SOME ISSUES OF ROLE-BASED COLLABORATION

Haibin Zhu
Member, IEEE, Dept. of Computer Science, Nipissing University, 100 College Dr.,
North Bay, ON P1B 8L7
haibinz@nipissingu.ca

Abstract

Roles are a powerful concept for facilitating distributed systems management. It has been accepted as a good solution for information sharing based on the research of RBAC (Role-Based Access Control).

Although no explicit roles are declared in traditional CSCW (Computer-Supported-Cooperative Work) systems, there are implicit roles granted in the design. Consequently, we may infer that "without roles, there would be no collaboration".

There is, however, a lack of specific, comprehensive research on building collaborative systems with role management and on role-based methods that have been successfully used in access control.

This paper aims to clarify the basic problems related to the research of role-based collaboration and to propose a basic method to do that research.

Keywords: Role, Role-Based, Collaboration

1. Introduction

Collaborative systems combine communication, computer, and decision-support technologies to support problem formulation and solution in group meetings. Collaboration on computer systems is totally different with that in face-to-face environments. Even though scientists are working hard to provide systems that provide virtual environments that make collaboration easy, and the collaborators work hard to be cooperative, there are still many problems for collaborators to know and overcome when using a collaborative system.

Roles are a powerful and policy neutral concept for facilitating distributed systems management and enforcing access control [9]. Research in the area of role-based access control has made rapid progress over the past few years [12].

Beside the communication platforms, research on collaborative systems is mainly concerned with two other categories: information sharing and coordination [11]. Currently, there are still problems unsolved in these two areas, such as, flexibility of a workflow system, protection, security and efficiency in information sharing, and conflict presentation and resolution.

Collaboration needs to share not only information but also view [13]. If we can support sharing with reasonable time and cost factors, we can provide acceptable collaboration platforms. It has been accepted that role-based methods are one good solution to access control [4, 12].

In coordination, workflow management systems model the sequence of subtasks in a work process and the roles performed by each individual. It has been accepted that the designation of roles is a basic requirement in coordination [3, 5].

Even though there are no roles declared in traditional CSCW systems, there are implicit roles granted in the design and application of these systems. We can state that without roles, there would be no collaboration. Therefore, role-based collaboration can make contributions on both sharing and coordination.

The basic idea of role-based collaboration is that if users log into a collaborative system that can designate clearly what objects they can access with specific rights, and can also designate which users they can manage or communicate with, they can then accomplish their jobs meaningfully and efficiently. In this style, collaboration is done successfully.

In the second section, we discuss the basic concepts in role-based collaboration. After that, we clarify what basic issues we should consider with more attention. Finally, we conclude that role-based collaboration requires further research and will bring us more advances in building collaborative systems.
2. Basic Concepts in Role-based Collaboration

Role definition is a difficult job in managerial and behavioral sciences and many consultant companies offer services such as position or role descriptions to help enterprises to find a correct person to do a special job [2, 6, 7].

In RBAC, a role is defined as a semantic construct forming the basis of access control policy [12]. However, we still need to define and clarify the role concept in collaboration with our specific requirements.

In behavioral science, a role is defined as a prescribed pattern of behavior expected of a person in a given situation by virtue of the person's position in that situation [6]. In other words, a role is defined by the responsibilities that a human can or must take. Nevertheless, it is difficult to describe a role clearly and strictly because natural languages are by their very nature ambiguous. For this reason, different persons in identical positions may make different contributions.

In a collaborative system, we may specify responsibilities clearly to a person with some special technical support. The possibility is due to the exactness of computer languages and the limited computer resources a person can control at any time.

In a role-based system, there should be users, roles, objects, operations and permissions [4].

In collaboration, there are human factors and system factors. We call the persons who join the collaboration human users. We call the system the human users are using a collaborative system.

Even though a human user in collaboration cannot be changed physically, his/her role in collaboration may be changed. Therefore, in a collaborative system, we can define a role as a definite state of a human user.

Roles are generally defined by responsibilities. From an object-oriented viewpoint, we can use messages to accomplish these responsibilities. Therefore, a role can be expressed by an interface between a human user and the system. The interface is actually composed of a list of messages that objects in the system can receive and respond to. The interface will change based on the messages a human user has issued.

Shared objects will be changed by the messages issued by the human users, and produce finally a result of collaboration.

In an object-oriented view, a collaborative system itself can be considered as a complex object that is composed of other objects. Every object has a list of messages the human users or other objects can send to it. An object provides a different list of messages to different human users based on their roles. Therefore, a collaborative system provides different interfaces to a human user when he or she changes his or her role.

In summary, in a role-based collaborative system, there are concepts as follows.

- An object is used to express everything in a collaborative system [8].
- A human user is a person who is participating in collaboration.
- A message is a command that may be issued by a human user or other objects.
- An interface is a list of messages sent to objects in the system or to the system itself.
- A role is a special object that symbolizes a logged human user in the system, and a role must have an interface.
- We accept classes as templates of objects [3].

Fig. 1. The collaborative mode based on roles

With the introduction of roles, a collaborative system will support the collaboration mode depicted in Fig. 1, where each human user should log into the system with a role. With a role, he/she uses a specific interface to access objects in the system, to send messages to the system and transfer to other roles. With the interactions of roles, collaboration takes place and the outcomes are the objects in the system.

3. Basic Issues in Role-based Collaboration

In role-based collaboration, many issues need to be clarified.
1) Role assignment
2) Role transitions
3) System architecture with roles
4) Role-based coordination
5) Role-based interface design
6) Role-based conflict resolution
7) Role-based information sharing
3.1 Role Assignments

In collaboration, every human user performs a role. We need a mechanism to decide what roles to assign and how to assign roles to human users. We also need to assume many roles in a collaborative system. At least, we should provide two kinds of role assignments, default and newly-defined. Default roles are those predefined with definite responsibilities assigned by the system. Newly defined roles might be defined by some specific roles such as an administrator or a group leader.

Default roles are relatively easy to be applied in the design of a system. For default role assignment, we can use a two dimensional array such as \( A[0...m-1][0...n-1] \) to express different roles’ accessibilities to different objects. One dimension is used to express role identification, while the other dimension to express the operations commonly operated on objects. \( A[i][j] \) (i = 0, …, m-1, and j = 0, …, n-1) is a set of objects. Role i can perform operation j on all the objects in \( A[i][j] \). This role assignment is similar to what was used in the UNIX operating system. In fact, this method is only practical when the operations are definite and common to all objects, such as files, in an operating system.

However, in reality, a system is more useful if it can accept new roles and apply them while the system is running. From the viewpoint of object-orientation, objects can accept many different messages. The difficulties are among the application of newly defined roles when the system is executing and are outlined as follows.

- How to define a new role and its interface by requirements?
- How to express a new role’s accessing of the objects in the system?
- How to apply a new role in current collaboration?

The problem is how to define a role that can be applied in the running system. That is to say, do we have an abstract entity available to express a role? The answer is no for the time being, even though there is some implementation in a simplified format in security and access control field. Therefore, we need to develop new methods to express newly defined roles and apply them in current collaboration.

3.2 Role Transitions

Without role transitions, role assignment would be a trivial thing. However, in collaboration, not all the human users keep their roles unchanged. The situation is that there are many transitions among roles for one human user, and some transitions of one human user might affect other transitions of other human users.

Role transition will become a very complicated process when the number of roles and the number of human users in the system grows large. At the same time, role transitions may lead to conflicts among the access to shared objects.

For example, a human user might join a group and become a member. He/She might create a group and become an owner or a creator of a group. A group leader might assign another member a role of manager of a group. If there are human users who can issue the same messages at the same time, there must a mechanism to deal with the conflicts among the human users with interleaved roles.

3.3 System Architecture with Roles

A collaborative system can be built on different architectures. Based on the interfaces and the information management components of a system, we have a categorization such as totally centralized, totally distributed, or partially distributed [13]. Another categorization of architecture is using purely centralized, fully replicated, or semi-replicated [1]. With roles, we can re-organize software components on these different architectures to obtain new availability of collaborative systems.

We need to develop the basic structures that are applicable and efficient in facilitating roles and role-based collaboration. Fig. 1 shows a conceptual structure of a role-based collaborative system.

3.4 Role-Based Coordination

Coordination is another key activity of collaboration. Workflow management systems are the basic method of supporting coordination in that they model the sequence of subtasks in a work process, as well as the roles performed by each individual.

Based on role assignments and role transition mechanisms, we can provide strategies, regulations or algorithms for the system design. With these algorithms, the system can support collaboration effectively and efficiently. The coordination among human users is made easy by assigning roles.

For example, a workflow can be defined by a serious of role transitions, a group of role interactions or a sequence of synchronized messages issued by human users based on their roles.
3.5 Role-Based Interface Design

A multi-user interface is important for a collaborative system. It generally provides a virtual environment for human users to collaborate. In a role-based collaborative system, a user interface can support more than traditional CSCW systems.

Based on role assignments and role transition mechanisms, we should provide special interface design strategies and provide fundamental requirements for interfaces supporting role-based applications. By these strategies, the human users can get collaboration results easily and productively.

3.6 Role-Based Conflict Resolution

Conflict resolution is one of the most important problems we must overcome in building a collaborative system. Early in 1988, this problem was introduced and began to be studied among scientists. For more than a decade, scientists made many contributions. However, there is little research on conflict management using role-based methods, even though RBAC has been very successful.

Though role assignments and transitions, we can eliminate many possible conflicts among human users and make the collaborative activities more consistent.

Role-based conflict resolution will be an important topic in role-based collaboration research.

3.7 Role-Based Information Sharing

In collaborative concepts, if we can support sharing with reasonable time and cost factors, we can provide acceptable collaboration platforms. Role-based methods have been accepted as one good potential solution for information sharing based on the research of accessing control that has been carried out thus far [1, 4, 9, 10, 12]. We can find a new application in collaborative systems for role-based methods that have been very successful in access control.

4. Conclusion

The role-based method is a successful method in access control. It can also be used in computer-supported collaboration. After reviewing the state-of-art of CSCW, we discussed the basic concepts of role-based collaboration, and outlined the fundamental issues in the research of role-based collaboration. We can conclude that role-based collaboration will bring in new development for CSCW and require more comprehensive research.

Thanks to the research funding from Nipissing University (No.: 10-3287-42195).

References