relationship. First, a 1-to-n relationship between action and meta-data types may exist: in an electronic negotiations case, one action – the leave action, triggered by a negotiator which wants to leave an existing group – may be associated with various meta-data types, such as a `rejectOffer`, when a negotiator wants to send a message concerning offer rejection, or a `logout` when a negotiator wants to log out the system. Reciprocally, a 1-to-n relationship between meta-data type and actions may exist: a meta-data type – the `rejectOffer` meta-data type, triggered by a negotiator which does not accept a previously proposed offer – may be associated with various actions, such as a leave action to leave an existing group or an end action to delete an existing group. The concept of behavioral element enables *n-to-n* relationships between actions and meta-data types.

Role A Role r consists of a set of behavioral elements, denoted r_{be} , and a role name, e.g. buyer, seller, or moderator.

The relationship between behavioral elements and roles is a *n-to-n* relationship, as presented in Figure 2. First, in the presented example, one role – the moderator role – may be associated with various behavioral elements, such as “send message”, “accept message”, or “reject message”. Second, on the same example, the behavioral element “send message” is associated with many roles, such as moderator or normal negotiator.



Figure 2. Example of *n-to-n* relationship between behavioral elements and roles

Social Behavioral Element For a given role r , the set of pairs ($role_name; be$), with $be \in r_{be}$ is denoted r_{sbe} . A pair ($role_name; be$) is called a *social behavioral element*.

Using the example presented in Figure 2, the social behavioral element (moderator, accept message) is defined, while the social behavioral element (normal user, accept message) is not defined. For the role moderator, r_{sbe} is a set containing the following social behavioral elements: (moderator, accept message), (moderator, send message), (moderator, reject message).

Group Protocol A group protocol p consists of:

- a set of states S ,
- one starting state $s_0 \in S$,

- a set of ending states $S_{ending} \subset S$, with $s_0 \notin S_{ending}$,
- a set of roles R ,
- a transition function t from $(S - S_{ending}) \times S \times \bigcup_{r \in R} r_{sbe}$ to $\{true, false\}$.

4. VALIDITY OF COLLABORATION PROTOCOLS

The former definition specifies the basic requirements for collaboration protocols. It links the notions of states, roles, and behavioral elements. However, the former definition does not ensure that such protocols are valid, neither structurally, nor semantically.

Structural validity A protocol is *structurally valid* iff:

- 1) there is a path from the starting state to all states,
- 2) there is a path from every state to an ending state.
- 3) from a given state, there is only one transition associate with a given social behavioral element.

The two first conditions ensure that no state are neither non-accessible (and therefore should be suppressed from the protocol), nor leading to a lock in the negotiation process, i.e. forbids the negotiation process to be finished.

The third condition ensures that no ambiguity exists in a group protocol. An ambiguity may occur when two or more transitions associated with one social behavioral element sbe may lead to two or more states from a single state s . In this case, it is not possible to decide to which state the social behavioral element sbe leads to. Formally, the last condition for structural validity may be formulated as follows:

$$\forall s \in (S - S_{ending}), \exists (s_i, s_j) \in S^2, sbe \in \bigcup_{r \in R} r_{sbe}, \\ t(s, s_i, sbe) = t(s, s_j, sbe) = true \Rightarrow s_i = s_j$$

Semantical validity A protocol is *semantically valid* iff:

- 1) all transitions to ending states are associated with social behavioral elements containing an ending action,
- 2) social behavioral elements containing an ending action are associated only with transitions leading to ending states,

An action a is an *ending action* iff the life of the group g ends when action a is called in group g . The life of a group g ends when no more message can be sent to group g . The set of ending actions of a given protocol is denoted *EndingActions*. One may notice that the group action end is obviously an ending action.

The first condition ensures that a transition leading to an ending state really ends the life of the group. The second condition ensures that the group protocol cannot be “interrupted” by a transition associated with an ending action.

Formally, the first condition for semantical validity may be formulated as follows:

$$\forall (s_i, s_j, sbe_k) \in (S - S_{ending}) \times S_{ending} \times \bigcup_{r \in R} r_{sbe}, \\ t(s_i, s_j, sbe_k) = true \Rightarrow a_k \in EndingActions$$

where a_k is the action associated with the social behavioral element sbe_k .

The second condition for semantical validity may be formulated as follows:

$$\forall (s_i, s_j, sbe_k) \in (S - S_{ending}) \times (S - S_{ending}) \times \cup_{r \in R} r_{sbe},$$

$$t(s_i, s_j, sbe_k) = true \Rightarrow a_k \notin EndingActions$$

The concepts of structural validity and semantical validity are illustrated on Figure 3. The case a) is an example of a valid group protocol, while the cases b) and c) are respectively examples of structurally and semantically invalid group protocols. In the case b), the group protocol is structurally invalid for two reasons: first, there is no path from the starting state to the state s_2 . Second, there is no path from the state s_1 to the ending state s_4 . In the case c), the group protocol is semantically invalid for two reasons: first, the transition t_1 , which is not associated with an ending action, leads from the state s_1 to the ending state s_4 . Second, the transition t_2 , which is associated with an ending action, leads from the state s_2 to the state s_3 , which is not ending state.

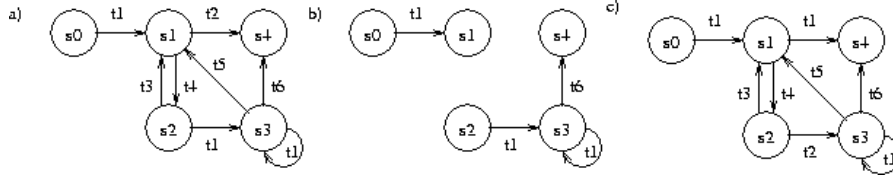


Figure 3. Examples of a) valid, b) structurally invalid, c) semantically invalid group protocols, where s_0, s_1, \dots, s_4 are states, s_4 being the only ending state, and t_1, t_2, \dots, t_6 are transitions, t_2 and t_6 being the only transitions associated with an ending action

5. CONCLUSIONS

The introduction of collaboration protocols and group actions allows to provide computer support to non-monolithic collaboration processes. To our best knowledge, it is the first model for electronic support for non-monolithic collaborative processes addressing the issues of group dynamics, meta-data management and protocol validity.

It would be possible to build complex support systems for complex collaborative processes using the model presented in this paper. The design of systems for non-monolithic collaboration processes may be resumed in the following steps: first, the roles involved in the collaboration process have to be identified. Next, the required actions have to be implemented. Then, meta-data types should be defined. Therefore, social behavioral elements may be defined. Finally, collaboration protocol(s) may be specified and their validity may be checked.

The presented model could be used in a broad spectrum of potential applications. The presented model may for instance be applied to non-monolithic negotiations, such as international negotiations or business-to-business contract establishment. Another field of applications is the legislative process in which various political parties, potentially presenting various opinions, are (or should) collaborate in order to establish laws in form of new or modified legal acts. The presented model could also be used to design support systems for collaborative documentation edition processes that often takes place between business actors.

Among future works, it would be interesting to investigate the possibilities to em-

bed a protocol instance into another protocol instance. This would allow to modularize protocols, to design protocols using smaller protocols, to develop protocol libraries. Another field which could be the object of future works is the concept of role. The addition of relationships between various roles, such as inheritance or composition, would be an interesting work to be done. A prototype has already been implemented [9] and it would be interesting to evaluate the usability of the proposed solution in real-world cases.

6. REFERENCES

- [1] M. Benyoucef and R. K. Keller, "An Evaluation of Formalisms for Negotiations in E-Commerce", Proceedings of the Workshop on Distributed Computing on the Web, Quebec City, Canada, pp. 45-54, 2000.
- [2] W. Cellary, W. Picard, and W. Wiczerzycki, "Web-based Business-to-Business Negotiation Support", Int. Conference on Electronic Commerce TrEC-98, Hamburg, Germany, pp. 80-89, 1998.
- [3] M. Ettorre, L. Pontieri, M. Ruffolo, P. Rullo, and D. Sacca, "A Prototypal Environment for Collaborative Work within a Research Organization", 3rd Workshop on Web Based Collaboration, associated to the DEXA Conference, Prague, Czech Republic, pp. 274-279, 2003.
- [4] P. Hung and J.-Y. Mao, "Modeling of e-Negotiation Activities with Petri Nets", 35th International Conference on System Sciences, HICSS'02, Hawaii, USA, 2002.
(<http://csdl.computer.org/comp/proceedings/hicss/2002/1435/01/14350026.pdf>)
- [5] G. E. Kersten and G. Lo, "Aspire: Integration of Negotiation Support System and Software Agents for E-Business Negotiation", *International Journal of Internet and Enterprise Management (IJEM)*, 1(3), pp. 293-315, 2003.
- [6] G. E. Kersten, S. E. Strecker, and K. P. Law, "Protocols for Electronic Negotiation Systems: Theoretical Foundations and Design Issues", 5th International Conference on E-Commerce and Web Technologies, EC-Web 2004, Zaragoza, Spain, pp. 106-115, 2004.
- [7] J. Kim and A. Segev, "A Framework for Dynamic eBusiness Negotiation Processes", IEEE Conference of E-Commerce: CITM, 2003.
(<http://groups.haas.berkeley.edu/citm/citm-home.htm>)
- [8] W. Picard, "Towards Support Systems for Non-Monolithic Collaborative Document Edition: the Document-Group-Message Model", 15th International Conference on Database and Expert Systems Applications DEXA 2004, Zaragoza, Spain, pp. 266-270, 2004.
- [9] W. Picard and T. Huriaux, "DynG: Enabling Structured Non-Monolithic Electronic Collaboration", The 9th International Conference on CSCW in Design, Coventry, UK, 2005, to be published.
- [10] H. Raiffa, J. Richardson, and D. Matcalf, *Negotiation Analysis, The Science and Art of Collaborative Decision Making*, The Belknap Press of Harvard University Press, 2002.
- [11] M. Schoop, A. Jertila, T. List, "Negoisst: A Negotiation Support System for Electronic Business-to-Business Negotiations in E-Commerce", *Data and Knowledge Engineering*, 47(3), pp 371-401.
- [12] M. Schoop, "An Introduction to the Language-Action Perspective", *SIGGROUP Bulletin*, vol. 22, No.2, August 2001, pp 3-8.
- [13] M. Schoop and C. Quix, "DOC.COM: a framework for effective negotiation support in electronic marketplaces", *Computer Networks*, vol. 37, No. 2, pp. 153-170, 2001.
- [14] M. Schoop and C. Quix, "DOC.COM, Combining document and communication management for negotiation support in business-to-business electronic commerce", Proceedings of the 34th Hawaiian International Conference On System Sciences (HICSS-34), Maui, Hawaii, Jan. 2001, IEEE Press.
- [15] H. Weigand, M. Schoop, A. De Moor, F. Dignum, "B2B Negotiation Support: The Need for a Communication Perspective", *Group Decision and Negotiation* 12, pp. 3-29, 2003.