

# **POSITION PAPER FOR THE WORKSHOP SOFTWARE ARCHITECTURES FOR COOPERATIVE SYSTEMS**

Tom Gross  
Institut für Informatik  
Johannes Kepler Universität Linz  
Altenbergerstr. 69  
A-4040 Linz, Austria

Phone: (++43) 732 2468-9586  
FAX: (++43) 732 2468-9308  
E-mail: tom@ifs.uni-linz.ac.at

## **ABSTRACT**

This position paper describes my interest in this workshop as well as some topics which I think should be dealt with. Most topics are outlined through questions or problems which as I think offer no or no sufficient solutions.

## **MOTIVATION**

Prices of hardware have dropped drastically within the last years. As a consequence faster processors are available for less money. More and more operating systems and applications take advantage of these circumstances and support a number of users simultaneously. The problem is that most designers of operating systems and multi-user applications have considered the individual users in isolation. The users had no possibility to notice the presence and actions of other users working with the same operating system or application at the same time but at different locations. They generally only became aware of their co-workers when they had to share common resources like printers, modems etc. or wanted to access objects locked by other users.

## **AWARENESS**

Recently researchers as well as software designers realised these shortcomings and tried to provide the users with awareness information

about each other. However, in my opinion the optimal manner for providing the users with this information could not yet be found.

One of the most important reasons is that you can measure the nearness of physical things within our 3D world - when you sit in front of your workstation you e.g. can measure that the distance between your eyes and the screen is 40 cm -, but it is very difficult to measure the distance between two users working simultaneously within the virtual reality of a groupware application. The question is, if it is the right way to use measurements of the real world within virtual realities of computer applications? Does it really make sense to state that the distance between two users working cooperatively with a multi-user text editor is e.g. 40 lines within a multi-user text editor or four rectangles within a multi-user graphic editor?

If it is sensible, what can we draw from this kind of information? Is there really a difference for a user between a coworker who is 80 lines far away from him/her and another one who is 30 lines away. There could exist a strong interdependence between two chapters although there are many lines between them. The question that arises is if there is actually any useful measurement and which?

Another problem is that awareness is not quantifiable. I think it is not possible to quantify awareness neither in reality nor in CSCW applications. Can you really say that e.g. you are aware of your secretary of  $x$  percent if you know the colour of her dress and  $x-y$  percent if you do not. How can you explain why a user working with a document of the folder which contains the document another user is editing has an awareness of the other user of  $z$  percent more than of a user working with a document of a different folder. Are there not several criteria to organise file hierarchies? Is the criterion selected for the organisation of the files really always the right criteria to calculate awareness? The problem is that possibly for different cases you could need different kinds of awareness measurements. If you use awareness to mainly coordinate the cooperative editing of documents you probably will need other calculus of awareness than if you want to coordinate and optimise the usage of hardware appliances. And you could need totally different awareness calculation schemes if you have to provide awareness of hackers in your system.

There are many questions as far as user awareness is concerned. No matter if awareness of other users at the level of operating systems, i.e. of users sharing the same objects like folders and documents etc. is concerned or if awareness of other users working within the same application and/or the same document, we have to find some (new) criteria to evaluate the distances between the users. Especially users collaborating at different locations simultaneously have to rely on this information because in most cases they have no other possibility to realise the presence and activities of other users.

Another very interesting and important topic in the field of architectures for cooperative systems are access control models.

## **ACCESS CONTROL MODELS**

There are different access control models emerging. Most of them have been derived from the classic access matrix. Generally access rights are defined as s,o,a triples, where s means subject, o means object and a specifies the access right. The matrix can be split into rows where each user has a list of accessible objects or it can be split into lines where each object has a list of users who are allowed to access the respective object.

Access models for CSCW systems have special needs. They have to provide multiple and dynamic user roles. Roles are dynamic if access rights can be inferred from the roles and the situations. The question is, if it is possible to forecast all combinations of roles and situations? What happens if there is a combination of roles and situations which has not been considered? How many and which parameters do we need to unequivocally define a certain situation? Given that multiple roles are necessary and permitted, how can role conflicts be handled? Should a person with several roles get the minimal or maximal access rights or how could a sensible average be calculated?

There is a need for flexibility and fine-grained subjects, objects, and access rights. It should be possible to specify each access right for each subject on each object independently. Which granularity do we need? Is the finest granularity always the best solution? Could administrative overheads be

reduced through a coarser specification? How and with which parameters can the optimal granularity be determined?

Finally there are requests for automation. The implementation of access rights should be easy. Especially in CSCW architectures many users are involved and a huge number of objects has to be administered, but is it not dangerous as far as security is concerned to have access rights specified automatically? How can access rights that have been specified automatically be checked for correctness? Does inheritance of rights not mean to run the risk of propagation of errors and manipulations without sufficient control?

There are several access control models suggested. Most of them content very good approaches, but the demand for faultless access right models and the danger of misuse of information etc. through shortcomings of the models and incorrect specifications are especially present in CSCW systems where many users cooperate or collaborate.

## **CONCLUSION**

Though the formulation of good questions requires some knowledge of a certain field, good answers are harder to find. I hope that we will find some answers in this workshop.