CSAC 2004

José Cordeiro and Joaquim Filipe (Eds.)

Computer Supported Activity Coordination

INSTICC
Press

José Cordeiro and
Joaquim Filipe (Eds.)

Computer Supported Activity Coordination

Proceedings of the
1st International Workshop on
Computer Supported Activity Coordination,
CSAC 2004
In conjunction with ICEIS 2004
Porto, Portugal, April 2004

Volume Editors

José Cordeiro (jcordeir@est.ips.pt)
Escola Superior de Tecnologia de Setúbal
Instituto Politécnico de Setúbal
Setúbal, Portugal.

and

Joaquim Filipe (j.filipe@est.ips.pt)
Escola Superior de Tecnologia de Setúbal
Instituto Politécnico de Setúbal
Setúbal, Portugal.

Activity Coordination in Collaborative Learning Environments

Carlos José M. Olguín[3,1], Alberto B. Raposo[2] and Ivan Luiz M. Ricarte[1]

[1] UNICAMP / FEEC / DCA
Caixa Postal 6101
13083-970 – Campinas, SP – Brazil
{olguin, ricarte}@dca.fee.unicamp.br
[2] PUC-Rio / DI / Tecgraf
R. Marquês de S. Vicente, 225
22453-900 – Rio de Janeiro, RJ – Brazil
abraposo@tecgraf.puc-rio.br
[3] UEM / CTC / DIN
Av. Colombo 579 – Bloco 019
87020-900 – Maringá, PR – Brazil
olguin@din.uem.br

Abstract. In the context of computer-supported collaborative learning, discussions are essential to increase the knowledge level of the members of a group. This work proposes the modeling of the discussion activities of a study group using an activities coordination model. Starting from the formal modeling of the system, and using coordination mechanisms based on Petri Nets, the behavior of the environment can be simulated and analyzed. These simulations allow anticipating possible problems and help to turn interactions among students more efficient.

1 Introduction

Studying techniques for computer-supported collaborative learning, some researchers have been working in the construction of systems to monitor collaboration automatically. Barros and Verdejo [1] defined a method to calculate a set of attributes that characterizes the individual and group behavior. Using labeled textual contributions and fuzzy logic the degree of collaboration of the group is calculated. Mühlenbrock and Hoppe [2] intended to consider users' actions in a collaborative work environment as the basis for the qualitative analysis of activities. This approach makes possible to derive descriptions of group activities through a method to recognize them.

Delgado et al. [3] defined a model for monitoring the activities accomplished inside a study group. In that work it was proposed that the actions and the monitored interactions were stored, as contributions, in a discussion graph. Through the analysis of this information by a human tutor, it would be possible to infer the cognitive process used by the student during the discussion. Extending that work, the objective of the present work is to model discussion activities, using the coordination model based

on Petri Nets (PN) proposed by Raposo and Fuks [4]. The resulting model is suscep-tible to simulations and analyses that allow the anticipation of problems. It makes possible the formal verification of the model, and helps to turn more efficient the students' interactions during the discussion process. Furthermore, starting from the formal model of the system, software components can be implemented to give sup-port to the discussion of the group.

This work is organized as follows: Section 2 presents the guidelines of a develop-ment methodology of collaborative learning environments that gives special attention to the coordination of tasks that are executed in the environment. This section also describes the model adopted to capture group discussions; Section 3 introduces as-pects of the adopted coordination model; Section 4 presents coordination mechanisms based on PN; and the final section presents the conclusions of the work with empha-sis in the directions to be followed.

2 The Development Methodology and The Model of Discussion

Besides defining the requirements and pedagogic objectives of collaborative learning environments, a development methodology of collaborative learning environments should consider the following activities:

1) Identification of the tasks that will be executed in the environment and their in-terdependencies;

2) Identification and modeling of the coordination mechanisms among those tasks;

3) Analysis of the model developed in (2) whose results can take back to (1);

4) When the result of the analysis (3) is satisfactory, to accomplish the project and the construction of the environment.

In this work the considered environment is the message exchange that happens, within a study group, when discussing about a certain subject, with the intention of increase the knowledge level that the members of the group have about that subject.

We will concentrate our attention on activities (1), (2) and (3) of the methodology. Our interest at this moment is the modeling of activities, through the use of a model that considers coordination aspects among the tasks. In this paper, we consider a model of an environment for the exchange of textual messages in a study group [3]. This model assumes that, during group activities, students use collaborative services of the environment to generate some kind of information that others can comment or discuss using the same services. All information or comment directly associated to the discussion is defined as a *contribution*. The contribution is the basic element used to monitor group activities. The model allows the quantification of accepted contribu-tions and the maintenance of a discussion log, represented by a sequence of contribu-tions.

A discussion is represented as a sequence of contributions. Contributions can be: 1) accepted or rejected; 2) substituted or reconsidered by another contribution; and 3) doubts, questions or answers to a contribution. The registration of contributions is organized in a discussion graph.

The discussion process of the group involves the concept of "knowledge negotia-tion" that is an important aspect in the construction and the management of the

knowledge generated by the study group. This process embraces a series of collaborative interactions among students, for example, to clear the meaning of some terms used in the placement of the contributions, to discuss alternatives, etc [5].

The dynamics of the discussion model shows the need to coordinate the activities. In the next section is presented the coordination model of activities used in this work.

3   The Coordination Model - Interdependencies

In our context a collaborative activity is defined as a set of interdependent tasks, executed by several actors in order to reach a common goal. In the example of our work, students that are part of the discussion group execute a series of interdependent tasks (for example, to make contributions, to decide about the acceptance or not of contributions as valid knowledge, etc.) with the objective of increasing the knowledge level on the discussed subject.

The interdependency concept is fundamental in the coordination theory. It is possible to characterize different types of interdependencies and identify the coordination mechanisms that manage them, creating a set of interdependencies and related coordination mechanisms capable to encompassing a wide range of collaborative applications [6].

3.1   The Basic Temporal Interdependencies

Temporal interdependencies establish the relative order of execution among a pair of tasks. The set of temporal interdependencies of the proposed model is based on the temporal relations defined by Allen. He proved that there is a set of primitive and mutually exclusive relations that could be applied over time intervals [7]. A time interval is characterized by two events, which in turn are associated to time instants. The first event is the beginning of the interval A, denoted i_a. The other event is the ending of the same interval, denoted f_a. According to Allen, a set of seven primitive relations may maintain temporal information on any pair of time intervals A and B (*A equals B*, *A starts B*, *A finishes B*, *A meets B*, *A overlaps B*, *A during B* and *A before B*). Based on these relations a group of axioms is defined to create a temporal logic. The fact of being applied over time intervals (and not over time instants) made the above relations suited for task coordination purposes, because tasks are generally non-instantaneous operations. The adaptation of Allen's primitives to the context of collaborative activities considers that any task T will take some time to be executed.

Allen's temporal logic is defined in a context where it is essential to have properties such as the definition of a minimum set of basic relations, the mutual exclusion between these relations and the possibility to get conclusions starting from them. However, temporal interdependencies among collaborative tasks are inserted in a different context. The important fact here is the management of the interdependencies and the appropriate understanding of the collaborative activity.

A limitation of Allen's relations is that they are merely descriptive, in other words, they don't express causal or functional relations among the intervals [8]. For all these

reasons, it was necessary to do some adaptations to Allen's basic relations. The goal of the proposed extensions is to offer a larger set of possibilities to create coordination mechanisms that may control many different situations.

3.2 Active and Passive Interdependencies

The merely descriptive characteristic of Allen's temporal relations allows different interpretations for the interdependencies. For instance, consider that two tasks, *Ta* and *Tb*, are related by the interdependency *Ta equals Tb*. In the coordination context, this interdependency can be interpreted in two different ways. In the first case, called *active interpretation*, this relation express that the beginning of a task should begin the other; and that the end of one of the tasks should conclude the other task. The second possible interpretation for any coordination mechanism is called *passive interpretation*. In this case, the coordination mechanism expresses a group of conditions that should be obeyed to take the activity to the end.

To deal with active and passive interpretations, two operators were defined: *enables* and *forces*. The operator *enables* represents the passive interpretation, while *forces* represents the active interpretation. These operations can be applied in the initial and final moments of each interdependent task. Additionally, these extreme points have two states: *ready* and *concluded*, indicating, respectively, that the task is ready to begin (or to finish) and that the task already began (or finished).

In order to exemplify the passive and active interpretations of the temporal interdependencies, consider the situation in which two students, A and B, have to vote for the acceptance or rejection of a contribution. We will denominate *Tva* and *Tvb* the tasks of voting associated to students A and B, with initial and final points i_{va}, i_{vb}, f_{va} and f_{vb} respectively. Imagine now that in our environment the voting process begins in a synchronous way. In this case, the interdependency associated to the tasks is *Tva starts Tvb* that can be extended in several interpretations, such as:

i_{va} *(ready) enables* i_{vb} *AND* i_{vb} *(ready) enables* i_{va} – this declaration indicates the passive situation in which the tasks will only begin their execution when both are ready (i.e., *Tvb* will only be able to begin when *Tva* is ready to begin, and vice-versa), but none will force the execution of the other.

i_{va} *(ready) forces* i_{vb} – in this situation, when *Tva* is ready to begin, *Tvb* will be forced to begin, indicating an active interdependency of the type "master/slave" (in the same way, *Tvb* could be considered the master task if i_{vb} *(ready) forces* i_{va}).

i_{va} *(ready) forces* i_{vb} *AND* i_{vb} *(ready) enables* i_{va} – *Tva* is the master task, since it forces the beginning of task *Tvb*, but *Tva* will only begin when *Tvb* is ready.

4 Coordination Mechanisms based on Petri Nets

Coordination mechanisms based on PN were developed to coordinate the set of interdependencies presented. The formal modeling allows the designer to anticipate and test the behavior of the environment. PN support models at different levels of abstraction and are appropriate for simulation and formal verification.

In this proposal, the project of a collaborative learning environment is divided in three hierarchical levels: workflow, coordination, and execution. In the workflow level, the interdependencies among the tasks attributed to the actors of the environment are established (activity 1 of the development methodology presented in Section 2). The coordination level is built from the workflow level, through the expansion of the interdependent tasks and the insertion of the corresponding coordination mechanisms (activity 2 of the methodology). The model of the environment is simulated and analyzed in this level (activity 3). The execution level deals with the execution of the tasks in the environment (activity 4).

During the passage from the workflow to the coordination level, each task that has interdependency with another is expanded in a sub-net, as presented in Figure 1. In this model, events *i* and *f* (beginning and end of the task) are represented as transitions, while the states *ready* and *concluded* are represented as connected places to the respective transitions. After having triggered event *i*, the flow is divided in two parallel flows, one that indicates that the task is in execution – *i(concluded)* – and another that represents the interaction with the execution of the task in the system. The task execution is modeled through a transition with token reservation (represented with the letter "R") that is a non-instantaneous transition – the tokens are removed from their input place when the transition is triggered and only some time later is increased to their output places, representing in this way the duration of the task.
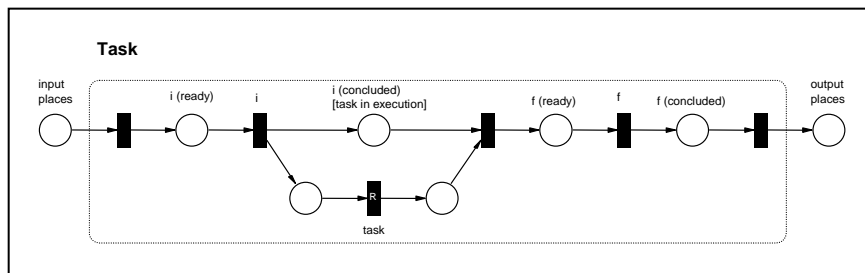

Figure 1. PN representation of an interdependent task at the coordination level.

When considering two related tasks for interdependencies, it is necessary to connect the places and the transitions of both models correctly to create the corresponding coordination mechanisms. To do this, it is necessary to define how to map the previously defined operators and parameters to the PN model. The mapping of the *forces* and other operators to the PN model are presented in [4].

5 Conclusions

In this paper, a proposal for the use of an activities coordination model of a study group was presented with the intention of turning students' interactions more efficient. This model offers a certain degree of flexibility through the separation between tasks and their interdependencies and it is adapted to deal with some interoperability aspects, since the set of interdependencies is generic and the implementation of the coordination mechanisms can be accomplished using any tool. This coordination

model is quite appropriate to represent interactions that occur among the members of a discussion group. Starting from the resulting model, software components can be derived to give subsidies to the discussion of the group. The software components will allow standardize the interactions between tasks and coordination mechanisms associated to them in a way that does not depend on the implementation.

Although the presented coordination mechanisms appropriately represent the interdependencies among tasks that compose a collaborative activity, the resulting model is static. This characteristic of the model does not allow that alternative situations could be represented. In this sense, the continuation of the work is centered in the extension of the presented coordination mechanisms in order to turn them more flexible. A possibility would be the definition of new operators that will consider concepts of fuzzy logic to deal with alternative situations.

References

1. Barros, B., Verdejo, M.: An approach to analyze collaboration when shared structured workspaces are used for carrying out group learning processes. In: Proc. of the 9th Intl. Conf. on Artificial Intelligence in Education, Lemans, France (1999) 449-456
2. Mühlenbrock, M., Hoppe, H.: A Collaboration Monitor for Shared Workspaces. In: Proc. of the 10th Intl. Conf. on Artificial Intelligence in Education, San Antonio, Texas, USA (2001)
3. Delgado, A.L.N., Olguín, C.J.M., Ricarte, I.L.M.: Monitoring Learners Activities in a Collaborative Environment. In: Proc. of the 7th Intl. Workshop on Groupware, Darmstadt, Germany (2001) 148-152
4. Raposo, A.B., Fuks, H.: Defining Task Interdependencies and Coordination Mechanisms for Collaborative Systems. In: Blay-Fornarino, M., Pinna-Dery, A. M., Schmidt, K., Zaraté, P. (eds.): Cooperative Systems Design. Frontiers in Artificial Intelligence and Applications, Vol. 74. IOS PRESS, Amsterdam (2002) 88-103
5. Stahl, G.: The Model of Collaborative Knowledge-Building. In: Proc. of the 4th Intl. Conf. of the Learning Sciences, Ann Arbor, USA (2000) 70-77
6. Malone, T.W., Crowston, K.: The Interdisciplinary Study of Coordination. *ACM Computing Surveys*, vol. 26, no. 1 (1994) 87-119
7. Allen, J.F.: Towards a General Theory of Action and Time. *Artificial Intelligence, vol.* 23 (1984) 123-154
8. Duda, A., Keramane, C.: Structured Temporal Composition of Multimedia Data. In: Proc. of the Intl. Workshop on Multi-Media Database Management Systems, Blue Mountain Lake, USA (1995) 136-142