

Taking The 'No' Out Of Lotus Notes: Activity Theory, Groupware, and Student Groupwork

John Halloran, Yvonne Rogers, Mike Scaife

Interact Lab, School of Cognitive and Computing Sciences, University of Sussex, UK
{johnhall,yvonner,mikesc}@cogs.susx.ac.uk

ABSTRACT

This paper describes a study that examined why groupware that was tailored to support collaborative student learning (Lotus Notes) was unsuccessful. In particular, it examines why the tutors' aspirations of helping the students to collaborate were not met. It was found that students tended not to use the groupware, preferring other, self-developed support tools. Our study shows that the failure was multidetermined: there was a complex interacting set of factors including software use problems, systems integration issues, conflicting tutor/student perceptions of the value of using the groupware, and conflicts in each group's view of how best to complete the course. There has been interest in using Activity Theory to approach multidimensional analysis in CSCL, but existing Activity Theory-based frameworks can be difficult to apply to instances of collaborative learning marked by conflict. To address this need, we use an Activity Theory-based analytic tool called the Activity Space. The tool is also used to show how multiple changes could be made to improve the potential for groupware to be used as intended.

Keywords

Student Groupwork, Groupware, Lotus Notes, Activity Theory, Activity System, Activity Space

INTRODUCTION

In this paper, we discuss a study of how students worked together on an undergraduate computer science course. On this course, students, working in groups, had to design and program a computer game. The course had two main objectives: first, to give collaborating students experience of dealing with a complex, distributed software development task; and second, to help them develop group management, coordination and communication skills - regarded as valuable preparation for postgraduate employment. Tutors were concerned to support both these objectives through the use of Lotus Notes, which was configured accordingly. However, despite Lotus Notes being apparently suited to students' and tutors' needs, the students avoided using it.

Resistance to use of Lotus Notes for collaborative work is not a new finding. Orlikowski (1992) discusses the non-uptake of Lotus Notes in an organization, elucidating the issues surrounding rejection. These are 'cognitive' - the absence of appropriate understanding of the software ('weak technological frames'), but also 'structural': the organization was hierarchically organized with competition for promotion, such that the collaborative, knowledge-sharing principles Lotus Notes represents were 'counter-cultural'. Thus, there was no incentive for staff to learn Lotus Notes. The implication here is that cognitive and structural factors need to be consistent with the purposes of groupware if there is to be genuine incentive, and the groupware is to be adopted. What is perhaps surprising about the rejection of Lotus Notes in our study is that the conditions for adoption appear to have been in place. Students did not have major problems understanding what sort of application Lotus Notes was and how it functions (appropriate technological frames existed); and the organization in which they work, a computer science department, is geared towards supporting learning through collaboration. Students are used to working together in groups, and need to do so to progress in their academic careers. Thus, there seems to have been ample incentive.

Our study, then, examines a persistent problem (non-uptake of Lotus Notes) in a new context (CSCL) with apparently different reasons. What were these reasons, and what are the general implications for supporting collaborative learning with groupware? Our study shows that the failure of Lotus Notes was determined by multiple, interacting factors such that is difficult to explain this failure in terms of a mismatch between the organizational ethos and the purposes of the groupware, or in terms of inadequate student understanding of the application. The reasons seem more elusive. On top of this, however, in common with Orlikowski's study (and other studies of groupware adoption problems, for example Grudin [1988]), there appears to have been conflict surrounding the use of the application. There was conflict in the perceptions of tutors and students on what the essential work necessary to complete the course was, conflict on the use of the tool as an aid to assessment, and conflict concerning the real value of the groupware for the groupwork.

To understand the failure of Lotus Notes to be used as intended on the course, and to address general problems of groupware adoption in student groupwork, we propose that a multidimensional analysis is needed. Within CSCL, there has been interest in multidimensional analysis and Activity Theory has been recognized as having potential for doing this (see, for example, Gifford and Enyedy, 1999). However, where tensions and contradictions are identified between the parties to collaborative learning, existing Activity Theory-based frameworks can be difficult to apply. In this paper, we use a new Activity Theory-based framework, called the Activity Space. This allows us to deal with the issue of conflict while preserving the distinctive interactionism of Activity Theory. We use the Activity Space framework first as an analytic tool to identify, and specify the interactions between, the factors involved; and second, as a means of generating recommendations, in terms of groupware, to better support collaborative learning.

THE STUDY

Background and Method

The course that Lotus Notes was used for is a ten-week first-year undergraduate computer science course in software design and evaluation. Students, typically working in groups of four, have, first, to develop one of three pieces of software which, when put together, compose a game. Next, the group acquires the two other modules from other groups, and integrates these with their own. A group term paper is submitted documenting the whole process.

The introduction of Lotus Notes (chosen over alternatives because the University has a site licence and knowledgeable administrators) was intended to make collaboration easier for students, through provision of shared space for storage and editing; messaging facilities for everyday communications, including threading by topic; and space for writing minutes of meetings. In addition, supporting and assessing students' groupwork skills (management, coordination, communication) had previously presented difficulties for tutors as these processes tended to be invisible (assessment was done through evaluating students' *post-hoc* self-report in their term paper). Lotus Notes was intended to provide a solution: students were made aware that material they put onto their Lotus Notes group accounts would be tutor-viewable, and that the reason for this was that tutors could comment on, and assist with, group processes as they unfolded. Some material, notably minutes of meetings, would contribute to tutors' formal evaluation of students' group management skills, an assessable component of the course. What tutors hoped for, then, was full use of Lotus Notes not only for students to support themselves but also to provide rich data enabling tutor involvement in the group process.

Over the ten weeks of the course, we shadowed selected groups of students using ethnographic techniques including video observations (researcher-present or non-present); non-video observations (researcher-present), notetaking, informal interviews, inspection of students' written materials and course instructions. Students' Lotus Notes accounts were also available for us to inspect. We found that the use of the groupware differed a good deal from what tutors had intended.

Findings

At the very beginning of the course (in the first week), students used Lotus Notes to make contact, to begin to communicate about what they needed to do, to organize meeting times, and as a space to store early versions of their software designs. However, a major finding across all groups was that, as the course went on, postings to Lotus Notes became infrequent and highly formulaic, consisting mainly of minutes of meetings organized according to a standard template with sparse entries, apparently quickly produced. As a messaging system, Lotus Notes was also neglected: students did not use it for formatting and organizing day-to-day communications. This was done through students' standard e-mail accounts, held on the UNIX teaching network, even though this meant that specific communications about the groupwork on the course would appear in inboxes unsorted. There was little or no use of Lotus Notes as a shared storage or editing space – as had been planned by the tutors.

Interestingly, while students did not use Lotus Notes as planned, they appeared to appreciate its functionality enough to try to recreate some of it elsewhere. Several groups of students made shared spaces on the UNIX network including functionality like code storage and message archiving – functions Lotus Notes already offered. This is surprising because it carried a work overhead, one of the reasons both Grudin and Orlikowski give for non-adoption of groupware. To create the UNIX space, an individual student needed to create a directory on his/her UNIX account. Student accounts are protected; there is no permission for others to access them. Therefore, the students creating the spaces had to approach support staff at the University to get the permissions changed for the relevant group of students. Postings to the shared space also needed to carry read/write permissions for the group. On top of this extra work, some of the functionality of the UNIX alternative seemed impoverished compared with Lotus Notes. One example is that there were no facilities for discussion threads: students needed to keep track of course-related communications as they came through with general e-mail, and needed to save important ones through their own

efforts. Another is that the spaces did not function well as awareness tools or as an immediately available resource centre, having to be specially accessed, while Lotus Notes could remain available permanently on desktops by keeping a browser open. However, in some ways the UNIX shared spaces worked better than Lotus Notes. There were difficulties in integrating Lotus Notes with UNIX. This had an impact on programming. Code stored in Lotus Notes could not be run there, but needed to be re-exported to UNIX, and the only form of communication between the two systems was via cut and paste. To move large amounts of code either way was troublesome.

These general findings concerning the students' avoidance of Lotus Notes to communicate and share, and instead a preference for a more basic support structure which they had to set up themselves, are puzzling. Why was this? One reason is that what we may regard as positive aspects of Lotus Notes - for example ease of producing minutes of meetings - were not seen as such by students. During one non-researcher present video observation, a group of students realized they needed, for assessable purposes, evidence of a 'standards committee meeting', a meeting in which standards for software development are agreed between different groups who will later trade software. However, they had not had such a meeting. So they organized a meeting with other groups involved to construct this evidence retrospectively. This meant tutors sometimes got rushed or even invented material not particularly representative of the actual collaborative practices that may have obtained and which they wished to be able to support. Equally, during another meeting where there was no researcher present, one student, from this same group, observed 'we had better put this stuff on Lotus Notes', adding that the system was 'a bloody nuisance'. This seems to reflect a widespread attitude. In other words, Lotus Notes represented work students might not have wished to do. However, again, this is problematic, since this work had to be done in order for the students to be successful.

Other reasons for students' avoidance of using Lotus Notes might be attributed to usability problems with some of the Lotus Notes interfaces. For example, Lotus Notes featured a pulldown menu of categories for postings (Figure 1a). There were four categories: 'agendas', 'meetings', 'general discussion', and 'other'. Some groups used the default, 'agendas', for all postings. Others used categories indiscriminately; for example, meeting minutes appeared under 'agendas', 'meetings', and 'general discussion'. There is also a messaging problem. To enter a posting from the front screen (not shown) the icon 'New Main Topic' must be clicked. This creates the top level of a thread - we are inside a message within such a thread in Figure 1b. To respond to a posting within the thread, the Lotus Notes user must be inside a posting (as shown) and then click 'New Response'. This continues the thread. However, an option near this is 'Close', which takes users back to the front screen. Using 'New Response' was only done in a few cases. Students tended to hit 'Close', return to the front screen, and use 'New Main Topic' to enter a posting. Thus, threading tended not to happen. Even if the posting was intended to be a response to a message inside a thread, it looked like the beginning of a new one. The implication is that the behaviour of the interface may have limited students' ability to organize material by topic. However, this limitation seems to be only part of the picture.



Figure 1a: Category Pulldown



Figure 1b: Responding to a Message

Our findings, in contrast to those of Orlikowski, do not give a clear picture of why Lotus Notes was not used. There seems to have been ample incentive for students to use it, and students appear to have understood the system and realized its usefulness. Despite this, they preferred to avoid the system - but paradoxically created alternatives which reproduced some of Lotus Notes' functionality. In not using Lotus Notes they also potentially risked loss of marks (by, for example, not keeping up with tutor-viewable minutes of meetings - and, more generally, not providing tutors with assessable materials). These findings seem to go counter to students' own interests and motivations, and, at least on the face of it, are difficult to analyze in such a way that we might arrive at better groupware solutions in such a context: they present an analytic challenge.

THE ANALYTIC CHALLENGE

Problems in Analyzing the Failure of Lotus Notes

Why did students say 'No' to Lotus Notes, despite the apparent suitability of the groupware for their purposes? In this section, we will show that there was an interacting set of factors which we will address in sequence: software, systems, 'cognitive', and 'structural'.

First, we will consider the software level. By this, we mean usability issues to do with the Lotus Notes interface. One approach to explaining such issues is to take an 'artefact perspective' (Bødker, 1996). When a software interface stops users fulfilling their purposes, Bødker calls this a 'breakdown'. Breakdowns can be identified by 'focus shifts' - where users look around for solutions - and by frustration and confusion. Breakdowns are evidence that the software needs improvements. If there were breakdowns in students' use of Lotus Notes, this could explain their avoidance. In terms of the category pulldown menu, it is not immediately obvious, as Lotus Notes has its own design protocol for a pulldown, that any category *other* than 'agendas' is available. However, there was little if any evidence of focus shifting, frustration or confusion in students' use of the pulldown, whether or not it was realized to be a pulldown. Equally, those students who did categorize appropriately, simply found the functionality through exploratory clicking (since the pulldown is represented by a button with a down arrow), and no breakdown appeared to have occurred. Those who posted to inappropriate categories did not seem concerned about this - implying that categorization issues were not just to do with the pulldown.

As we saw, to respond to a posting inside a thread, the Lotus Notes user must first access the message and then click on 'New Response' inside that message. Use of the 'Close' option means a message is exited. The continuation of the thread is lost when 'New Main Topic' is used. However, again, this did not appear to be a source of frustration and was not associated with focus shifts, suggesting that students were simply not highly concerned to organize messaging as threads.

This discussion suggests that there may be issues other than, or additional to, those identifiable at the software level that are responsible for the avoidance of Lotus Notes. Our next level of analysis is the systems level. This refers to applications rather than application features (the software level), as well as integration of different applications. Two issues we have already seen are learning overheads, and UNIX/Lotus Notes integration problems. We might observe that because students already knew how to use UNIX, they simply wished to avoid the work overheads represented by Lotus Notes. This would explain why shared spaces were created on UNIX. However, work avoidance because of difficult-to-use systems seems to be contradicted by the fact that the creation of shared spaces on UNIX also carried significant work overheads. A possible way of accounting for this is the poor integration of Lotus Notes with UNIX. Students use UNIX for their day-to-day work, especially programming and e-mail. Posting evidence of collaborative work to Lotus Notes involved cutting and pasting. Students were especially concerned to produce working code, but if this was posted to Lotus Notes it needed to be immediately transferred back to UNIX to be run. Thus it may have been easier for students to simply mail code with any comments directly across UNIX, bypassing Lotus Notes. However, in itself, this does not explain why students avoided Lotus Notes, which offered other, direct functions like messaging and writing minutes which could be displayed on the desktop alongside UNIX windows.

Our next level of analysis considers issues to do with student perceptions of Lotus Notes - a cognitive level of analysis. By 'cognitive' is meant not just understanding of systems (Orlikowski's 'technological frames') but also beliefs and attitudes. The student who regarded the production of material for Lotus Notes as 'a bloody nuisance' (without dissent from others in the group) appears to have done so for two reasons. First, doing messaging and writing minutes at all appears to have been something perceived as extraneous to the issue of producing software - this work was not done on the UNIX shared spaces. The students' understanding appears to have been that this was not a valuable thing to do. There was also a public/private issue. Students had been told that material posted to Lotus Notes, notably minutes of meetings, would be tutor-accessible. However, students appeared to feel uncomfortable with this - that they were being 'checked up on' without their consent. Thus, students preferred to go private, explaining why, faced with other functionality problems, they were happy to make use of UNIX-based private shared spaces. However, this still does not explain why students appeared to regard the tutor- and groupware-supported development of group management, coordination, and communications skills as something inessential which *could* be avoided, especially as this was assessable.

The three levels of analysis we have looked at - software, systems, and cognitive - have not provided us with clear answers as to what really was responsible for the avoidance of using Lotus Notes. Another possible answer might lie at the 'structural' level of analysis. After Orlikowski, this refers to the organizational context, e.g. the priorities of the different people working in the organization, and their interactions. There were two parties to the collaborative learning - tutors and students. Collaboration was not only intended to be between students; but also between tutors and students, reflected in tutors' introduction of Lotus Notes into the course. However, tutors and students are different sets of people with different experiences and concerns. Tutors appear interested in process: teaching the skills, transferable to professional settings, that lead to successful product development. In contrast, students seem product- rather than process-oriented: they were more interested in getting the software working by any means rather than developing group management strategies. This suggests that the avoidance of Lotus Notes was, in fact, to do with the way students define their work, with the use of Lotus Notes being seen as extraneous to this. Thus, in

Orlikowski's terms, Lotus Notes could be seen as, for students, 'counter-cultural' – and if so, it should be expected to fail. However, this explanation is still problematic: it is unclear why students might neglect an assessable component of the course, which results in marks and so should have been strongly motivated. We might expect Lotus Notes use to have been pro-cultural not counter-cultural.

Developing an Analytic Tool

There are two basic issues which need to be addressed when analyzing the problems of using Lotus Notes for supporting students. The first is that there are contributory factors at different levels of analysis, none of which appears, on its own, to explain the failure of Lotus Notes. The second, related, issue is conflict that can arise between the parties to the collaborative learning – tutors and students. On the course, there were conflictual attitudes between what each party saw as necessary to complete it, including different perceptions of the level of salience of some work, and differences between attitudes toward inspectability. Thus we need an account of how and why conflict arose and persisted.

To try to answer the need for an analytic tool capable of coordinating and integrating findings at different levels of analysis, and to elucidate types of conflict, we have developed a framework called the Activity Space (Halloran, 2000). This is based on Activity Theory (Kuutti, 1994; Leont'ev, 1982; Nardi, 1996; Vygotsky, 1978). There has been interest in Activity Theory for CSCL, and use of Activity Theory-based frameworks, notably Engeström's Activity System (1990: see, for example, Bellamy, 1996; McAteer *et al*, 2000; Mwanza, 2001). Why not just use this? One reason is that the Activity System approach focuses on the interactions that enable an activity to be stable and functional, rather than on conflictual activities. A diagram of the Activity System appears as Figure 2:

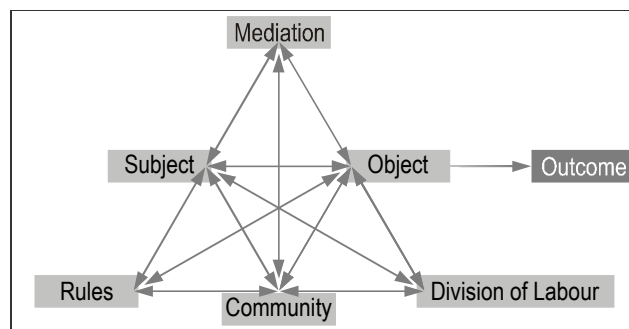


Figure 2: Activity System

In an activity system, the 'object' is what is driving the activity, for example, to provide educational services. Where this is a group activity, this object is 'common': everyone, regardless of what their specific job is, is working towards the same thing. There is also a 'collective subject' – the parties to the activity are all united in wishing to achieve the common object. 'Mediation' means the tools used to complete the activity. These can include what we commonly regard as tools (computers, filing cabinets, phones, fax machines) but there also cognitive tools (beliefs, attitudes, skills, concepts, etc.). These are well coordinated. The other nodes are 'rules', 'community', and 'division of labour', all responsible for the coordination of the system – these nodes can be seen as relating to Orlikowski's 'structural' level. The interaction of the nodes in the activity leads to the outcome, i.e. the product of the activity.

There are problems in using the Activity System framework to analyze conflict between different groups of people within the same activity. Because nodes do not feature internal conflicts, for example those which occurred between student and tutor perceptions of Lotus Notes, we need to conceive of two different activity systems with each group as the subject of that system. We then need to describe the 'contradictions' between those two systems. This seems counter-intuitive as the activity is just one – in our study, specifically the software design and evaluation course – and we would like to be able to represent this. A second problem is that it is difficult to conceptualize what the 'rules', 'community' and 'division of labour' would be on the course, as all of these things are ambiguous given the conflicts between tutors and students in terms of their perceptions, attitudes, and actions. In other words, there is not a strong 'structure' in Orlikowski's sense. Rather, there seems to have been two different groups (tutors and students) which, although they might, to greater or lesser degrees, have rules, community, and a division of labour within themselves, do not have an overall set governing and uniting both groups.

The Activity Space is another extension of Activity Theory designed to show, in contrast to the Activity System, how it is that an activity becomes conflictual through problematic interactions. The Activity Space is shown in Figure 3:

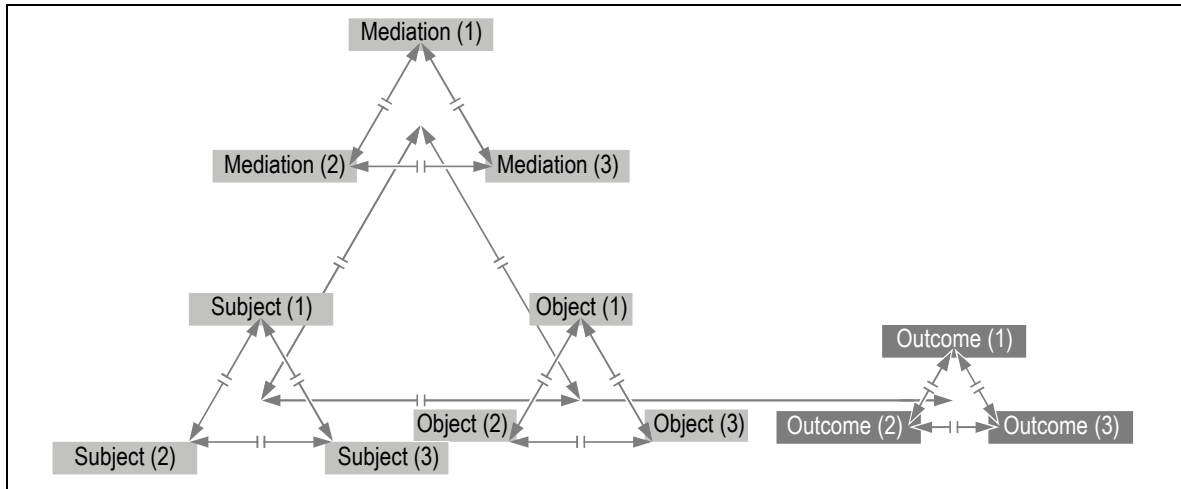


Figure 3: Activity Space

The Activity Space consists of a base triangle with three main nodes: subject, object, and mediation. Because of the difficulties of conceptualizing ‘rules’, ‘community’ and ‘division of labour’, these nodes are omitted. Between each of these nodes, there can be conflicts, represented as parallel lines (=). In the diagram, each node is exploded into a number of sub-nodes. This enables us to conceptualize different subjects, objects, and mediation within the same main node, and, again, there can be conflicts. In addition, the outcome of the activity can be conflictual. In this particular diagram, there are three sub-nodes to each main node, but this number is arbitrary. The number could vary according to the activity we were looking at, the diagram changed accordingly.

The Activity Space predicts that conflicts at one node will propagate to the others and vice versa, implying that they are co-determining (they all affect each other). This means we cannot explain conflicts at a particular node, for example between tutor and student views of the significance of Lotus Notes, without reference to the other nodes. Thus, the representation enables a multidimensional analysis, offering a systematic way of examining relations between the different levels of analysis. These include groupware features, integration of applications and systems, cognitive and structural factors. The implication is that none of these levels of analysis is sufficient on its own.

STUDENT GROUPWORK AS AN ACTIVITY SPACE

The main findings identified in our study of the use of Lotus Notes by students are represented as an Activity Space in Figure 4. In this diagram, there are only two subject sub-nodes, two object sub-nodes, and two outcome sub-nodes. The other sub-nodes, which have no significance in this context, are greyed out.

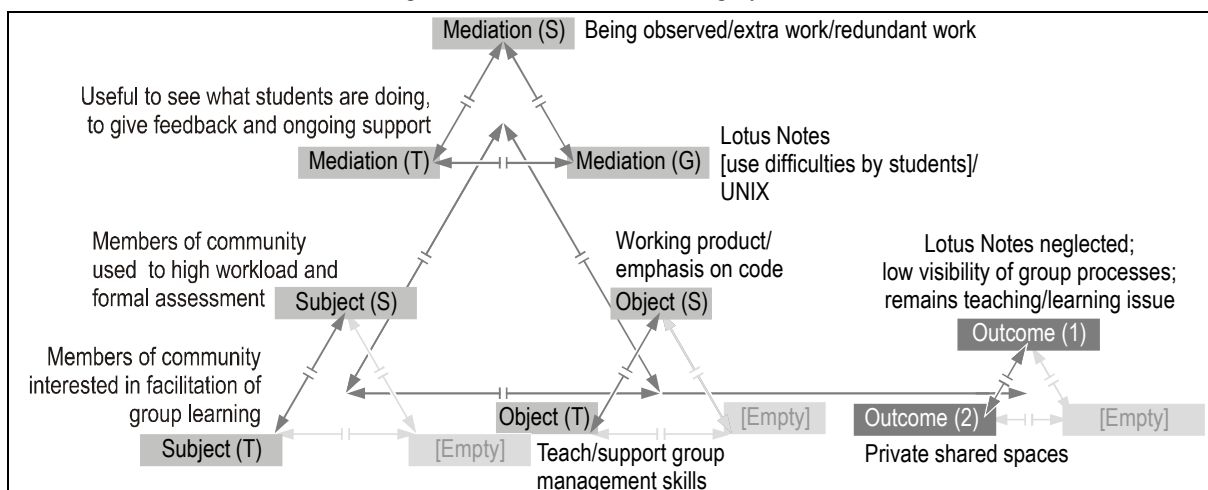


Figure 4: The course as an activity space

We saw that there were issues with the use of mediation in the shape of Lotus Notes on the course. There are cognitive conflicts between tutors/students in terms of beliefs concerning the value and significance of using Lotus

Notes. At the Mediation (S) node (S for 'Students'), students believe they are being observed, and are uncomfortable with this – possibly because group processes felt to be 'messy' rather than well-organized might not be positively assessed. On the other hand, tutors (Mediation (T)) wish to be able to see exactly these sorts of process so that they can support group coordination and communication. Equally, students appear to believe that the work represented by using Lotus Notes is redundant, which contradicts the tutor view. There are also software/cognitive conflicts, in two senses. First, students did not make use of Lotus Notes functions by virtue of their belief about the use of the groupware; but also the cognition required to master these functions was not particularly exercised. This reveals that the functions of Lotus Notes as a piece of software in isolation are not necessarily problematic; but become so because of the tool's position in an activity where it is seen as representing something undesirable.

As we have noted, students made shared spaces on UNIX. This software mediation is consistent with their beliefs about Lotus Notes. The two forms of software mediation are in conflict (Mediation (G) [G for 'Groupware']). However, the conflict between these systems is not just a function of their design, but also relates to cognition. Students believed that UNIX was private, limited to work they themselves saw as valuable, while Lotus Notes was public in a non-desirable way, involving redundant work. The belief about UNIX meant that students were prepared to undergo the learning overheads in creating shared spaces on UNIX and the difficulties of use. Therefore, the coordination of these systems was not something students particularly attempted. Again, we cannot say that the avoidance of Lotus Notes was due to other functionality issues like difficulty of access from UNIX. In other words, having only two levels of analysis - the software and systems levels of analysis – is inadequate.

At the mediation node, we can see there are several interacting types of conflict. But where do the mediation conflicts come from? At the subject node in Figure 4, there are conflicts between the two groups, students and tutors. These are different sets of people with different experiences and concerns – they have different 'subject characteristics'. As we suggested in our initial analysis, students tend to be product-oriented, while tutors are interested not only in product but also in process. This goes further, reflecting affiliations to different models of learning (Rogoff, forthcoming). Implicit in the tutors' idea of using Lotus Notes is a 'participant' model of collaborative learning, where parties work together non-hierarchically to solve a problem. The tutor acts as a facilitator. This implies that in collaborative learning, tutors are an active party. However students seem to interpret what they are doing in terms of an 'acquisition' model: that is, they are left to acquire a solution (a product) through their own efforts, which will then be assessed. Lotus Notes was there to help support the process. The students' model of learning had an impact at the Mediation (S) node both in terms of beliefs about Lotus Notes and attempts to use it. It was seen as an assessment tool. Students were uncomfortable with the extra assessment it appeared to entail. Thus, the belief concerning observation as assessment is reinforced by this group's subject characteristics. This conflicts with the belief held by tutors (Mediation (T)): their observation is important to help students develop essential process skills. This belief led to the introduction of the groupware. Here, then, we see how cognitive factors to do with the tutors' and students' different subject characteristics can help explain the students' avoidance of Lotus Notes despite the tutors' view that it would be a useful support tool.

According to Activity Theory, the driver of any activity is its object. An object is what the party or parties to an activity wish to achieve. On the course, objects are in conflict. Tutors wanted to be able to support students in learning how to work together and this was their purpose in introducing Lotus Notes. Being able to see the details of their meetings would help tutors to address issues as they came up. Thus, the tutor object in relation to Lotus Notes was 'help students collaborate' by teaching/supporting group management skills (Object (T)). Being able to look at Lotus Notes entries served this object. However, for students, the object was 'avoid creating assessable materials except where this is absolutely essential. Instead, concentrate on the most important assessable components, i.e. working product' (Object (S)). Thus, the two objects are in contradiction. Again, we can begin to relate these nodes to others. The objects proceed from beliefs at the mediation level, beliefs concerning the meaning of making group processes public. However, each group's subject characteristics, which derive from its cultural affiliations and social position within education, are also conflictual.

Thus, the Activity System helps us to conceptualize how nodes (which can be seen as effects at particular levels of analysis) are co-determining. For example, the subject characteristics of each group are related to their beliefs and attitudes at the mediation node; and to their objects. However, at the same time, the objects reinforce the mediation, and, indeed, the subject characteristics – especially when there is an apparently satisfactory outcome (students managed to get through the course). In parallel, tutors still saw facilitation of group process as a live issue – in fact, more urgent than before. Their own objects, mediation, and subject characteristics also persisted, reinforcing each other. The factors leading to students' avoidance of Lotus Notes, then, appear to be self-perpetuating.

DISCUSSION

What use can be made of such an analysis in terms of better supporting collaborative learning with groupware? The analysis suggests that supporting collaborative learning involves more than changing software functions of groupware (for example, improving interfaces). In addition to these, other kinds changes to the course are needed.

We can conceptualize conflicts at each node of an activity space as being *artifacts* of conflicts at the others – artifacts in the sense that these conflicts are undesirable and would not necessarily exist without determination by those other nodes. However, because there is interaction, changes at single nodes might not work. Although changes at particular nodes to reduce conflict might have positive effects at others, we might simultaneously need to concentrate on changes at those other nodes. In this section we describe two scenarios whereby the Activity Space framework can be used to conceptualize change. These are derived from our study, but are intended to have general relevance.

Scenario 1: ‘Heavyweight’ Change

The first scenario for change involves improving the functionality of groupware and dropping any inspectability requirement. These changes, and the effects on the activity space, are shown in Figure 5:

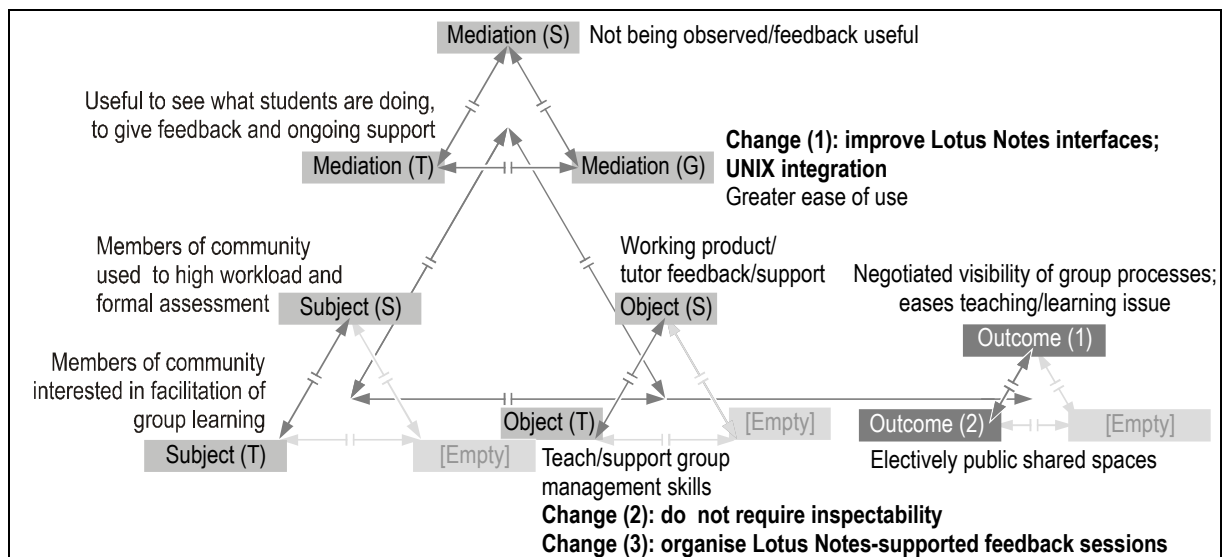


Figure 5: Using the Activity Space framework to approach heavyweight change

In this proposal, three ‘active’ changes (shown in bold) would be made. First, the groupware interface functionality is improved, together with integration with other systems (Change (1)). In terms of the software design and evaluation course, this could be achieved, for example, by changing the pull-down menu to radio buttons; and to adding a warning to the ‘Close’ button when inside a message. Integration with UNIX could be assisted through creating weblinking facilities under Lotus Notes. Second, tutors could drop the inspectability requirement; that is, they could choose not to require that Lotus Notes postings are viewable remotely (Change (2)). Third, they introduce feedback sessions for groups at which groupware, and other materials, could be used, depending on the choices of students (Change (3)).

These active changes can be hypothesized to produce ‘referred’ change; that is, to affect other nodes. Changes both active and referred feed back and reinforce each other. This means that not every node need necessarily be subjected to active change. The way the scenario would work is by raising the salience of group processes in a way which reduces the students’ notion of redundancy and obviates their fears concerning visibility. Thus, compared with Figure 4 (the unchanged activity), we see that student beliefs about being observed might change (Mediation (S)) without there being any active change needed at this node. This would have effects in terms of mediation and objects coming more closely into line for each group, not to mention outcomes. At the same time, subject characteristics might start to change a little with each group coming to recognize the other’s position and motivations.

The changes proposed are quite significant. There would need to be a fair amount of work done on the systems (interface improvements to Lotus Notes; integration of Lotus Notes with UNIX). In addition, tutors would need to free up time to create the feedback sessions which would effectively replace the remote access practice. This may not be realistic. Our second scenario features changes which are more lightweight.

Scenario 2: 'Lightweight' Change

Scenario 2 is shown in Figure 6. In this scenario, there would be no changes to the groupware. Students would be required, as part of the course, to produce assessable coursework on group processes (management, coordination, communications) (Change (1)). Again, tutors would not require access to the groupware (Change (2)), but would, during the lecture series, include material on group management which could be made use of as students produced the required coursework (Change (3)). This proposal suggests that existing time (both tutors' and students') would be re-used (lecture topic change/coursework change), so the changes are less demanding of time/resources than in Scenario 1. Again, we could expect these active changes to produce referred change (for example, students learn to work around Lotus Notes interface issues) with feedback effects. The importance of group processes would be raised by both the need to produce assessed work, and by having been lectured on this topic. This would drive the use of groupware, and this interaction would in turn raise the salience of group processes and provide motivation to address the issue.

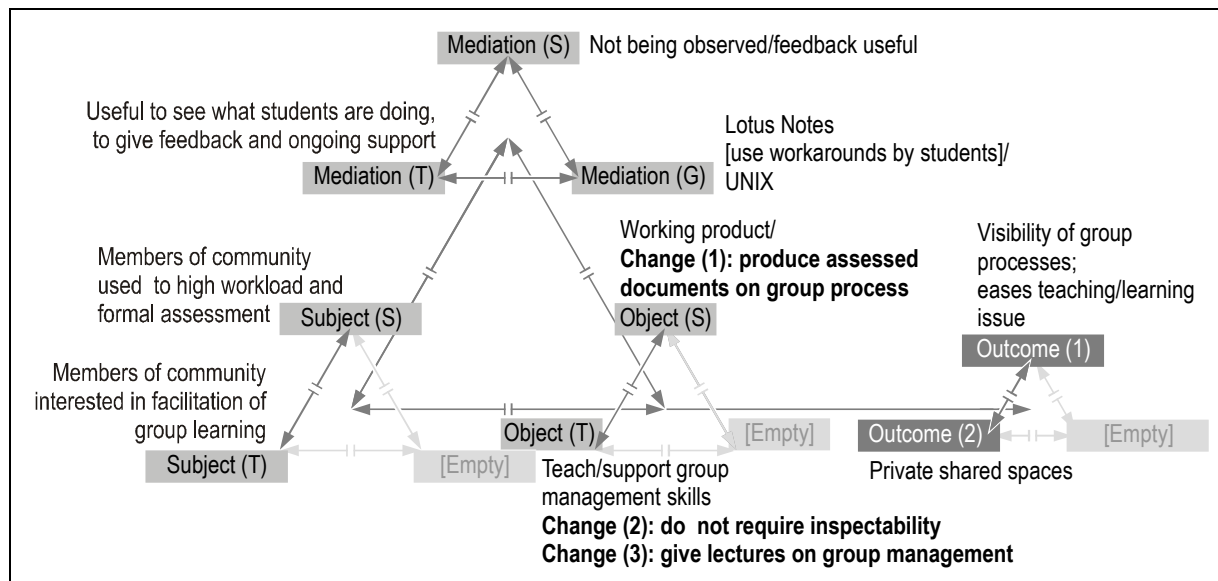


Figure 6: Using the Activity Space framework to approach lightweight change

The referred changes might differ somewhat, and possibly be more superficial, than those resulting from the heavyweight changes suggested in Scenario 1. The hope is, here, that the students' belief that using the groupware was not a priority, would come to be overridden by a real need to use it. However, if this need was interpreted solely in the interests of assessment, there is some doubt about its real use to students, or in terms of tutors' aspirations to teach group processes as valuable skills in themselves. Thus, problematic beliefs at the mediation node might persist, as could students' subject characteristics. There might be negative feedback effects on the active changes in that students might not like being 'forced' to use Lotus Notes. On the other hand, the scenario is designed to produce an increase in student perception of the value of using groupware beyond assessment purposes.

CONCLUSION

We began our paper with an empirical problem: the failure of Lotus Notes to support collaborative learning on a software design course as expected/intended by tutors. As we saw, groupware adoption problems are familiar: it is a persistent problem. In the CSCW literature, failure can be traced to conflict between what the groupware stands for, and the way the organization actually works. These can be inconsistent. However, we focus on a different context: a CSCL context where groupware is intended to support student groupwork. A major issue is that it is difficult to determine why groupware fails in this context, too. Organizational and groupware aims seem consistent. The failure, then, is harder to analyze, and appears multidetermined by a complex set of interacting factors including conflicting systems, perceptions, and cultural affiliations.

In CSCL, there is interest in multidimensional analysis which relates groupware to social context. Activity Theory has been used to do this. However, existing frameworks can be difficult to apply to instances of collaborative learning marked by conflict. The purpose of our framework - the Activity Space - is to produce a practical analytic tool which preserves the distinctive interactionism of Activity Theory, but which can deal more easily with conflict. Here, we show how an Activity Theory-based framework can act as a tool for systematically organizing and relating

observations at different levels of analysis in such a way as to produce a detailed account of groupware roles and functions in collaborative learning contexts. It provides a way of treating these roles and functions as the outcomes of many interacting factors. It allows us to hypothesize how change could occur in an activity space, both 'active', and 'referred'. It also shows what the scope and desirability of that change would be. The Activity Space framework needs further research involving implementing the scenarios and assessing it in light of the results to see how well it works as a predictive tool. As it stands, its main value is to show that in thinking about how to support collaborative learning, we need to go beyond groupware features *per se* to analysis of a range of interacting factors. These include perceptions, attitudes, the different roles of tutors and students in educational establishments, and their different cultural affiliations.

ACKNOWLEDGMENTS

The research this study is based on was supported by an ESRC/CASE DPhil studentship with British Telecom. Thanks go to Dr Michael Gardner of BT. Thanks also go to Dr Rose Luckin of Sussex University for giving permission to observe the student groupwork described in the study.

REFERENCES

- Bellamy, R. E. (1996) Designing educational technology: computer-mediated change. In B. Nardi, (Ed). *Context and Consciousness: Activity Theory and Human-Computer Interaction*. Cambridge, Massachusetts: MIT Press, 123-146.
- Bødker, S. (1996) Applying activity theory to video analysis: How to make sense of video data in HCI. In B. Nardi, (Ed). *Context and Consciousness: Activity Theory and Human-Computer Interaction*. Cambridge, Massachusetts: MIT Press, 69-102.
- Engeström, Y. (1990) *Learning, Working and Imagining*. Helsinki: Orienta-Konsultit.
- Gifford, B. R., and Enyedy, N. D. (1999) Activity centered design: towards a theoretical framework for CSCL. Proceedings of Computer Support for Collaborative Learning, CSCL 99 (Palo Alto, December 1999), unext.com, 189-196.
- Grudin, J. (1988) Why CSCW applications fail: problems in the design and evaluation of organizational interfaces. Proceedings of the Conference on Computer-Supported Cooperative Work. September 26-28, 1988, Portland, OR USA.
- Halloran, J. (2000) *The Activity Space: Analyzing Intentionality in Open Cooperative Work*. Unpublished Doctoral Thesis, University of Sussex.
- Kuutti, K. (1994) *Information Systems, Cooperative Work and Active Subjects: The Activity Theoretical Perspective*. Oulu: University of Oulu Printing Center.
- Leont'ev, A. N. (1982) *Problems of the Development of the Mind*. Moscow: Progress.
- McAteer, E., Tolmie, A., Crook, C., and Macleod, H. (2000). Grounding communication skills development in practice and play. European Conference on Educational Research (ECER 2000), Edinburgh.
- Mwanza, D. (2001). Changing tools changing attitudes: effects of introducing a computer system to promote learning at work. Proceedings of the First European International Conference on Computer-Supported Collaborative Learning, Euro-CSCL 01, March 22-24, 2001, University of Maastricht, the Netherlands.
- Nardi, B. (Ed) (1996). *Context and Consciousness: Activity Theory and Human-Computer Interaction*. Cambridge, Massachusetts: MIT Press.
- Orlikowski, W. (1992) Learning from Notes: organizational issues in groupware implementation. Proceedings of the ACM 1992 Conference on Computer-Supported Cooperative Work. October 31-November 4, 1992, Toronto, Canada.
- Rogoff, B. (forthcoming) *The Cultural Nature of Human Development*. New York: Oxford University Press.
- Vygotsky, L. S. (1978) *Mind in Society: The Development of Higher Psychological Processes*. Cambridge: Harvard University Press.