# SUPPORTING TEAM AWARENESS IN A COLLABORATION

KUK HWAN MIEUSSET
*Architecture and CAAD*
*HIL D 74.3*
*CH-8093 Zürich-Hönggerberg*

**Abstract**. People expect to profit from more efficient and enhanced collaboration using computer and internet technology. Unfortunately, certain behavioral dimensions existing in the face-to-face work environment, such as team spirit, gestures, atmosphere and artifacts, can be greatly compromised. In this paper, we propose to use a database framework to facilitate the realization of groupware to support team awareness. We will take a look at the development of our framework through two case studies and a discussion of our findings.

## 1. Introduction

The widespread use of the internet is radically changing the classroom, where more and more is being communicated with the help of web-based technology. Even though students are not necessarily sitting in the same room at the same time, they may still be curious to know who is attending the same lecture, what sort of work other colleagues in the course are doing and who is working at any given time. In addition, after the finished work is published in the internet they may be interested to know what other people think about it. Teachers can also use the internet to organize and oversee a class, as well as receive feedback from the students. This information could prove invaluable for simultaneously enhancing coursework as well as improving it for future courses.

However, existing systems designed to facilitate these processes are not very convincing. When we enter a lecture site, we often see a lot of information on it. But until you understand what really happened, who did what and why, you may waste a lot of time. Different work habits, schedules and a wider variety of settings make tracking information in the workplace difficult. As Kurmann pointed out, data organization is one of the major problems with existing interfaces. (Kurmann1999) Specifically, data is not defined in a way that allows it to be easily retrieved by the user. In effect, each operation must be programmed separately at the highest application level. This leads to further

problems for the designer of a virtual classroom, where data, course concept and relevant information are always changing.

To improve existing groupware technology, we analyzed the information which is produced in a teamwork environment through the development of two systems. One escorts people through the decision-making process of a multi-disciplinary team of virtual companies. The second system was intended to enhance a collaborative classroom environment for architecture learning how to program in 2 to 3 people teams. In light of these studies, we defined a framework which could ease the modeling and handling of information in a database using data manipulating language.

We will first describe the background of our methods, followed by the implementation and findings of two different systems. Our main findings will be discussed while presenting our framework - an awareness database. In conclusion, we will discuss how our experiences help define our future work strategies.

## 2. Motivation and Background

While developing groupware for architecture offices, we began by asking ourselves why the database is used solely as a depository of data, as tables or object sets where information is saved in a certain form and retrieved through predefined query language. We think the reason for this misunderstanding is because the database is regarded as being static

What if a database could work out and send you certain information before you asked about it, so that you only had to prepare to receive it? If a database could inform you what happened in the last two days before you sent a specific query? Or in other words, actively participate in the collaborative process? We then began to think of ways we could bring a database closer to the user.

We used many different sources in our research, including awareness theory, groupware systems designed for awareness, and database developments.

### 2.1. AWARENESS AND CSCW

Human factors researchers define awareness as "knowing what is going on" or an "understanding of the activities of others to provide a context for your own activities". Endsley (1995) describes team awareness with specific character coordination and knowledge-sharing among team members. CSCW researchers have taken this concept and tried to categorize different types of awareness. Stary (1993) defined approaches into five types: Conversational-based, Activity/role, Information sharing, Decision support, and Goal-based approaches.

Tollmar et al. (1996) developed a system to help one be aware of another’s approachability; Key functions include tracking colleagues to see when and

how they can be reached, if they are busy or can be disturbed, etc. Portholes (Lee et al, 1997) provide video-based background awareness and have additional capabilities such as giving feedback. Timewarp is a toolkit (Edwards et al, 1997) which shows the state of a shared document at any point in time as well as histories of changes with timelines. Other design approaches use abstract representation of data (Pedersen and Sokolar, 1997) or consider mixed-focus collaboration to support both individual and group work (Gutwin and Greenberg, 1998). In this research area, data related to more personal issues including emotion, personal habit, and morals is gaining more and more interest from researchers.

Groupware, designed to facilitate Team Awareness handles data that is dynamic and continuously changing. In the ideal case, the user must do as little as possible to be informed of a certain activity or situation. Here we see the complex requirements of a database which acts actively and collaboratively.

### 2.2. A SMARTER DATABASE

Fuchs (1998) declared the database as one of the foundations of the CSCW system after the distributed system and the internet. Tarumi et al. (1998) proposed the Group Activity Database (GADB), designed for agent-based groupware. The GADB supports the evolution and deployment of groupware systems and is able to track log data related to process, task, and workers within a time period or at a given time for reengineering.

Even though there have been many efforts in the CSCW area to make users conscious of the group work environment, the potential use of the database in this context has received very little attention. Paradoxically, the main goal of CSCW researchers is to try to keep users abreast of data, in other words, help them be aware of the events occurring inside of a database.

We believe a database framework could improve the visualization of data in the user interface by modeling and keeping track of awareness data objects instead of having to gather all relevant pieces each time it is called for. As a result, communication between database and applications for the user interface can be significantly reduced.

## 3. Case Studies

A prototype for a framework was tested during the Swiss national foundation project *A Tool Set for The Virtual AEC Company* and in our yearly Programming Course. Both systems were internet-based groupware and implemented using database Oracle, JDBC, Java in the backend system, and Java and JavaScript for user interface data handling.

### 3.1. DECISION SUPPORT SYSTEM

Our Virtual AEC Company project (Stouffs et al, 1998) aimed at providing a manageable platform for information, communication, and collaboration between all partners in the building planning, design, construction, and management processes. For this project, we used a function-related subsystem called "Decision".

The user of this subsystem can be a member of one or more groups and can be involved in several different projects at once. We visualized the process of decision-making not only in a group but also over the course of entire projects.

To define a decision process, we can use the second icon of the menu on top of the image to define issues, choose members, and set deadlines. The state of a process can be seen from a global view as well as a personal view, where overall progress is shown by the height of the bars and the state of the personal decision is shown by their color. (Figure 1-2)

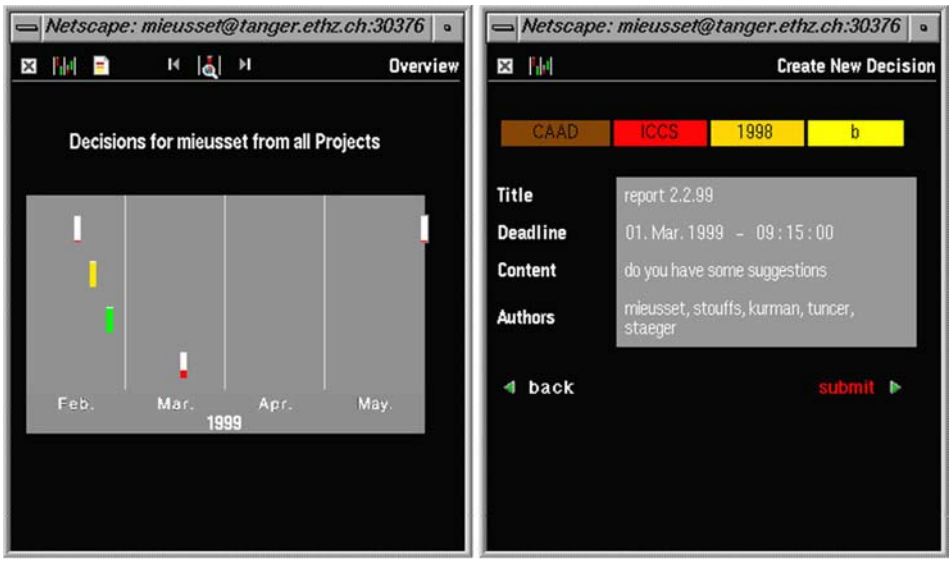


*Figure 1.* Interface of the Decision Agent-Overview and Creation Result

Every action which changes the interface content is held at the interface application side at the same moment it enters the database. The disadvantage of this approach is the effort required to enter the main system, an inconvenience that is partially offset by the subsystems' ease of use.

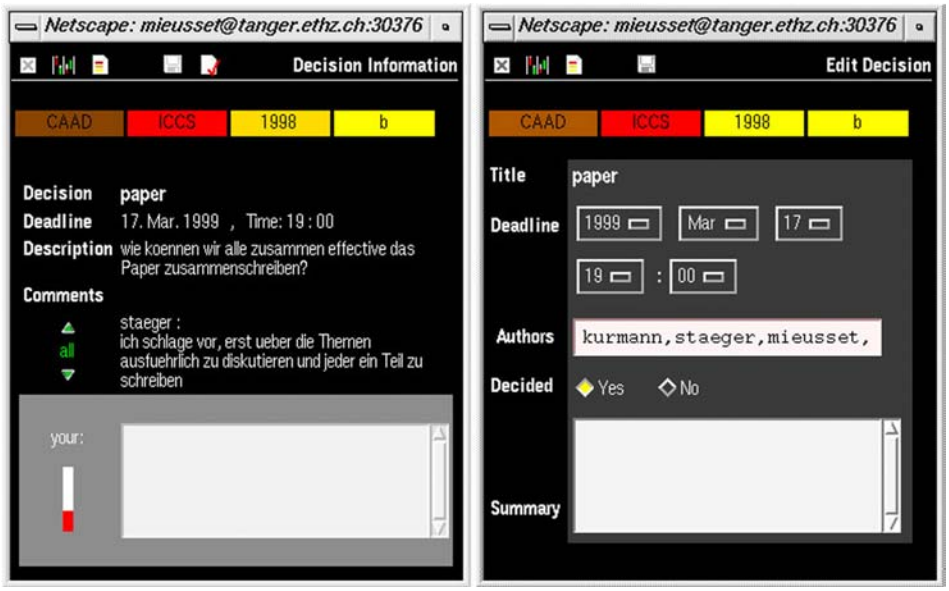


*Figure 2.* Details of a Decision-making Process

## 3.2. PROGRAMMING COURSE ENVIRONMENT

This course, designed for architecture students who are experienced in using CAAD tools but are new to programming, was characterized by the limited life-span of teams and a continually changing variety of goals and tasks.

To support team awareness, we created frames to show up-to-date work status, students' queries, program scripts, course schedules, and a list of student teams and teachers with their pictures. We created a little window to show synchronous activity information which can stay on the screen after the main window is minimized. (Figure 3-4) When the database was updated, the action was immediately recorded in this small window. This window was particularly useful for teachers; they could leave this window open overnight and automatically see an overview of activities in the course the next morning.

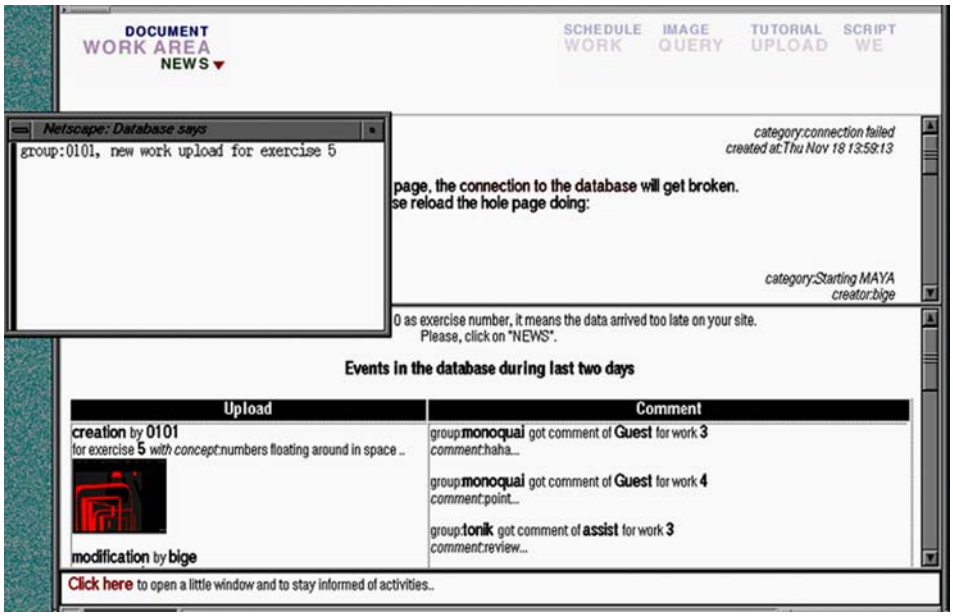


*Figure 3.* Programming Course Interface

On the page where the students' work is automatically displayed after being uploaded, we gave the possibility to download and also to comment on other work. (Figure 4) This could also be done by people from outside the course. Here, the most frequent comment was "I would like to see HOW you did it". The students were learning by looking, copying and comparing their own idea to another's. Because process was visible, the authors were able to keep track of who and downloaded their work and how often.

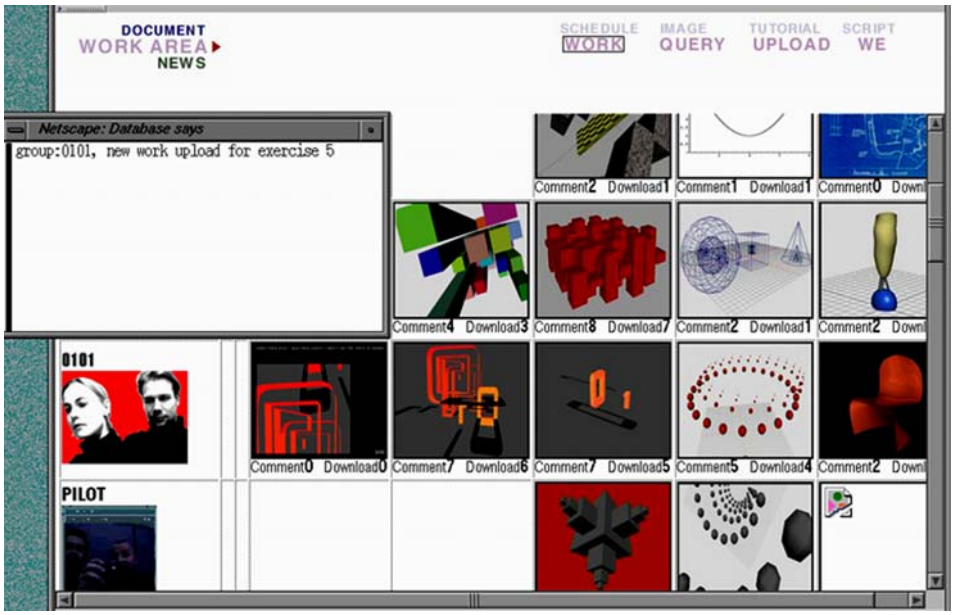


*Figure 4.* Work Overview Interface

### 3.3. LESSONS LEARNED

A student survey showed that students thought that the system should be readily accessible and the interface should be user-adaptable. Students in a class expect different system interfaces and functions depending on their own taste, needs, habits, etc. Therefore it is important that the database can model the user and accommodate more personalized information.

The inconsistency between the organization of objects in a database and the user interface seems problematic because translation work can quickly overload the middleware. Therefore a representative way to make the information compatible for both the user interface as well as the database is needed. Defining actions performed on an object as data types could reduce the distance between database and interface.

## 4. Framework - Awareness Database

The goal for many people designing and programming a course environment is to offer a better and more attractive one than the one offered during the previous course. As a result, expectations of both students and teachers correspondingly increase and may need to be significantly changed. To simplify this time-consuming process we propose a new framework, or "awareness database", which contains the following components:

1) query/modeling language
2) user request/database response interface
3) consistency check
4) awareness modeling
5) linkage to basic schema

The first component of an Awareness Database is a specific language models an object or process related to an awareness requirement. For example, if a new person joins a group, all members of the group need to be informed, which can be defined by be_aware ([person], [create], [inform, all person of group A]). The second component parses a designer's queries to other levels and handles the communication between the database and several users. Notation has to be checked in this level before the command is forwarded to other components of the framework, and when the update has to be sent to many different users. A consistency check verifies the compatibility of queries or definitions sent by the user and fundamental data objects and objects from the database. Awareness modeling defines objects with customizable methods. In this part of the meta model, each piece of information can be created as an instance of a collection of objects and the methods can from now on be processed on it. Each designer presents different data for different interface, which requires setting up correspondingly different databases. In the linkage to the basic schema, they can be attached to the meta model from each database schema.

Using an Awareness Framework the system developer can more easily manipulate the information and activities on it as data type and save the relevant operations by adapting the interface to each new case.

## 5. Conclusion and Future Directions

A framework-awareness database has the potential to model and handle data in a way that enhances team awareness. In this paper we considered the problem of how to better support collaborative work by better utilizing a database. In two case studies we analyzed different kinds of information, methods of representation and how they are used. We found out that proper representation awareness information can save much unnecessary effort on the part of the designer.

Further research will involve realizing this framework with different database environments, starting with the object-oriented database management system OMS of the Global Information Systems research group of the ETH Zürich. This object-oriented system has much potential to give the freedom to realize our framework and procedural method writing possibilities. We will also continue to develop easier methods of integrating different user models for our framework.

## Acknowledgments

Thanks to Arley Kim for her English assistance.

## References

Edwards W.K. and Mynatt E.D.: 1997, Timewarp: Technique for Autonomous Collaboration, CHI 97, Atlanta GA USA, pp. 219-225.

Endsley M.R.: 1995, Toward a Theory of Situation Awareness in Dynamic Systems, *Human Factors,* 37(1), pp.65-84.

Fuchs, L.: 1998, Situationsorientierte Unterstützung von Gruppenwahrnehmung in CSCW-Systemen, GMD Research Series, GMD-Forschungszentrum Informationstechnik GmbH, Nr. 3/1998.

Gutwin, C. Greenberg, S.: 1998, Design for Individuals, Design for Groups: Tradeoffs Between Power and Workspace Awareness, in Proceedings of the conference on Computer supported cooperative work (CSCW '98), Nov. 14-18, Seattle, WA, pp. 207-216.

Kurmann, David: 1999, InforMotion, in Proceedings of the Fourth Conference on Computer Aided Architectural Design Research in Asia, pp. 133-141.

Lee, A., Girgensohn, A. and Schlueter K.: 1997, NYNEX Portholes: Initial User Reactions and Redesign Implications, in GROUP'97, International Conference on Supporting Group Work, ACM, New York, pp. 385-394.

Pedersen E.R. and Sokolar, T.: AROMA: abstract representation of presence supporting mutual awareness, CHI 97, pp. 51-58.

Stouffs, R., Kurmann, D., Tuncer, B., Mieusset, K.H. and Staeger, B.: 1998, An information architecture for the virtual AEC company, in R. Amor (ed.) Product and Process Modelling in the Building Industry, pp. 479-486, Warford, UK.

Stary C.: 1993, Dynamic Modeling of Collaboration among Rational Agents: Redefining the research Agenda, Human Organizational, and Social Dimensions of Information Systems Development (A-24), D.Avison, J.E. Kendall and J.I. DeGross (Editors), Elsevier Science Publishers B.V. (North-Holland), pp. 219-239.

Tarumi, H., Matsuyama, T., and Kambayashi, Y.: 1998, Group Activity Database for Groupware Evolution, in Advances in Database Technologies, Lecture Notes in Computer Science, Vol. 1552, pp. 408-420, Springer-Verlag.

Tollmar, K., Sandor, O. and Schoemer, A.: 1996, Supporting Social Awareness @Work Design and Experience, in Computer Supported Cooperative Work '96, ACM, Cambridge MA USA, pp. 298-307.

URLs:
A Tool Set for The Virtual AEC Company: http://iccs.arch.ethz.ch/
Programming Course: http://prog.arch.ethz.ch/
OMS: http://www.globis.ethz.ch/