IDEA GROUP PUBLISHING

701 E. Chocolate Avenue, Suite 200, Hershey PA 17033, USA
Tel: 717/533-8845; Fax 717/533-8661; URL-http://www.idea-group.com

ITP4804

Knowledge Sharing Infrastructure for Virtual Enterprises

Farhad Daneshgar, Pradeep Ray and Fethi Rabhi
School of Information Systems, Technology and Management,
The University of New South Wales,
Sydney, Australia
(f.daneshgar, p.ray,f.rabhi)@unsw.edu.au

ABSTRACT

A knowledge-sharing infrastructure is required to facilitate collaborative work in virtual enterprises. In doing so, it is important to have a methodology for mapping business level requirements to collaborative software tools. This paper introduces a methodology for analysis and design of collaborative information systems with the objective of maintaining a shared awareness among knowledge workers in virtual enterprises. A LotusNOTES implementation prototype of this methodology is also shown and its functions are highlighted using a Network Management/Trouble-shooting Case Study.

1. INTRODUCTION

Virtual enterprises enable groups of people from remote locations and at any time to collaborate over computer communication networks to perform business tasks. In today's information age economy, it is important to share knowledge and information within and across organizations. Such knowledge sharing is a common need for virtual enterprises in all business sectors covering finance, healthcare, telecommunications, retailing, aviation etc. Virtual enterprises are now being supported through intranets, extranets, and Computer Supported Cooperative Work (CSCW) techniques, such as groupware, workflows, and object-oriented software [Molli 2001].

Despite the productive history of research in groupware design, the majority of current groupware implementations are based on a bottom-up approach. One main reason could be a lack of high-level analytical models and frameworks for designing collaborative applications. This paper presents a top-down methodology for design of collaborative systems from a knowledge-sharing perspective. This methodology uses a conceptual model called the *awareness net*. This model helps in identifying the gaps in the collaborative process support. The methodology also allows designs to be systematically derived from the awareness net and later on, be implemented using a groupware system.

2. AWARENESS NET

2.1 Related Work and Motivation

The awareness net is a model for collaborative business processes. The main motivation behind the awareness net is to construct an analytical tool for modeling collaborative processes in a way that knowledge-sharing requirements of the actors within the processes can be easily identified and measured. This will then pave the road for introducing design and implementation directives for groupware systems that support such knowledge sharing and collaboration.

Traditional process-flow models such as Data Flow and Workflow models prove to be inadequate in addressing the awareness requirements of the users in collaborative business processes. Awareness modeling emerged from the area of Computer Supported Cooperative Work (CSCW) and has been an active research topic in the last ten years beginning with milestone research by researchers such as [Dourish 1992] and [Benford 1994]. This paper's major contribution is to establish links between the traditional process modeling in one hand, and the awareness modeling on the other hand, in order to guide design of groupware systems. An Object-based process model is used to model collaborative business processes by which awareness requirements can easily be derived from, or demonstrated by. This study also continues previous works in awareness modeling within the fields of Computer-Supported Cooperative Work (CSCW) and Knowledge Management.

2.2 Introducing the Awareness Net

Figure 1 shows the awareness net for a Network Management/ Trouble-ticketing collaborative business process. In this particular case, the role User reports a network problem to the role Operator. The Operator assigns a 'trouble ticket' (or T-Ticket) to the problem and then sends it to the role Technician for technical considerations. The Technician fixes the problem and arranges a meeting with the Test Coordinator for further tests. The Technician may then place a Change Request to the Change Manager requesting for some changes to be made in certain aspects of the network in order to apply the solution proposed by the Technician. Change Manager may then discuss the issue with the Users and assess the impact of the proposed change.

Within the awareness net each *role* object is associated with one or more 'task' objects. Association between a role object and each of its task objects is established via another object called the *role artifact*. A *role artifact* object carries information that is required by the *role* to satisfy the *need-to-know-about* requirement for that particular task. Two *task* objects can also be associated via another object called a *task artifact* object. A *task artifact* object carries information that is required

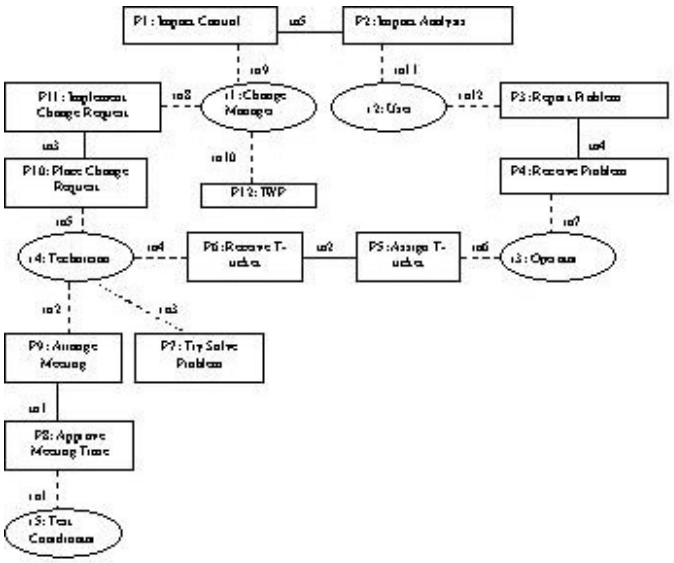

Figure 1: An Awareness Net for a network management

by both of the collaborating roles to satisfy the *need-to-know-about* requirement for a particular pair of 'task' objects.

In addition to the above concepts, the following two semantic concepts need to be clarified at this stage:

- ***Required Level of Awareness:*** this is the level required by the *task*. This is a level below which the *task* cannot be successfully implemented. This field is a property of the *task* object.
- ***Actual Level of Awareness:*** This is the level that the *role* object actually possess. This is a property of the *role* object. If the *actual level of awareness* of a role is lower than the *required level of awareness* of the task, then the role is not capable of performing the task successfully.

3. DESIGN ISSUES

3.1 Introduction

Given an *awareness net*, the next step in enhancing collaboration involves grouping of its objects into components that correspond to chunks of related to knowledge structures called *awareness spaces*. These awareness spaces can then be mapped onto a variety of implementation concepts such as database records or objects in an object-oriented language.

3.2 The Awareness levels

This section introduces measures or 'levels' that represent the potential degree of intensity of collaboration as a by-product of having satisfied the roles awareness requirements within the process. These levels represent different capabilities for the roles to perceive objects available within the various awareness spaces for knowledge sharing [Daneshgar 2001]. These levels are described below:

Level-0 Awarenes (Failed Level): Is awareness about the objects that lead an actor to knowledge about all *tasks* that the role performs within the process. For example, in Figure 1 level-0 awareness for the role 'User' is a structure with the following set of objects:

{Ra11,P2,ra12,P3}

The above structure can now be regarded as a context for knowledge sharing capabilities of the role 'User' within the process. Due to its relatively small size, this awareness space cannot enable the 'User' to share knowledge with others within the process. It does not include knowledge about other roles, their tasks, and their artifacts.

Level-1 Awareness (Direct Cooperation and Communication): This is the role's level-0, plus awareness about the objects that lead the role to knowledge about other related roles within the process. For example, in Figure 1 the role 'User' has task dependencies with the roles 'Change Manager', and 'Operator'. Level-1 awareness space for the role 'User' will be the space specified by the following mathematical set:

{{level-0-space},ta5,P1,ra9,r1,ta4,P4,ra7,r3}

The above awareness space provides a context for the role 'User' to initiate knowledge sharing transactions with his related roles, 'r1' and 'r3'.

Level-2 Awareness (Extended Cooperation): A role's level-2 awareness is his/her level-1, plus awareness about all other (or, the rest of) *roles* within the process. There may be multiple alternatives for this space. One alternative for the role 'User' is:

{{level-1-space},ra8,P11,ta3,P10,ra5,r4,ra2,P9,ta1,P8,ra1,r5}

Level-3 Awareness (Extended Communication): A role's level-3 awareness is his/her level-2, plus awareness about all the interactions (represented by the *task artifacts*) that occur between any two roles within the process. From the previous level, 'User' already is aware of a limited number of interactions that s/he has with 'r1' and 'r3', as well as between the roles that are not necessarily directly related to the 'User' but constitute part of the context for level-2; that is the context through which 'User' will be able to be aware of them. These interactions include User's interactions with Change Manage ('ta5'), and with Operator ('ta4'), as well as interactions between the Change Manager and the Technician (that is, 'ta3'), and between the Technician and the Test Coordinator (that is, 'ta1'). The only remaining interaction within the process that the User is not yet aware of is the one between the Technician and the Operator (that is, 'ta2'). Following are two alternatives for the awareness space for the User:

Alternative 1:{{level-2-space},r4,ra4,P6,***ta2***}
Alternative 2:{{level-2-space},r3,ra6,P5,***ta2***}

Level-4 knowledge sharing (Coordination): This is the highest level of process awareness within the scope of the process. It is awareness about all the objects within the process. In other words, this level will bring all remaining objects on the awareness net within the focus of the actor. For the User the remaining objects on the awareness net which have not yet been put within his/her focus is the following Set:

{{ra10,P12},{ra3,P7}}

If the above portion is added to the User's level-3 awareness the result will constitute the User's level-4 awareness. It is in fact the entire awareness net.

3.3 First-Level Normalization of the Awareness Net Structures

The next step in designing the proposed IT infrastructures is to store the awareness net in a knowledge base. It is proposed that all the structures that represent the awareness net be normalized in a way that they all start and end with a *role* object. The reason for this is that such structure incorporates all the contextual information that any pair of collaborating roles would need in order to collaborate with one another. Obviously, an exception is where a *task* is not related to any other *task* (and therefore, to any other roles) e.g. P7 and P12. In these cases these structures remain unchanged. In other words, the normalized structures for the awareness net of Figure 1 will simply be all the paths between every pair of roles:

(1) {r1,ra9,P1,ta5,P2,ra11,r2}
(2) {r2,ra12,P3,ta4,P4,ra7,r3}
(3) {r3,ra6,P5,ta2,P6,ra4,r4}
(4) {r4,ra2,P9,ta1,P8,ra1,r5}
(5) {r4,ra5,P10,ta3,P11,ra8,r1}
(6) {r1,ra10,P12}
(7) {r4,ra3, P7}

3.4 Second-Level Normalization of the Awareness Net Structures

As mentioned before, the *required level of awareness* is a single-value property of the *task* object and is represented by an integer ranging from 0 to 4 and is a parameter to the model. As a result, the analyst needs to assign one such attribute to each *task* object. Therefore, a further grouping of the above set of objects is required in order to prevent possibility of assigning more than one such attribute for each task [Daneshgar 1999]. Therefore the above seven structures will have to be expanded to the set shown in Table 1. This is so because each of the structures 1 to 5 above consists of two *tasks*, and each of these *tasks* is to be associated with its own *required level of awareness*, hence possibility of having two for the *required awareness level* in a single structure. This is not desirable, as it will create different kinds of anomalies. Also, notice that as a result of splitting these five structures into ten new structures, the last two fields of each of these newly established ten structures are deleted in order to remove redundancies within the new structures. Table 1 shows a list of 12 revised structures each representing an object package in a collaborative environment. These structures can be used to construct desired software for maintaining cooperation, coordination and communication among actors in collaborative processes.

Table 1: List of normalized *structures for the proposed system.*

Structure ID	Objects within Structure	Required Level of Awareness for the Primary Key
(1)	r1,ra9,**P1**,ta5,*P2*	4
(2)	r2,ra12,**P3**,ta4,*P4*	2
(3)	r3,ra6,**P5**,ta2,*P6*	4
(4)	r4,ra2,**P9**,ta1,*P8*	1
(5)	r4,ra5,**P10**,ta3,*P11*	3
(6)	r1,ra10,**P12**	0
(7)	r4,ra3,**P7**	1
(8)	r2,ra11,**P2**,ta5,*P1*	2
(9)	r3,ra7,**P4**,ta4,*P3*	4
(10)	r4,ra4,**P6**,ta2,*P5*	4
(11)	r5,ra1,**P8**,ta1,*P9*	1
(12)	r1,ra8,**P11**,ta3,*P10*	4

Bold: primary key

Italic: foreign key

3.5 Summary of Design Methodology for a Knowledge Sharing Collaborative System

This methodology consists of the following steps:

- **STEP 1**: *Construct the awareness net and derive all awareness structures*. The net can now be browsed for (i) objects that constitute each role's actual level of awareness, (ii) objects that constitute each tasks required level of awareness, and (iii) objects that constitute the excess of the *required level* over the *actual level* for each role performing a task, that is, the *awareness gap*.
- **STEP 2**: Normalize, at two levels, the structures derived in STEP 1 above using the method described in Section 3.3. Identify a set of structures similar to the Section 3.5.
- **STEP 3**: The Knowledge Sharing Collaborative System can now be constructed preferably using an object-oriented development technology.

3.6 Architectural implications

The system design approach in this paper corresponds to the multi-tiered collaborative view of system design/architecture layers described by [Brown 2002]. In our case, it proposes a new finer tier called *awareness tier*/*layer* to the existing two popular tiers: *domain* and *service* tiers. The *awareness object layer* contains the components derived from the awareness net. These awareness objects are specialized types of domain objects that are responsible for fulfilling the knowledge sharing requirements as specified in the collaborative business process framework.

4. CONCLUSION AND FUTURE WORK

This paper has presented a methodology for analysis and design of collaborative systems with the objective of maintaining awareness and knowledge-sharing among knowledge workers in virtual enterprises. The paper has described how concepts from the fields of Computer Supported Cooperative Work (CSCW) and Knowledge Management can be combined to effectively enable development of knowledge sharing tools that enhance collaboration in virtual enterprise processes. Such enhancements are achieved by identifying awareness gaps that may exist for these actors, and subsequently filling up these gaps. Finally, this paper has demonstrated an application of the methodology to a network management case study. More works are in progress to validate this methodology in other business sectors such as healthcare (e.g., [Weerakkody 2002]), real estate and finance industries.

Work is in progress in the areas of organizational culture and its effects on the required levels of awareness. Work is also in progress to improve the functionality of the development environment as well as using a workflow generator tool that provides both the workflow process definitions, as well as the functions for controlling the flow; that is, who must be aware of what objects, when, and for what purpose/task.

REFERENCES

[Benford 1994] Benford S., Bowers J., Fahlen L., Marian J., and Rodden T., "Supporting Cooperative Work in Virtual Environments", *The Computer Journal* 37(8) Oxford University Press, UK 1994, pp. 653-668

[Brown 2002] David William Brown, *An Introduction to Object-Oriented Analysis*, John Wiley & Sons, Brisbane, Australia.

[Daneshgar 1999] Daneshgar Farhad, "A Methodology for Planning, Analysis, Design and Implementation of Collaborative Databases: Introducing AWT Software", *2nd International Symposium on Cooperative Database Systems for Advanced Applications*, Wollongong, Australia.

[Daneshgar 2001] Daneshgar Farhad, "Maintaining Collaborative Process Awareness as a Mechanism for Knowledge Sharing", *2nd European Conference on Knowledge Management*, Bled, Slovenia,

[Dourish 1992] Dourish Paul and Victoria Bellotti, "Awareness and Coordination in Shared Workspaces", Rank Xerox EuroPARC, CSCW'92 Proceedings, 1992.

[Molli 2001] P. Molli, H. Skaf, C. Godart, P.Ray, R. Shankaran, and V. Varadharajan, "Integrating Network Services for Virtual Teams", *IEEE International Conference on Enterprise Information Systems (ICEIS01)*, Setubal, Portugal, July 2001

[Weerakkody 2002], Gamini Weerakkody and Pradeep Ray, *CSCW Based System Design for Clinical Practice Environments for Improving Quality of Service*, 4th International Workshop on Enterprise Networking and Computing (Healthcom2002), Nancy France, June 2002.