Hierarchy and Competition in CSCW applications: Model and case study

J.I. Asensio, R. Calmeau, Y.A. Dimitriadis, E. García, M. Martín, I. Soto

Department of Signal Theory, Communications and Telematics Engineering
School of Telecommunications Engineering
University of Valladolid
Camino del Cementerio s/n
Campus Miguel Delibes
47011 Valladolid
Spain

juaase@tel.uva.es

ABSTRACT. CSCW applications need to adapt themselves to the functional and organizational structures of people that use them. However they do not usually support division in groups with a certain hierarchical structure among them. In this paper, we propose and study a theoretical model of groupware appliations that reflects those hierarchical interactions. The proposed model is also intended to evaluate the effects in performance derived from competitive and collaborative relationships among the components of a hierarchy of groups. In order to demonstrate the above ideas, a groupware game, called Alymod, was designed and implemented using a modified version of a well-known CSCW Toolkit, namely Groupkit. Groupkit was modified in order to support group interactions in the same CSCW application. In Alymod, participants compete or collaborate within a hierarchical structure to achieve a common goal (completing gaps in a text, finishing numerical series, resolving University course examinations, etc.).

RÉSUMÉ. Abstract translated into French.

KEYWORDS: Hierarchical system, cooperation-competition, groupware.

MOTS-CLÉS: The same rule applies as for the English version.

Introduction

CSCW (Computer Supported Cooperative Work) or groupware is a rapidly evolving field whose main goal is to assist groups in carrying out a common task through a shared interface or environment [GIB 91,MAR 92,BAE 93]. Although the advances of the supporting technologies are impressive, real world applications of CSCW have not reached the expected level of success. This is mainly due to the lack of adaptation of the CSCW applications to the internal structures of the organizations in which they are intended to be used [BUL 91,GRU 90,ROD 93 ,ORL 91,ORL 92a,ORL 92,TYR 94,DES 94,OKA 94].

A first important fact to take into account is that some kind of hierarchical relationship usually appears in the internal structure of organizations (not all the employees belong to the same group, with the same rights) although ``pure'' hierarchical structures may not always be the most suitable organization [CHA 91]. Nevertheless, hierarchical structures have been recognized to be very useful under concrete circumstances [CIB 88] characterized by a mid-level degree of task uncertainty and goal congruence.

It is also important to point out that, although typically subgroups cooperate directly or indirectly in order to achieve the global objectives of the, it is normal that they also compete in order to provide the best solution for a given problem (the ``market-like'' structure described in [CIB 88]). For example, the compiler group of a computer manufacturing company may suggest a better solution for increasing computer performance, instead of adopting a solution for higher chip integration proposed by the hardware group. Another example in a larger scale refers to the preparation of proposals for research projects by competing consortia.

Furthermore, groups are usually established dynamically, with varying members and/or roles [COL 92].

However, existing CSCW platforms, like Lotus Notes [ORL 92a], do not manage efficiently such hierarchical organizations with cooperating or competing dynamic groups. Some efforts were reported in Multigroup Decision-Support Systems, that assist synchronous activities of loosely coupled, nonhomogeneous small groups [PAL 94]. This can be partially explained by the difficulty in creating groupware rapidly, that will be flexible enough in order to get adapted to the structure of a specific organization. Since the overall complexity of creating groupware is very high, a lot of effort has been devoted in proposing toolkits that cover the basic common functions, such as access control, registration of users, concurrence or communication mechanisms [TOU 94]. One of the most widely used CSCW toolkits is Groupkit [ROS 92], a public domain, object oriented software package, that is available for different platforms.

The work presented in this paper deals with the development of a new software platform above standard Groupkit, that permits an efficient management of subgroups. By means of this feature, hierarchical relationships and competitive/cooperative interactions can be supported by the CSCW applications developed on top of this platform: grouping of individuals from different hierarchical levels in different subgroups enables hierarchical relationships, and restriction of information dissemination among different subgroups, gives support to different levels of competitive/cooperative interactions.

By means of this platform, organizations, software developers, or business administration researchers could create new applications adapted to their own hierarchical structure, or simply to study different organization alternatives by efficiently simulating different alternatives.

In order to study the possible impact of the new platform and obtain experimental results about different organization alternatives, we developed a new educational groupware game, called *Alymod*, with various groups that compete/cooperate using different strategies. Such an application and the associated experimental work could then validate our expectations about this type of CSCW systems.

CSCL (Computer Supported Cooperative Learning) is a particular field of application of all the above principles and, therefore, taking advantage of our current involvement in the CSCL world [BLA 96,GON 97,GON 97a] we have used *Alymod* in real educational environments by means of a set of examinations of two courses belonging to the telecommunications studies of the University of Valladolid, taught by two of the authors of this article.

In this paper, we initially explain why hierarchical systems have to be supported in groupware. After a brief explanation of the way Groupkit works, we present a new platform that permits it to support hierarchical systems. Then Alymod is described, an application that serves us to study this type of cooperation and competition in hierarchical systems. Later, the design of experiments follows together with experimental results. The final part of this paper is dedicated to an analysis of the experimental results, conclusions and directions for further research.

1. Hierarchical systems in groupware

1.1. The need for hierarchy and competition in CSCW

Hierarchical relationships among groups of people, and among people inside groups, has been recognized as the best group structure under certain circumstances [CIB 88]. Hierarchical arrangement of groups and people is a reasonable candidate able to maintain the cohesion and functionality of (mainly) large groups: distribution of tasks, assignment of decision responsibilities, etc. That hierarchical distribution can be considered, to some extent, as the ``constitution'' of the group or organization. Military hierarchy, typical enterprise hierarchies, etc., are real life examples of hierarchy among groups and hierarchy inside groups.

Hierarchical distribution of groups and individuals shows some advantages with respect to ``flat'' relationships among groups and among individuals:

- When a huge amount of people has to devote its efforts to a particular goal, the division of the whole set of people in variable-sized groups distributed in a hierarchical fashion gives a great flexibility and adaptability to changing situations: size of groups can be adapted to the difficulty of particular tasks, individuals can be assigned different responsibility levels according to their abilities, etc.
- Hierarchy inside groups helps in reducing the time inherently devoted to the achievement of agreements by means of ``democratic'' social protocols. This reduction is of a great importance in those situations in which the group or organization has to distribute its human resources in order to fulfill tasks with very strong time restrictions.

Hierarchy in implementations of CSCW applications had been already considered in other works as in [LIA 94]. Nevertheless, those models dealt with hierarchies of cooperative applications supporting disjoint non-collaborative groups. In this paper we are studying the effect of using a hierarchy of several instances (a conference) of the same CSCW application. In that hierarchy, from the communications point of view, each instance of a CSCW application shows both client and server roles when exchanging information.

Obviously, this hierarchical scheme could also be adapted in order to include situations of cooperation among working groups that are not performing tasks with the same overall goal (as proposed in [LIA 94]).

When talking about hierarchy in groups and among groups, we are just dealing, in some way, with the number and particular arrangement of relationships among groups and among individuals. But, how are these relationships?, are they of the same type?. In the field of group dynamics, two main intergroupal and interpersonal types of relationships are distinguished: cooperative and competitive relationships [DEU 60]. Cooperative situations have often been considered as more productive than competitive as a result of typical social implications that affect the individual components of groups [BUL 91]. Nevertheless those social implications are not always the same for each person [ACK 96]. In fact, in some cases, a competitive environment can be much more attractive for a particular person than a cooperative one thus increasing the productivity of individuals and, as a result, enhancing the overall results of the group. A typical example of a very productive competitive environment is that of an Internet security newsgroup in which lots of ``hackers'' compete so as to discover potential security holes (and solutions) in a (presumably) secure system.

1.1.1. *Hierarchy, competition and collaboration in the educational environment*

Principles exposed above can also be applied educational environments and, more concretely, in the CSCL (Computer Supported Cooperative Learning) world. In a typical class, there is an obvious hierarchical structure that separates the teacher from the pupils. This classical tight structure, with very little feedback for the teacher about how the class is going on and with no interaction at all, can be substituted by different new structures with more cooperative--competitive relationships between the teacher and the pupils and among the pupils themselves, in order to speed up learning processes.

A first step toward this approach are the questions directed by the teacher to the pupils in order to get some feedback during the class, or a collaborative discussions directed by the teacher. Further steps involve bidirectional questions and discussions, reaching a new scenario with a very high level of interaction. For example, with the creation of cooperative groups of pupils competing in looking for the best answer to question.

Work overload for the teacher increases with the level of interaction, moreover taking into account that it is quite important not only the interaction, but the analysis of the interaction processes themselves.

So, new CSCL tools are needed to help teachers in this tasks, that could record all interactions among components of the group and save them for subsequent revision and analysis. This on--line research approach is better than an after--the--fact analysis based in interviews and surveys.

1.2. *Groupkit: A development kit for groupware applications*

Groupkit is a programming toolkit for building real-time groupware applications [ROS 92]. It runs on Unix workstations under an X-Window interface and is based on Tcl/Tk [WEL 95]. We have been working with Groupkit version 3.0, although version 4.0 is already available.

As a toolkit, Groupkit offers a platform for fast development of prototypes and permits the simulation and study of real cooperative work scenarios and groupware applications. Various users run different instances of the same application and, in this way, they form a conference.

Groupkit cooperative work capability is based on information sharing and interaction among users in these conferences. Interaction capability is achieved through a Tcl extension called Tcl--DP (Distributed Programming Tcl) [COR 98] which permits the execution of a Remote Procedure Call (RPC) on other users' processes. Primitives offered by Groupkit are oriented to the use of these RPCs on every user in a conference, although the selection of a specific user is possible. Information sharing is also based on this model and therefore, it is made at a conference level. Due to this model, Groupkit is not a suitable toolkit for dealing with groups and hierarchies within a conference.

The original logical arrangement of Groupkit components simply consists of a set of individual users interacting and sharing information with no distinction among them. An example of that logical arrangement is shown in figure 1.

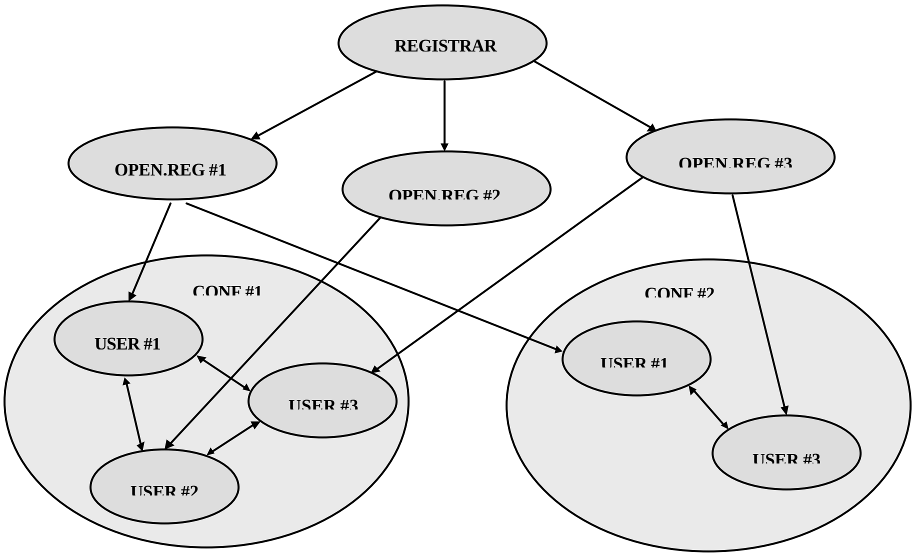

Figure 1. *Original logical arrangement of Groupkit v3.0 components*

In that figure, every dark grey ellipse represents a process running in any workstation under UNIX (the platform that supports Groupkit v3.0). A continuous line between two processes is equivalent to a communication channel according to the (RPC-like) client/server paradigm followed by Tcl--DP. An arrow at a line end-point means that the pointed out process acts as a client of the counterpart process which behaves as an RPC server. A line with arrows at both end-points represents a two-way communication channel.

Registrar is a daemon process that maintains a database of *Registrar* clients (called *open.reg*) and a database of ongoing conferences (a group of running instances of the same CSCW application) and participants. Basically, it is an RPC server whose main goal is maintaining the consistency of the information stored and displayed by its clients (i.e., *open.reg* processes).

Each *open.reg* process shows a graphical interface that allows a user to join different conferences and groups. Usually, each *open.reg* process is associated to just one particular user.

At the same time, each participant in a conference is represented as another process which is an instance of the CSCW application involved in the conference (e.g., a brainstorming tool, a shared editor, etc.).

In Groupkit, the typical mechanism used in order to share information among participants in a conference is the so--called ``shared environment'', data structures whose changes (in any of their components) are propagated to the whole set of participants in the conference by means of established communications channels (conferences in figure 1 are shown by means of light grey ellipses). In the original Groupkit v3.0, information stored in ``shared environments'' is accessible by the whole set of participants in a conference and there is no possibility of restricting the access to a particular subset of users. In order to overcome that restriction, CSCW applications that wanted to handle the concept of group of participants within a conference, had to include additional code thus increasing their developing complexity and decreasing their performance (a problem that affected our preliminary experiments as it will be shown in section 2.3.2).

1.3. *A new platform for hierarchical systems in Groupkit*

To avoid the previously described problems, we developed a new platform using Groupkit as a basis. The new platform developed on top of Groupkit v3.0 is intended to give support for the grouping of users within a particular conference. Figure 2 shows its logical architecture.

As it can be seen in the figure, a new hierarchical level has been introduced: groups of users (participants) within a single conference. All those groups have the same overall goal (the one that characterizes the conference) and compete or collaborate in order to achieve it. Simultaneously, participants within a group, compete or cooperate in order to participate in the conference as a single entity. Note that the architecture of figure 1 could be seen as a particular application case of the architecture shown in figure 2 where each group is composed by just one participant.

It is important to take into account the flexibility and scalability of the new solution: a single user (associated to one *open.reg* process) can take part in different conferences as a member of different groups. This feature enables the adaptation of the software developed on top of this platform to a broad range of organizational structures (which is one of the main goals of our work).

Several characteristics have been added to the new CSCW platform that was developed by us:

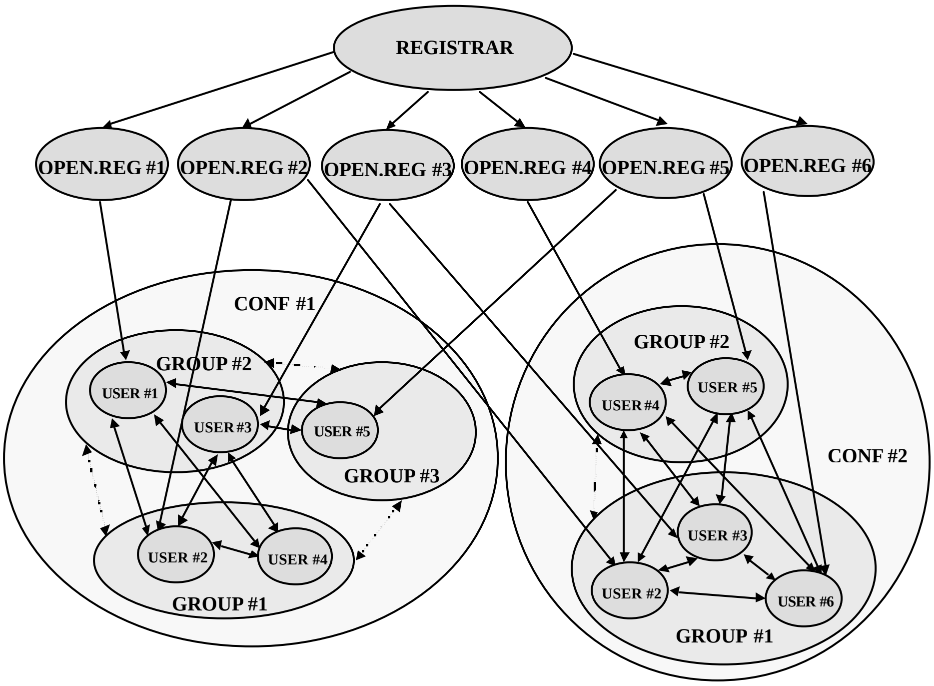

Figure 2. *An example of the logical architecture of the new platform developed on top of Groupkit v3.0 (dotted lines represent virtual communication channels.*

1. **Group support of *open.reg* processes**: as shown in figure 3 each *open.reg* contains information regarding the ongoing conferences, the groups belonging to them and the set of participants in each group (and additional detailed information about participants). At any time, the user that launched the *open.reg* process is able to join running conferences within any group that share them.
2. **Group communication**: support has been added in order to send messages to all members of a user's group, messages to all members of a user's group except itself, and messages to members of another group.
3. **Group information sharing**: with the new platform, it is possible to define environments just shared by members of a particular group (avoiding the access by other participants of the same conference that belong to different groups).
4. **Other miscellaneous functional modifications**: support for new events (asynchronous messages involved in interprocess communications) regarding addition, deletion, updating of users and groups has been added.

At the same time, new routines have been added to fix and retrieve information about conferences, groups, participants, etc.

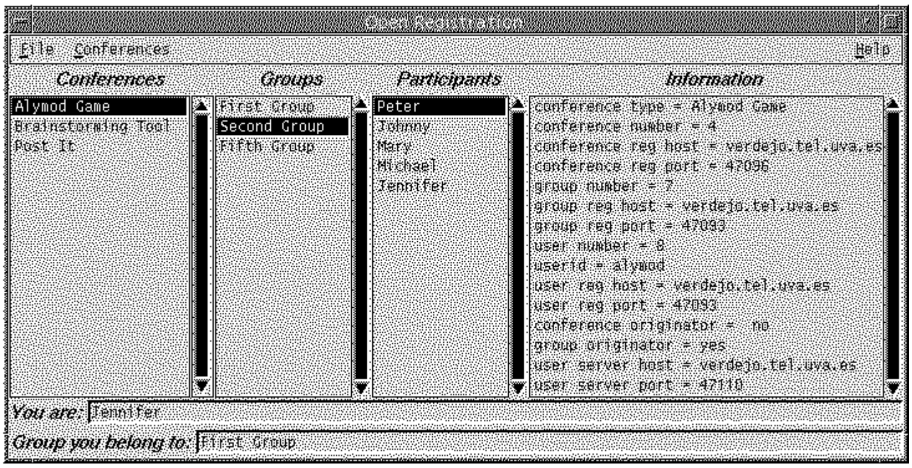

Figure 3. *The open.reg graphical user interface developed for the new platform*

With all these additions and modifications, CSCW applications developed over the original Groupkit v3.0 can directly be run over the new platform with no additional complexity or performance degradation, while incorporating the ability to manage variable-sized groups.

2. Experimental work

Introducing the concept of hierarchy within groups (each group member adopts a particular role according to its behavior with respect to the rest of group members) and hierarchy among groups, poses two key questions:

1. Are there useful CSCW applications that are inherently hierarchical?
2. To what extent do present development tools for groupware facilitate the implementation of such applications?

Taking into account the above questions, we formulated a case study, called *Alymod* (Aprende a Leer Y Mejora tu Ortografía Divirtiéndote, or Learn to Read and Enhance your Orthography Enjoying yourself). Although initially devoted to orthographic learning, *Alymod* happened to be easily customizable to support almost any kind of test (e.g. psycho-technical tests, course examinations, etc.). Such an application had to be generic enough in order to be able to study the performance of different types of groups (co-operative, competitive, hierarchical, etc.) devoted to the resolution of problems of varying difficulty and response time constrains.

We used an Object-Oriented (OO) modeling methodology [RUM 96] in order to perform the analysis and design of our proposed new application. OO modeling was chosen because of its fulfillment of the following requisites:

1. Ability for an easy specification of distributed applications
2. Clear representation of hierarchical environment and roles as well as of the interrelation among them

As a result of the above requirements, *Alymod* was designed as an educational application running on top of the new subgroup supporting platform added to the original Groupkit. Its main aim is binding one or more groups of students with the goal of solving a problem (e.g. orthographic or psycho-technical tests) establishing cooperative or competitive relationships among them.

An example of its graphical user interface is shown in figure 4. It can be observed that there are two different game boards with the same set of text gaps to fill out. In this example, the goal consisted of completing a set of numeric series.

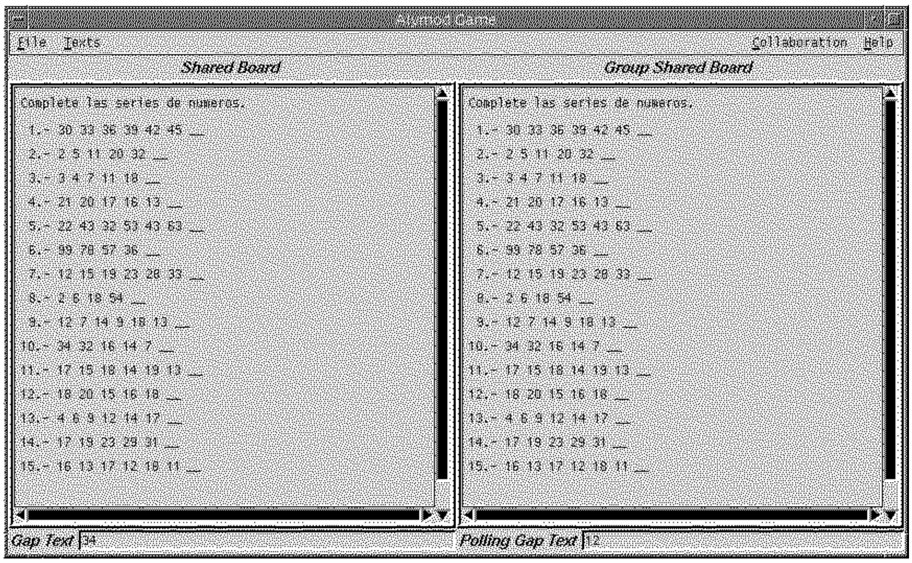

Figure 4. *Example of the Alymod Graphical User Interface*

Alymod is flexible enough so as to support different sizes and types of groups and participants, as is shown in the setting-up of experiments described in the following sections.

2.1. *Objectives of the experimental work*

We carried out experiments using Alymod to fulfill the following objectives:

1. To study different dynamics of collaboration and/or competition among people to solve a problem
2. To find out what organizational models work better in different situations of difficulty of a problem or time pressure
3. To learn how good is our application supporting those organizational models
4. To study the suitability of our software in real educational scenarios

2.2. *Design of experiments*

We are going to use two main types of strategies among participants within a group:

1. **Centralized**: in this type of strategy one of the members of the group represents the so-called group coordinator. This coordinator, chosen by the group, has to take into account the proposals from the other members of the group to accomplish the final work assigned to it as, for instance, filling out gaps in a text. The coordinator may also split the overall exercise in different tasks and assign them to the members of his group. He may also select the following task that all group members of his group have to analyze.

   The decision making process depends on the particular situation. The range of possible situations extends from that in which the coordinator makes decisions by himself to the one in which the coordinator makes decisions through the use of group participants proposals.
2. **Distributed**: in this case the coordinator does not exist. The final decision about different tasks depends on the final agreement achieved by the members of a group. The same idea applies to the previous assignment of those tasks to different participants: some kind of consensus has to be achieved about the assignment procedure.

Another criterion for strategy classification is related to collaborative and competitive relationships within the organization:

1. **Collaborative**: all the participants of a group make proposals on the same task in order to reach an agreement about the solution and then to proceed to another task.
2. **Competitive**: each participant can select a task and make final decisions about it by his own.

By setting up groups with different strategies or group structures, we will be able to find out, to some extent, how the structural features of *Alymod* are appropriated by the different groups depending on their particular strategy. In other words, according to [DES 94], the structural features of *Alymod* (including the group support added to Groupkit) can be used with different ``spirit'' by each particular group.

It is important to point out, at this moment, that our work focuses only on structural properties of groups and their relations with structural features of the supporting software. Psychological aspects of group members with respect to technology (what Orlikowski [ORL 92a] calls `` *people's cognitions or mental models about technology and their work*'') have not been taken into account (as in [CIB 88]).

Three sets of experiments have been performed in order to fulfil the objectives described in section 2.1:

1. In the first set, we used a version of *Alymod* programmed directly over Groupkit (without the modifications described in section 1.3). The goal of this set of experiments was to test whether the existing features of Groupkit were enough or not to support the requirement posed by *Alymod*.
2. The second set of experiments was devoted to test the group support of the new platform developed on top of Groupkit. But the experiments did not only want to test the software but also to test the flexibility of *Alymod* when coping with different group strategies and to obtain some conclusions about the advantages and disadvantages of each strategy. In these experiments, psycho-technical tests were employed.
3. The final set of experiments was designed in order to test the suitability of *Alymod* in real educational scenarios. In this case, the *Alymod* games consisted on examinations of two different courses of the Telecommunications studies of the University of Valladolid (courses taught by two of the authors of this article).

2.3. *Experiments without Groupkit modifications*

2.3.1. *Description of the experiments*

A first version of *Alymod* was developed using an unmodified version of Groupkit. All the software needed for group support was included in the application. The characteristics of the experiment were:

1. It consisted of three Alymod games. The first one let users learn the environment and the others formed the core of the experiment. Each game was composed of six English sentences with one or more gaps (a missing word) to be filled out by the participants. The first of the two games was

easier than the second one.

2. Eighteen people participated in the experiment organized in six groups of three persons. All the participants had a mid-level knowledge of English. The activity of each group was supervised by the co-authors of this paper.
3. Each group member received 10 points for filling out correctly a gap and -5 for filling it wrongly.
4. Each group followed a different organizational structure:

   - Group 1 (G1): centralized with a coordinator who filled out the gaps in the text. He had to consider the proposals from the other group members.
   - Group 2 (G2): centralized with a coordinator who chose a gap from the text. The other group members made proposals about how to fill it out, and the coordinator filled out the gap accordingly.
   - Group 3 (G3): centralized with a coordinator who assigned a subset of gaps to each group member. Those group members had to make proposals about the potential solution of their correspondent gaps. The coordinator had to analyze those proposals to fill out the gaps.
   - Group 4 (G4): Distributed. Each group member made his proposals about each gap in the text. The most voted proposal was chosen to fill out the gap.
   - Group 5 (G5): Distributed with random gaps. In that case each group member chose a gap and all the group members made proposals about it. The most voted proposal was used to fill out the gap.
   - Group 6 (G6): Distributed with a mutual agreement. All the group members had to reach an agreement in order to distribute the responsibility of filling out gaps among them.

2.3.2. *Analysis of experimental results*

Table 1 shows the results of the experiment for each group and the total time to finish each one of the two games.

	G1	**G2**	**G3**	**G4**	**G5**	**G6**	**Time**
1	0	20	-10	0	10	5	1' 45"
2	15	-25	-15	-10	-25	0	3'

Table 1. *Results of the first experiment*

More interesting than these results are the observations gathered by the co-authors of this paper while supervising the work of the groups. We discovered a series of failures that prevented us from achieving the objectives that we were searching for by means of the experiment. These problems were:

1. The difficulties of using the application. They appeared mainly because of

the complex code needed in the application to overcome the lack of support for hierarchical structures in Groupkit. This meant that the application was extremely slow and thus, not very user-friendly.

2. There were few options to choose for filling out a gap. Subsequently, under time pressure due to competition, participants tended to forget the strategies and to fill out gaps as quickly as possible.

As a result from this set of experiments, we decided to develop a new version of *Alymod* but using the new features of the modified Groupkit and to change the orthographic series by more complex psycho-technical tests.

2.4. *Experiments with the modified Groupkit*

2.4.1. *Description of the experiments*

To solve the problems of the first experiment, three new subsets of experiments were planned and performed. We tried to achieve our objectives using the new *Alymod* application developed over the modified Groupkit version described in section 1.3.

The aim of these experiments was, as previously described, twofold:

1. To test the flexibility of the developed software by applying it in groups with very different strategies.
2. To compare different types of group structures with different degrees of centralization *vs.* distribution and competition *vs.* cooperation.

In order to fulfill the second objective, three group structures or strategies were chosen for the setting up of groups:

1. A centralized cooperative strategy that followed the strategy of ``the coordinator assigns to the group members the gaps or holes to be filled out". Such a situation might emulate an organization in which a manager assigns the different tasks to his subordinates.
2. A distributed cooperative group with the strategy of a ``self-assignment of holes or gaps", i.e. that simulates a group whose members assign the tasks with a mutual agreement.
3. A purely competitive group, where each group member took its own decisions trying to improve both their individual results and the overall group performance. This might simulate the case in which the manager takes into account both individual and group performance.

These three strategies match the three stereotypical organizational structures and contexts for team work used by Ciborra [CIB 88] to study the effects of the introduction of work group support systems according to the ``transaction cost theory'':

1. The *market* structure, characterized by a low level of task uncertainty and goal congruence, can be compared to the pure competitive strategy.
2. The *hierarchy* structure, characterized by a mid level of task uncertainty and goal congruence, can be compared to the centralized cooperative strategy.
3. The *clan* structure, characterized by high levels of task uncertainty and goal congruence, can be compared to the distributed cooperative strategy.

This matching between our strategies and the group structures used by Ciborra will be very useful to obtain conclusions from the results of the performed experiments.

In this set of experiments, the texts to be completed in the *Alymod* games were composed of psycho-technical tests [COR 95] instead of orthographic exercises.

Three subsets of experiments were performed:

1. In the first subset, three groups of three people each were formed. These three groups had to compete in seven consecutive *Alymod* games with psycho-technical tests of up to ten questions (the first test of that series of seven was not taken into account in the results as it was devoted to give the participants a preliminary idea of the application). The individual and group performance was measured in terms of total time required in each game and their score. Total time required by the winner group indicated the difficulty of each task, as well as the influence of the competition among groups.
2. In the second subset of experiments, we pretended to compare the performance for the following alternatives:
   - only one cooperative homogeneous group
   - various groups competing among them

   We formed a new homogeneous group (also with three people) with the strategy that showed the best performance in the first subset of experiments, and gave it the same texts used previously. This new group was formed by people that had not participated in the first subset of experiments. This group had not to compete with other groups but it just had a time limit to complete the exercises.
3. In the last subset of experiments, we formed six groups of three people each. Each one of the three described strategies were assigned to two different groups. The same set of exercises was employed and there was no competition among groups (they had to complete the exercises within a limited time interval).

All the experiments performed in this set, were characterized by the following aspects:

1. All the responses given in the group shared board of the *Alymod* Graphical User Interface (see figure 4) were only visible by the members of the corresponding group. Those responses did not suppose a change in the score of the participants or their groups.
2. All the responses given in the overall shared board were visible by all the members of the competing groups. A correct answer supposed 10 points for the participant (and its group) and an incorrect answer subtracted 5 points to the score.
3. If a particular hole was already filled out, a participant had the opportunity of introducing a new answer if he believed the existing response was not correct. If the old answer was not removed, that implied it was right and the participant that tried to change it, got an score of -5 points. If the already existing value of the hole was changed by the new value, that meant that the old value was wrong. In this case the participant got 10 or -5 points depending on whether his new answer was correct or not. This way of scoring put additional time pressure on competing groups: if they did not give quick answers, they might eventually miss the opportunity of obtaining points.
4. Each *Alymod* game ended when all the holes had been correctly filled out or when the assigned time interval expired.
5. The participants in each group could directly dialogue without using any special tool (they all were located in the same room).

2.4.2. *Experimental set-up*

The first and second subsets of experiments were performed in February 1997 and the third took place in December 1997.

1. For the first and second subsets of experiments, we used our Alymod software package, developed using the new platform above Groupkit for Solaris 2.5 operating environment of Sun Microsystems. Since each new conference consisted of various processes, the processor load was distributed among 3 SparcStations 10, that were not exclusively devoted to our experiments. We used a variety of terminals including workstations and personal computers with X-window servers.

   Twelve people participated in the experiments, nine of them in the first subset, and three in the second one. Each group used terminals physically located in the same room. Participants in the experiments were chosen with the objective to have a (more or less) similar intellectual ability and experience with the use of the graphical environment. Three of them were in the third year of telecommunication engineering studies, five in the fourth year, and four were postgraduate students.
2. For the third subset of experiments, we run all the *Alymod* conferences in a

Sun UltraSparc 2 with two processors. Eighteen Sun SparcStation 2 were simultaneously used as X-window terminals.

Eighteen students from the fourth and fifth year of the telecommunication engineering studies formed the six groups employed in this subset of experiments.

The co-authors of this paper supervised all the experiments, taking care of its correct development and observing the behavior and reactions of the users. This was necessary, because technical difficulties might distort the results, besides the fact that CSCW deals with human-to-human interaction, and therefore human reactions had to be also taken into account.

2.4.3. *Analysis of experimental results*

Table 2 shows the results of the first and second subset of experiments, and the total time needed to finish each game. The strategy used by the group of the second experiment (group **G2 (new)** in table2) was ``distributed with self-assignment of gaps'', i.e., the same as the group **G2** of the first subset, which was the one who got the highest score.

	G1	**G2**	**G3**	**Time (minutes)**	**G2 (new)**	**Time (minutes)**
1	-5	75	5	4	80	15
2	10	45	5	13	20	15
3	10	5	10	10	60	15
4	-15	20	30	6	50	15
5	40	30	0	6	80	15
6	20	35	20	3	70	10
Total	60	210	70		360	

Table 2. *Results of the first and second subset of experiments*

Table 3 shows the total score of each participant of each group.

	P1	**P2**	**P3**
G1	60	0	0
G2	70	70	70
G3	15	-5	60
G2 (new)	115	100	145

Table 3. *Scores of the participants in the second subset of experiments*

G1 used a centralized strategy, so only one of its members (the coordinator) obtained points. **G2** and **G2 (new)** used a distributed strategy and thus the score is divided between the members of the groups. **G3** used a competitive strategy and its

score is affected by the bad results of one of its participants (there was not a mechanism to tell a participant that some guesses were wrong).
The following preliminary conclusions were obtained from these two subsets of experiments:

1. The new application worked much better than the old one. With the new application we could undertake a higher number of games with less user disappointment.
2. The new exercises consisting in psycho-technical tests worked much better than the English grammar exercises of the first set of experiments. The participants needed more time to calculate the reply for them and therefore, the strategies were really observed.
3. The second subset of experiments showed that without the time pressure imposed by the competition, participants tended to use all the available time. On the other hand, they got higher results (in the sense that with less people they were able to get similar scores as the sum of the groups of the first subset of experiments, see table 4).

	G1+G2+G3	G2 (new)
1	75	80
2	60	20
3	25	60
4	35	50
5	70	80
6	75	70
Total	340	360

Table 4. *Comparison between the first and second subset of experiments*

The results of the third subset of experiments, for each of the six groups and each of the 6 psycho-technical exercises, are shown in table 5.

	Ce1	Ce2	Di1	Di2	Co1	Co2
1	70	80	85	75	-85	-240
2	80	15	20	55	45	30
3	90	25	60	65	55	40
4	65	-5	60	45	65	55
5	55	70	100	75	80	60
6	30	35	45	55	65	55
Total	390	220	370	370	225	0

Table 5. *Results of the third subset of experiments*

Ce1 and **Ce2** were the groups with the centralized strategy, **Di1** and **Di2** were the groups with the distributed/cooperative strategy, and **Co1** and **Co2** were the groups with the competitive strategy.

The results of these experiments reinforce those obtained in the first and second subsets of experiments: the distributed strategy obtains better average results than the other two strategies.

But there is another very interesting result. If we analyze the temporal evolution of the average scores of the groups with each strategy (shown in figure 5) we can derive the following conclusions (for the distributed strategy, we have also taken into account the results of the second subset of experiments that also employed it):

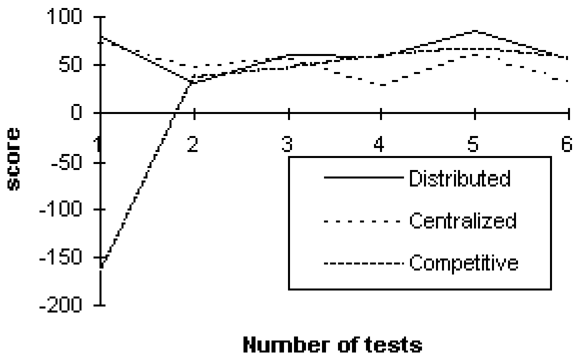

Figure 5. *Temporal evolution of the average scores obtained by each strategy*

1. The results of the distributed and centralized strategies were highly correlated. This was due to the variability in the difficulty of each series of exercises.
2. The results of the distributed strategy were better than those obtained by the centralized and competitive strategies as was observed in the first and second subsets of experiments.
3. The competitive strategy obtained better results as more series were completed. The results were worse at the beginning probably because, as stated by [CIB 88], the uncertainty of tasks (the lack of knowledge about how to use the application and how to complete the tests) was very high and then the ``market-like'' group structure (which can be compared to the competitive strategy) was not adequate at all. According to [CIB 88] the best strategy or structure when the uncertainty of tasks is high, is the ``clan-like'' structure, similar to our distributed/cooperative strategy which was the one that got better results.

4. The lack of correlation between the competitive results and the other two strategies could be explained by the changes in the strategy made, in some situations, by the participants in the competitive groups: after completing almost all the exercises in a series, they usually started to cooperate in order to solve the most difficult ones. They cooperated by using the group shared board or by direct dialogue. So, in this case, the same tool (*Alymod*) was used with different ``spirit" [DES 94] by the same group. This is a new demonstration of how the effect of CSCW in groups does not only depend on the structural features of the application but also on how the technology is appropriated by the group members [DES 94,ORL 91,ORL 92a].

2.5. *Experiments in real educational environment*

2.5.1. *Description of the experiments*

The final set of experiments were devoted to the testing of the *Alymod* application in a real educational scenario. In this case, we tested it in real examinations of pupils of two courses of the telecommunications studies of the University of Valladolid: the so-called ``Telematics Laboratory" and ``Computer architecture Laboratory".

We wanted to test the three strategies used in the second set of experiments in order to get an idea of how pupils compete/cooperate with or without a coordinator when they work in groups.

2.5.2. *Experimental set-up*

We set up six groups: three for each course with the three described strategies. Each group was composed of three pupils. Each group had to complete a test with a mix of theoretical and practical questions about their respective course.
The rest of conditions used in the second set of experiments (hardware/software resources, scoring, etc.) were also used in this set of experiments.

2.5.3. *Analysis of experimental results*

The average results of the tests of the two courses for each strategy are shown in table 6.

	Ce	**Di**	**Co**
1	-30	60	5
2	16	30	28
Total	-14	90	33

Table 6. *Results of the experiments in a real educational environment*

The **Ce** column shows the results of the centralized strategy, the **Di** column the results of the distributed strategy and finally, the **Co** column represents the results of the competitive strategy.

The results obtained herein are basically aligned with those obtained with the psycho-technical tests:

- The distributed strategy obtains the best results, i.e. the ``clan-like'' structure of Ciborra [CIB 88] is the most adequate in these cases (the pupils, that have a good personal relationship, cooperate in order to face the uncertainty in the resolution of the examinations).
- The competitive strategy improves its results from the first exercise to the second one.
- *Alymod* can perfectly be used as a group examination tool or as a collaborative support for the resolution of academic problems in groups, thus improving the pupil initiative (see section 1.1.1). The pupils can also get used to working in group structures that they might eventually find in the enterprises they will work for after leaving the University (with just a moderate increment in the teacher's amount of work).

As described in section 1.1.1, from an educational point of view, it would be of great interest the logging of the sequences of ideas given by all the pupils during the resolution of the *Alymod* exercises. We are currently working on this direction and we have already modified the Groupkit brainstorming tool to use the subgroup supporting facility of the new developed CSCW platform. This modified brainstorming tool might be used as the means of interchanging ideas during cooperative sessions (ideas that might eventually be stored for off-line analysis).

3. Conclusions

The objective of this paper was to study the relationship between CSCW systems and co-operative--competitive transactions that take place within organizations.

A specialized CSCW toolkit, called Groupkit, was employed for the test of our models and case studies. Toolkits or other high level tools are necessary for the development of CSCW applications that involve human-human communications. Although this widely-accepted toolkit has been successfully tested in simple co-operative groups, we experimentally found that transition to more complex organizational structures, such a hierarchy, is not straightforward. More concretely, such application development was time-consuming and prone to implementation and user-interface errors.

In order to tackle the above problem, we developed a new software platform above Groupkit, that easily handles complex relationships among groups that employ strategies with various degrees and types of co-operation--competition. This new platform serves for the construction of applications that can suit to a specific organization type, thus overcoming the inherent difficulty of building robust CSCW systems. On the other hand, such a platform may be useful for study and performance comparison of specific aspects, when different organization structures are employed.

Our platform was applied either on building a new CSCW application (Alymod game) or on adapting existing CSCW tools (e.g. Brainstorming) to new organizational structures. The main experimental work focused on using Alymod game in different experimental conditions, where groups had to maximize group score, while minimizing time to accomplish the required task. Using a variety of tasks as well as strategies within groups, we were able to test effectiveness of each strategy, that corresponds to organizations of markets, clans or hierarchies.

Besides these simple tests, we employed the new platform within the context of collaborative learning and the relationships--structures of two University courses. Experimental studies in real conditions unveiled interesting learning patterns of cooperation--competition beyond those present in flat student groups. On--going research work in this field includes registration of communication and decision patterns within groups, and analysis of the most efficient process in terms of learning results. Besides educational objectives, such an analysis could help us in better preparing students for a realistic cooperative-- competitive environments in real-world organizations after graduation, as many companies often require from Universities.

Acknowledgements

We would like to thank all volunteers that participated in the experiments, as well as the anonymous referees that permitted a significant enhancement of this paper through their suggestions.

References

[ACK 96] ACKERMAN, M.S, STARR, B., "Social activity indicators for groupware", *Computer*, pages 38-58, June 1996.

[BAE 93] BAECKER, R.M. (editor). *Readings in Groupware and Computer-Supported Cooperative Work: Assisting human-human collaboration*, Morgan Kaufmann Publishers Inc., San Francisco, CA, 1993.

[BLA 96] BLASCO, M.T., BARRIO, J.L., GONZÁLEZ, O., VERDÚ, M.J., GARCÍA, E., DIMITRIADIS, Y.A., LÓPEZ, J., "Study of collaborative writing supported by an informatique pen-based system". In *Proceedings of the 1996 European Conference on Educational Research*, page 249, Sevilla, Spain, September 1996.

[BUL 91] BULLEN, C.V., BENNETT, J.L., "Groupware in Practice: An Interpretation of Work Experiences", in DUNLOP, C., KLING, R. (editors), *Computerization Conflicts and Social Choices*, pages 7-8, Academic Press, 1991. Also reprinted in [BAE 93, pages 69-84].

[GIB 91] GIBBS, S.J., ELLIS, C.A., REIN, G.L., "Groupware: Some issues and experiences", *Communications of the ACM*, Vol. 34, nº 1, pages 38-58, January 1991. Also reprinted in [BAE 93, pages 9-28].

[CAR 60] CARTWRIGHT, D., ZANDER, A., *Group Dynamics: Research and Theory,* Harper and Row Publisher, Inc., New York, second edition, 1960.

[CHA 91] CHARAN, R., "Now networks reshape organizations - for results". *Harvard Business Review*, vol. 69, nº 5, pages 104-115, 1991.

[CIB 88] CIBORRA, C.C., OLSON, M.H., "Encountering electronic work groups: a transaction costs perspective". In *CSCW 88 Proceedings*, 1988.

[COL 92] COLE, P., NAST-COLE, J., "A primer on group dynamics for groupware developers", in [MAR 92], pages 44 -- 57, 1992.

[DES 94] DESANCTIS, G., POOLE, M.S., "Capturing the complexity in advanced technology use: Adaptive structuration theory". *Organization Science*, vol. 5, nº 2, pages 121-147, 1994.

[DEU 60] DEUTSCH, "Effects of cooperation and competition over the group process", in [CAR 60, pages 503-525].

[TOU 94] TOU, I. et al., "Prototyping synchronous group applications". *Computer*, vol. 27, nº 5, pages 48 - 56, May 1994.

[RUM 96] RUMBAUGH, J. et al., *Object-oriented modelling and design*. Prentice Hall, 1996.

[LIA 94] LIANG, T.P. et al. "Why client/server isn't enough: Coordinating multiple distributed tasks", *Computer*, vol. 27, nº 5, pages 73-79, May 1994.

[GON 97] GONZÁLEZ, O., VERDÚ, M.J., DIMITRIADIS, Y.A., BARRIO, J.L., BLASCO, M.T., LÓPEZ, J., "Pen-computing and groupware: A chance to improve collaborative writing", in *Proceedings of the 7th International Conference on Human Computer Interaction*, San Francisco, California, USA, August 1997.

[GON 97a] GONZÁLEZ, O., VERDÚ, M.J., DIMITRIADIS, Y.A., C. OSUNA, IGLESIAS, C.A., LÓPEZ, J., ``PEN Computer Aided COmposing coLLAborative System". Groupware for learning", in *Proceedings of the 3rd International Workshop on Groupware*, Madrid, Spain, October 1997.

[GRU 90] GRUDIN, J., "Groupware and Cooperative Work: Problems and Prospects", in APPLE COMPUTERS (editor), *The Art of Human Computer Interface Design*, Addison-Wesley Publishing Co., 1990. Also reprinted in [BAE 93, pages 97-105].

[MAR 92] MARCA, D., BOCK, G., *Groupware: Software for computer-supported cooperative work,* IEEE Computer Society Press, Los Alamitos, CA, 1992.

[COR 98] UNIVERSITY OF CORNELL, *Tcl-dp documentation*. http://www.cs.cornell.edu/ /Info/Projects/zeno/Tcl-DP/html40.

[OKA 94] OKAMURA, K., FUJIMOTO, M., ORLIKOWSKI, W.J., YATES, J., "Helping CSCW applications succeed: the role of mediators in the context of use", in *CSCW 94 Proceedings*, 1994.

[ORL 92] ORLIKOKWSKI, W.J., "The duality of technology: Rethinking the concept of technology in organizations", *Organization Science*, vol. 3, nº 3, pages 398-427, 1992.

[ORL 92a] ORLIKOWSKI, W.J., "Learning from notes: Organizational issues in groupware implementation", in *CSCW 92 Proceedings*, 1992.

[ORL 91] ORLIKOWSKI, W.J., ROBEY, D., "Information technology and the structuring of organizations". *Information Systems Research*, vol. 2, nº 2, pages 143-169, 1991.

[PAL 94] PALMER, J.D., FIELDS, N.A., BROUSE, P.L., "Multigroup decision-support systems in CSCW", *Computer*, vol. 27, nº 5, pages 67-72, May 1994.

[COR 95] CORROCHANO RESINO, M.A., CIDÓN COLINO, F., *Psycotechnical Tests*. PSICOS, Majadahonda, Madrid, september 1995. (In Spanish).

[ROD 93] RODRÍGUEZ, J.M., *Human factor in companies*. Ediciones Deusto S.A., Bilbao, 1993. (In Spanish).

[ROS 92] ROSEMAN, M., GREENBERG, S., "Groupkit: A groupware toolkit for building real-time conferencing applications", in *Proceedings of the ACM International Conference on Computer Supported Cooperative Work, CSCW92*, Toronto, Canada, 1992.

[TYR 94] TYRE, M.J., ORLIKOWSKI, W.J., "Windows of opportunity: Temporal patterns of technological adaptation in organizations", *Organization Science*, vol. 5, nº 1, pages 98-118, 1994.

[WEL 95] WELCH, B.B., *Practical Programming in Tcl and Tk*. Prentice Hall, Mountain View, CA, May 1995.