Group Activity Database for Groupware Evolution

Hiroyuki Tarumi[1], Tetsuya Matsuyama[2], Yahiko Kambayashi[1] *

[1] Graduate School of Informatics, Kyoto University
[2] Department of Information Science, Kyoto University

Abstract. This paper proposes a new type of database for collaborative work. Known types of databases for collaborative work are databases of shared data and those of process data. A new type proposed in this paper is Group Activity Database (GADB). It contains history of group communication, which is used to analyze collaborative work. With a GADB, evaluation, user modeling, reengineering, simulation, and team awareness support will be helped. Evolution of groupware is supported by these activities. We adopt agent-based groupware architecture in order to realize open and distributed groupwork management. With this architecture, communication logs among agents or between agents and users are captured by a GADB for analysis.

1 Introduction

Research of *CSCW* (Computer Supported Cooperative Work) has been active for over ten years. So-called *groupware* products have been developed, e.g., enhanced e-mail systems, bulletin board systems, workflow or business process support systems, shared editors and hypermedia, collaborative writing systems, video conferencing systems, etc.

From the viewpoint of databases, two kinds of them have been important for CSCW. One is the type of databases that contain shared data on which the collaborative work is applied. This kind of database includes shared hypermedia systems[13, 20], repository database for software development (e.g., [14]), etc.

Databases of the other type contain *process data*. For example, a workflow management system may have a database that contains workflow process definitions and status of workflow cases.

In this paper, we propose a new category of databases for new age groupware: *Group Activity Database (GADB)*. It is purposed to support evolution and deployment of groupware systems.

2 Group Activity Database

An ideal GADB should be able to capture all histories of group activities. Histories include communication logs among groupwork participants, supporting

* {tarumi,matuyama,yahiko}@kuis.kyoto-u.ac.jp

software modules, and environments (outside of the working group and supporting system) as well as other system logs.

The purpose of GADB is mainly analyses of groupwork. Analyses benefit groupware designers and users in the following ways.

1. Evaluation:
   The data indicates how the groupware system is used, how people behave with the system, and how the work has changed in terms of efficiency. Designers and users can evaluate the groupware system based on these data.
2. User Modeling:
   Groupware design depends strongly on the user model, which abstracts user behaviors, e.g. when the user starts processing tasks, in what order, and when finishes them. The log data tell us other informal user behaviors like forwarding task requests to another person, etc. According to these data, the designer can improve the quality of user models in order to fit them to the reality.
3. Reengineering:
   Like user modeling, processes can also be remodeled based on the log data. For example, skipping or reversing the task processing order often occurs in reality. Professionals can sometimes find process bottlenecks from the log data. These findings can give a better process definition. This is so-called *business process reengineering (BPR).*
4. Simulation:
   After reengineering design, the new business process should be validated prior to implementation. Simulation is one of the promising tools of validation. To enhance the reality of simulation, user models obtained from the log data would be useful.
5. Team Awareness:
   Team awareness[18] is a very important concept in cooperative work. It means enabling people to be aware of how and what other team members are doing. In order to support team awareness, visualizing the current status of other members and communications among them is important. The log data, especially very recent data, can be given to the visualizing process. It shows users what task each member is doing, how many backlogs each member has, etc.

A process database may also contain log data (e.g., EvE[5]). For example, workflow case status data are log data that indicate when and by whom these cases have been processed. In this sense, a workflow database can be regarded as a GADB. However, we insist that an exclusive database for the purpose of group activity record should be prepared. Reasons are as follows:

– A workflow database is dedicated to the workflow tasks managed by the system. Other informal communications are ignored.
– Basically, most of workflow databases cannot record histories that ignored the formal process definitions. For example, skipping or reversing the processing order may not be able to be recorded correctly.

– Sometimes a worker is concurrently doing jobs for multiple business processes. If workflow databases are used, the worker's activities are recorded separately on each database, so that user modeling would be difficult.

The next discussion is how to implement a GADB. The most difficult problem is capturing log data of all group activities that may be separately managed. We do not think that a GADB is applicable to all current groupware products. We assume agent-based groupware. It is not only to realize GADB, but because agent-based groupware has many promising features as the next generation groupware. The next section describes the agent-based groupware.

3 Agent-based Groupware

3.1 Features

In this paper we use the word "agent" in the following sense. An agent is a software module that is a surrogate for a user, or implements functions for an aspect of group activities (cost optimization, mediation of people, etc.). Agent can communicate with another agent by exchanging messages.

Agent-based Groupware is an architecture of groupware consisting of multiple agents. Multi-agent systems are very suitable to implement groupware due to the following features.

1. Openness:
   This basic feature means that it is widely applicable to large companies that have many divisions, joint ventures, loosely organized groups like those of alumnus/alumna or volunteers, and virtual enterprises. Concretely it means that the groupware satisfies the following conditions:
   – Each user can belong to multiple workgroups concurrently. To realize this, supporting functions are needed to enable users to concurrently take part in multiple business processes[8], and also personal tasks[9].
   – It positively utilize wide area networks like the Internet, intranets, and extranets.
   – Inter-organizational security issues are consciously maintained.
   – Distributed management is possible. It is important to allow each manager of divisions or sub-organizations to manage her or his group in her or his own way.
   – The system accepts the cultural differences among different sub-groups. For example, each sub-group has its own glossaries[18], constraints in schedule, job priorities, and management styles.
2. Scalability:
   It should be possible to append new elements to the system, or remove elements from it. It should also be possible for the system to run efficiently regardless of its size.
3. Evolution:
   It should be dynamically adaptable to exceptional events as well as to accept evolutionary process refinement or reengineering.

3.2 Existing Examples

The number of already existing agent-based groupware is not so many, but some of them are already in use.

Pan, et al. gave a very early example[17]. They give a framework of agent-based enterprise modeling. A enterprise is modeled with many discrete elements, each of which is an IA (Intelligent Agent) or a PA (Personal Assistant). The system was applied to a semiconductor fabrication process, in which case examples of IAs are machine control and monitoring processes. A PA can also be regarded as an agent that interfaces a user and IAs.

ADEPT project applied a multi-agent technology to the BT company's customer quote business processes[11, 12]. In its architecture, each division involved in the business process is represented as a group of agent (agency). They claim that agent architecture has the following benefits.

- Process management and resource management are conducted at the same time with one architecture.
- Exception handling is easy.
- It has robustness and scalability thanks to distributed architecture.

MACIV project in Portugal adopted a multi-agent technology for the resource management of civil construction companies[4]. Construction companies have many sites all over the countries. The project settled LAN at each site and regards a site as an agent.

INA/LI is designed with a multi-agent architecture supporting both group and personal work[9]. With this architecture, the WorkWeb system was proposed [22]. It realizes dynamic adaptation of workflow to exceptional and unpredicted situations, like sudden absence of a worker. The adaptation process is designed with negotiations among workflow management agents and human interface agents. This paper also proposes the concept of *Business Process Tactics* in contrast to business process reengineering. Business Process Tactics means dynamic, case-based change of a process, while reengineering changes the process definition.

3.3 Our Position

In this research, our assumption on the agent-based groupware is like those found in [17] and [9]. An interface agent for each user is mandatory because we should maintain user models. Communications between a user and its interface agent should be recorded as well as communications among agents.

Fig. 1 shows the concept of GADB, capturing all communication logs among agents and those between agents and users.

3.4 Data Capturing

It is rather easy to record communications among agents because of very formal protocol definitions. Captured data items are, generally, like Message ID, source

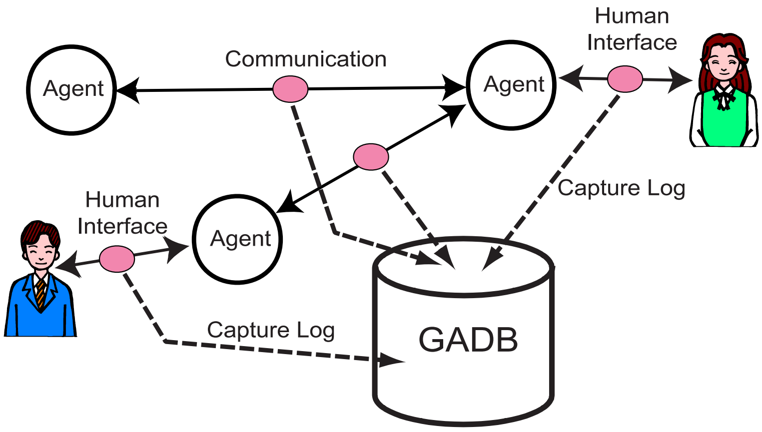

Fig. 1. Group Activity Database

and destination of messages, type of communication (task order, inquiry, cancellation, request for deadline extension, etc.), task name, time, priority value, deadline, etc. These data indicate history of processes.

Besides these data, there are more data items that depend on each application. At the time of agent design, a support system that can automatically define the database schema for application data should be given.

On the other hand, it is difficult to confirm correspondence of logged data of interaction between users and agents with group tasks, because interaction-level data are too fine-grained and including noisy information like meaningless mouse movement. The architecture of an user interface agent should be like Fig. 2.

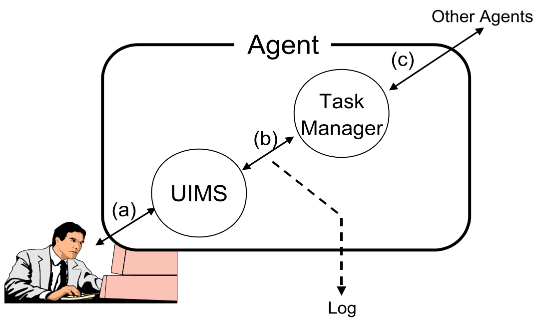

Fig. 2. Internal Agent Architecture to Take Interaction Logs

In this architecture, the concept of UIMS (User Interface Management Systems) is adopted. UIMS transforms device-level, machine-dependent UI operations into abstract level UI operations. It enables to get abstracted level interaction between user and the agent at (b).

Task Manager takes the role of user's surrogate. It may automatically answer requests from other agents. Hence communication logs between agents can be obtained at (c).

Logged data at (b) include:

1. Subset of logged data at (c).
2. User's manual input, like inputting a task description to his or her schedule management agent in case that the task is ordered off-line.
3. Information request from the user to the agent, and its answer.
4. Active message from the agent to the user, e.g. an urging message.

Note that logged data which appear at (c) but do not appear at (b) are those related to user requests resolved at the agent internally.

3.5 Requirements to GADB

A GADB is used both at the time of simulation and runtime operation. In both cases the same visualization feature can be utilized. However, at the time of simulation, it might be easier to acquire data because the simulation engine can simulate all agents and users on one machine.

GADB should be able to answer at least the following types of queries.

- For BPR and User Modeling
  - Give all log data related to a process.
  - Give all log data related to a worker, and sort them chronological order.
  - Give past tasks that could not keep deadlines.
  - Give a list of BPT activities that happened within a time period, or related to a given process.
  - Give all log data related to a given BPT activity.
  - Give all task requests that a given worker has at a given time.
- For Simulation
  - Give all task requests that a given worker has.
- For Visualization (at simulation and runtime)
  - Give all task requests that a group of workers have at the present time.
  - Give a list of workers involved with a given process.
  - The above queries with respect to a given past time.

4 Simulation

In this section, we will describe simulation functions in detail, which is one of the major applications of GADB.

4.1 Background

Business process simulation is an important key to solve problems in groupware design and deployment.

In the first author's experiences in developing and deploying office groupware systems based on asynchronous communication, like workflow systems and group calendar systems, he encountered difficulties as follows.

1. Difficulties in estimating its impact
   It is difficult for a groupware designer to make a quantitative estimation of the cost-effectiveness prior to the introduction of groupware to the business field. However, managers of groupware users require persuasive explanation of its impact before making up his or her mind to buy the groupware. Hence groupware designers are expected to explain the impact, although it is not easy. It is because the effect of (especially asynchronous type of) groupware is mainly in reducing communication loss or inefficiency, not in the main job (like document production) itself.
2. Difficulties in practical evaluation and test
   Another approach to solving the problem of giving a quantitative measurement of groupware impact is to test the groupware in the real business field. However, because groupware involves many people, it is also difficult to conduct the experimental use. Moreover, in case of large-scale open groupware, small experiments do not totally give the estimation of large-scale usage. In case of special groupware systems like those used in the situation of disaster, testing at the real field is intrinsically impossible.
3. Difficulties in predicting human behavior
   As Grudin pointed out[6], it is difficult for groupware designers to imagine how user behaviors change between before and after the groupware introduction. Hence they cannot design groupware with taking account of its usage and impact.

One of the most successful examples in solving these problems can be found in the development of POLITeam[19]. One of the goals of POLITeam is development of telecooperation systems that support the cooperation among the distributed governmental functions between Bonn and Berlin. In the development of POLITeam, an evolutionary development cycle was taken[25]. Thanks to this participatory development cycle including many interviews and workshops with users, users accepted the designed groupware very smoothly. However, this kind of development cycle is very difficult to take in usual cases. In case of POLITeam, there was a very strong background, the movement of capital city, which caused a feeling of crisis and focused all people's intention, so that it was possible to get all people's cooperation for interviews and workshops. POLITeam's case seems to be very exceptional.

Simulation gives another approach to overcoming difficulties in groupware development and deployment given above. Its most important advantage is that it does not need users' cooperation. By simulating new business processes, designers and users can evaluate the impact of the groupware introduction or changing business processes.

On the contrary, a problem of simulation is its reliability. As the third difficulty given above shows, human behaviors are very difficult to predict. Behaviors also vary person to person. For example, even in case of very simple workflow processing, it is very difficult to estimate when a particular person at a step of workflow finishes a task. This is because we cannot predict when the person really starts the task and how the person uses his or her time for the task (intensively or not). Personal behavior is a very important factor of business process simulation and they make simulation difficult and unreliable.

In spite of this problem, business process simulation has been one of the hottest topics in the research field of simulation systems. In the next subsection, business process simulation system models will be reviewed.

4.2 Business Process Simulation

The aim of business process simulation is to give data of newly introduced business processes in order to estimate the performance of new processes and to predict the behavior of the processes. Given data will help the *business process reengineering (BPR)* process.

The aims of BPR are, for example, to increase service level, to reduce total process cycle time, to increase throughput, to reduce waiting time, to reduce activity cost, or to reduce inventory cost. To achieve these goals, some techniques like combining duplicate activities, eliminating multiple reviews and approvals, processing in parallel, outsourcing inefficient activities, eliminating movement of work, etc. are applied. New business processes are validated by simulating random factors of the environment like customer arrivals[23].

A basic model of business process simulation consists of four elements[24]:

Entities Also called as flow objects, tokens, or transactions. These are the objects processed by resources. Customers, products, and documents are examples.

Resources These are objects that are used for adding value to Entities. Service representatives, automated process equipment, and transportation equipment are examples. They are allocated to activities.

Activities They are linked by connectors in order to represent the flow of entities through the simulation model.

Connectors They are used for linking processes and activities.

4.3 Our Approach

In our approach, simulation is performed on the agent-based groupware model. The general simulation model and categories can be interpreted on this model as follows.

- A *resource* is a pair of human and agent (user-interfacing agent, or personal assistant).

- *Connectors* are messages exchanged among agents.
- Because agent-based groupware is open, multiple processes should be simulated simultaneously. By simulation, interrelationship among processes should be able to be evaluated.

The objective of simulating agent-based groupware is to estimate the result of business process reengineering a priori. It is not different from the case of the general business process simulation. More detailed objectives are as follows.

- To estimate the impact of introducing agents. For example, to compare two cases: a case where resources are only human and a case where some or all resources are pairs of human and agents.
- To estimate the impact of changes of rules on some agents.
- To estimate the impact of process changes (e.g. activity order, resource assignment, etc.)
- To estimate the robustness of process against changes of randomly occurring events like interrupting jobs or absence of a person.
- To estimate the inter-dependency of two or more processes. For example, when the number of cases of a process increases, what effect will occur on other processes?

Introduction of rule-based agents and consideration of multiple processes make the simulation model more complex than usual business process simulation. Among them the most important factor is the behavior of resources (i.e. people) when multiple tasks are ordered from multiple processes. To model this factor, the order of beginning tasks and time distribution to tasks by a person. Below are some extreme models.

- Processing all tasks by first come first service basis.
- Processing tasks by the order of deadline.
- Distributing processing time evenly to all tasks.

Again, they are very extreme patterns and real people do not behave like these. By constructing user models with data from GADB, models of each person will be refined. Establishing user models is very important and inevitable task for the simulation.

The most important factor of simulation is to estimate when an activity will be finished. There are two approaches to the estimation.

- The finishing time is estimated relative to the deadline. For example, on the day of deadline, on the day before the deadline, on the next day of the deadline, etc.
- The finishing time is estimated as the time after some standard processing period from the beginning of the task.

The first approach is suitable to project-based[24] processes, like document writing and specification development. In this approach, dealing with multiple processes is not an important factor as long as the workload is not heavy. Required data only appear at the (c) part of Fig. 2. Important data items are type of communication, task or business process name, time, and deadline.

The second approach is suitable to production-based[24] processes like order fulfillment, claims processing, etc. Resources are supposed to be always fully assigned to tasks. At the time after a constant period from a task delivery, the task is supposed to be finished. In this approach, the order of beginning tasks is an important factor for simulation. User models described above are utilized. However, we should say these models are too simple to get reliable simulation results. We must acquire a lot of data before simulation. For example, a successful case of business process simulation took a long time and much cost to acquire reliable values for simulation[15].

5 Evolution

In this section we describe how groupware can evolve with simulation, tactics and reengineering techniques, utilizing group activity data. The order of evolutionary cycle steps is: (1) tactics, (2) reengineering, and then (3) simulation.

5.1 Business Process Tactics

As described earlier, Business Process Tactics (BPT) refers to activities that change process operation to dynamically adapt the process to exceptional cases. To show how BPT is possible is out of the focus of this paper. From the viewpoint of GADB, BPT is a very important source of data. BPT log will be used for later reengineering.

5.2 Business Process Reengineering

There are some research results in the BPR (Business Process Reengineering) field. Some of them are on the reengineering model[3, 1, 7], some applied knowledge technologies [10]. In ActionWorkflow[16] and Regatta[21], reengineering is represented as incremental re-definitions of a process definition. Borghoff discussed how two kinds of business processes can co-exist, those processes before and after the reengineering[2].

In our research approach, there are some approaches to BPR.

- From the log data of BPT, dynamic changes that often occur are found. These give hints for BPR.
- Visualized data in GADB show some hints for BPR. For example, bottlenecks of process flow.

5.3 Business Process Simulation

After reengineering design, simulation should be exercised. Simulation may have several stages.

1. Initial simulation: Total process is simulated with user model data that have been obtained from old processes. The new process should be validated with various settings of simulation parameters.
2. Small introduction: Before totally introducing the new process, it should be partially tested, if possible. This field test can involve small number of key persons and agents used at the previous simulation stage. At this stage, user models of key persons are verified. If their user models are different from those at the previous stage, the models are modified and the simulation should be retried.
3. Beta testing: If the small introduction stage was successful, the new process is totally introduced to the real field. New data with the new process are captured by a GADB. These new data should be compared to the data taken at the simulation stage. If user and process behaviors are much different from those supposed at the simulation stage, simulation should be retried with new data. According to the new simulation results, another reengineering will be sometimes necessary.

6 Conclusion

This paper proposes the concept of Group Activity Database (GADB), which is aimed at supporting groupware evolution, i.e. reengineering and simulation of business processes. It can also support team awareness functions if the data are visualized to group members.

We are now implementing the simulation engine and framework of GADB. Current design is based on usual RDB, but temporal database would also be suitable. Detailed specification of required queries and analyses and visualization of captured data are next important research issues.

Acknowledghment

Part of this work is supported by the Japanese Ministry of Science and Education, Grant-in-Aid for Scientific Research on Priority Areas (A), "Advanced Database Systems for Integration of Media and User Environments."

References

1. Bogia, D. P. and Kaplan, S. M.: Flexibility and Control for Dynamic Workflows in the wOrlds Environment. Proc. of Conf. on Organizational Computing Systems '95, ACM (1995) 148–159
2. Borghoff, U.M., Bottoni, P., Mussio, P., and Pareschi, R.: Reflective Agents for Adaptive Workflows. Proc. of the 2nd Int. Conf. and Exhibition on Practical Application of Intelligent Agents and Multi-Agents (PAAM'97), London, UK, (1997) 405–420
3. Ellis, C., Keddara, K., and Rozenberg, G. : Dynamic Change Within Workflow Systems. Proc. of Conf. on Organizational Computing Systems '95, ACM (1995) 10–21
4. Fonseca, J.M., de Oliveira, E., and Steiger-Garção, A.: MACIV, A DAI Based Resource Management System. Proc. of the 1st Int. Conf. and Exhibition on Practical Application of Intelligent Agents and Multi-Agents (PAAM'96), London, UK, (1996) 263–277
5. Geppert, A. and Tombros, D.: Logging and Post-Moretm Analysis of Workflow Executions based on Event Histories. Proc. of 3rd Int. Workshop on Rules in Database Systems (RIDS), Lecture Notes in Computer Science **1312**, (1997) 67–82
6. Grudin, J.: Groupware and Social Dynamics: Eight Challenges for Developers. Comm. ACM **37**(1) (1994) 92–105
7. Herrmann, T.: Workflow Management Systems: Ensuring Organizational Flexibility by Possibilities of Adaption and Negotiation. Proc. of Conf. Organizational Computing Systems '95, ACM (1995) 83–94
8. Holt, A.: Organized Activity and Its Support by Computer. Kluwer Academic (1997)
9. Ishiguro, Y, Tarumi, H., Asakura, T, Kusui, D, and Yoshifu, K : An Agent Architecture for Personal and Group Work Support. Proc. of Int. Conf. on Multi Agent Systems '96, (1996) 134–141
10. Jaeger, T. and Prakash, A.: Management and Utilization of Knowledge for the Automatic Improvement of Workflow Performance. Proc. of Conf. on Organizational Computing Systems '95, ACM (1995) 32–43
11. Jennings, N.R., et al.: Using Intelligent Agents to Manage Business Processes. Proc. of the 1st Int. Conf. and Exhibition on Practical Application of Intelligent Agents and Multi-Agents (PAAM'96) London, UK (1996) 345–360
12. Jennings, N.R., et al.: Agent-Based Process Management. Int. J. of Cooperative Information Systems **5**(2-3) (1996) 105–130
13. Konomi, S., Yokota, Y., Sakata, K. and Kambayashi, Y.: Cooperative View Mechanisms in Distributed Multiuser Hypermedia Environments. Proc. of 2nd IFCIS Int. Conf. on Cooperative Information Systems (CoopIS'97) (1997) 15–24
14. Matsumoto, Y. and Ajisaka, T. : A Data Model in the Software Project Database KyotoDB. Advances in Software Science and Technology bf 2, Japan Society for Software Science and Technology (1990) 103–121
15. McGuire, F., Using Simulation to Reduce Length of Stay in Emergency Departments. Proc. of the 1994 Winter Simulation Conference (1994) 861–867
16. Medina-Mora, R., Winograd, T., et al.: The Action Workflow Approach to Workflow Management Technology. The Information Society **9** (1993) 391–404
17. Pan, J.Y.C. and Tenenbaum, J.M.: An Intelligent Agent Framework for Enterprise Integration. Trans. on Systems, Man, and Cybernetics **21**(6) (1991) 1391–1408

18. Poltrock, S.E. and Engelbeck, G.: Requirements for a Virtual Collocation Environment. Proc. of ACM Int. Conf. on Supporting Group Work (Group'97) (1997) 61–70
19. Prinz, W. and Kolvenbach, S.: Support for Workflows in a Ministerial Environment. Proc. of Conf. on Computer Supported Cooperative Work (CSCW'96), ACM (1996) 199–208
20. Streitz, N.A., Geißer, J., Haake, J.M., and Hol, J.: DOLPHIN: Integrated Meeting Support across Local and Remote Desktop Environments and LiveBoards. Proc. of Conf. on Computer Supported Cooperative Work (CSCW'94), ACM (1994) 345–358
21. Swenson, K.D.: Visual Support for Reengineering Work Processes. Proc. of Conf. on Organizational Computing Systems, ACM (1993) 130–141
22. Tarumi, H., Kida, K., Ishiguro, Y., Yoshifu, K., and Asakura, T.: WorkWeb System — Multi-Workflow Management with a Multi-Agent System. Proc. of ACM Int. Conf. on Supporting Group Work (Group'97) (1997) 299-308
23. Tumay, K.: Business Process Simulation. Proc. of the 1995 Winter Simulation Conference (1995) 55–60
24. Tumay, K.: Business Process Simulation. Proc. of the 1996 Winter Simulation Conference (1996) 93–98
25. http://orgwis.gmd.de/projects/POLITeam/design/

This article was processed using the LaTeX macro package with LLNCS style