Beyond "Social Protocols": Multi-User Coordination Policies for Co-located Groupware

Meredith Ringel Morris
Stanford University
353 Serra Mall
Stanford, CA 94305
merrie@cs.stanford.edu

Kathy Ryall, Chia Shen, Clifton Forlines
Mitsubishi Electric Research Laboratories
201 Broadway
Cambridge, MA 02139
{ryall, shen, forlines}@merl.com

Frederic Vernier
University of Paris 11
LIMSI-CNRS, BP 133
91403, Orsay, France
frederic.vernier@limsi.fr

ABSTRACT

The status quo for co-located groupware is to assume that "social protocols" (standards of polite behavior) are sufficient to coordinate the actions of a group of users; however, prior studies of groupware use as well as our own observations of groups using a shared tabletop display suggest potential for improving groupware interfaces by incorporating *coordination policies* – direct manipulation mechanisms for avoiding and resolving conflicts. We discuss our observations of group tabletop usage and present our coordination framework. We conclude with example usage scenarios and discuss future research suggested by this framework.

Categories and Subject Descriptors

H.5.3. [**Information Interfaces and Presentation (e.g., HCI)**]: Group and Organization Interfaces.

General Terms

Design, Human Factors.

Keywords

Groupware, tabletop interfaces, conflict resolution, coordination.

1. INTRODUCTION AND MOTIVATION

Along with the benefits of enabling and enhancing group productivity, co-located CSCW applications also introduce new challenges. In particular, allowing multiple co-located people to simultaneously access a shared display gives rise to several types of conflicts. For instance, one user may change an application setting that impacts the activities of other users. The ease of "reach out and touch" on direct-manipulation devices such as shared multi-user tabletops makes reaching into another user's space or manipulating another user's documents tempting, further motivating software-level coordination mechanisms.

We propose a variety of coordination policies that aim to provide applications with more structure and predictability than social protocols, yet also allow for more flexibility than rigid access permissions. The ideas we present regarding coordination policies focus on policies applicable to direct manipulation on shared tabletops, although many of the concepts are also relevant for shared vertical displays.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

CSCW '04, November 6–10, 2004, Chicago, Illinois, USA.

Copyright 2004 ACM 1-58113-810-5/04/0011…$5.00.

Previous work on conflict resolution and avoidance in multi-user applications, such as [3], has focused on remote collaboration, and is concerned chiefly with preventing inconsistent states that can arise due to network latencies. In contrast, our work does not focus on conflicts caused by network latencies, but rather on the conflicts that arise in a co-located, single-display, direct-manipulation environment. Scott et al. [7] cite policies for accessing shared digital objects as a major design issue facing the emerging field of tabletop CSCW systems. Furthermore, Stewart et al.'s landmark paper on Single Display Groupware [12] warns that a potential SDG drawback is that "new conflicts and frustrations may arise between users when they attempt simultaneous incompatible actions."

Relying solely on social protocols to prevent or resolve conflicts is not sufficient in many situations. Greenberg and Marwood [3] observed that although in some cases social protocols provide sufficient mediation in groupware, they cannot prevent many classes of conflicts including those caused by accident or confusion, those caused by unanticipated side effects of a user's action, and those caused by interruptions or power struggles. In the Kansas system [8], Smith et al. originally felt that social protocols were sufficient for access control, but then observed that problems arose from unintentional actions. When conducting user studies of the Dynamo system [4], which relies largely on social protocols for handling conflicts, Izadi et al. observed that users had problems with "overlaps" – situations where one user's interactions interfered with another's. They noted several "overlaps," such as one user closing a document that belonged to someone else in order to make room for his own document. Users testing the Dynamo software also expressed concern that other users might steal copies of their work without permission.

Our own observations of groups of people using shared tabletop applications offer further support for the potential benefit of software-level coordination policies. Over the course of our work developing Table-for-N [9], the Magnetic Poetry Table [9], and other software designed for use on a DiamondTouch [2], we have seen both accidental and intentional conflicts arise.

Table-for-N (TFN) is an application for up to four people sitting around a table collaboratively annotating, manipulating, and browsing various types of documents. We have observed a variety of coordination difficulties among TFN users. For example, TFN offers multiple "views," analogous to the different screens provided by a "virtual desktop" application. We have seen one user switch to a new view while others were in the midst of manipulating items in the current view, thereby disrupting their work. Another example relates to TFN's "magnet" feature. When a user presses her magnet, all of the open documents on the table reorient to face that user. We have seen that people often magnetize all documents to face themselves even though another user may have been examining one of them.

We observed additional conflicts among users of a tabletop version of "magnetic poetry." This software allows up to four users to simultaneously rearrange a variety of word tiles to create poems, using either an English or a Japanese tile set. We saw one person switch the tile set while other people were in the midst of creating poems. We also observed people "stealing" words from one another for use in their own creations.

Another table-based poetry application we have made allows people to collaboratively reproduce a target poem by finding the correct words from a group of word tiles and arranging them on a piece of virtual "paper." One common conflict that we have noticed occurs when one user is in the midst of adding words to the paper and another user takes the paper away from him. Social norms suggest it is rude to infringe on other people's personal space, but while testing both our poetry application and another application that allowed four users to browse music, we found people violating this social protocol by reaching into other users' areas of the table rather than asking them to pass something.

All of our observations took place with groups of two to four people sitting around an 80 or 107 cm diagonal table. Intuition told us that small groups working in a small, easy-to-monitor area should be able to use social negotiation to coordinate their actions, but even in these circumstances we observed frequent mishaps! We hypothesize that as the size of the group grows, social negotiation will become even more challenging since it will become more difficult to monitor everyone else's actions, although a corresponding increase in table size might mitigate this by making it physically difficult to reach other users' documents.

2. COORDINATION POLICIES

2.1 Design Considerations

Through our experiences with multi-user tabletop applications, we have observed two key conflict dimensions. The first is *conflict type*, which refers to the level at which the conflict occurs – whether it affects the state of the entire application or only a single document. The second is *initiative*, which refers to whose actions determine the outcome of the conflict.

The three conflict types we observed are *global*, *whole-element*, and *sub-element*. *Global conflicts* involve changes that affect the application as a whole. Examples include changing the current "virtual table" being viewed, or issuing a command that changes the layout of all documents on the table. These actions are potentially disruptive to other group members. The notion of introducing policies to mediate global conflicts is supported by Shen and Dewan's suggestion that multi-user applications ought to define collaboration rights not only for traditional editing operations, but for any operation that might affect multiple users [10]. *Whole-element conflicts* involve access to a single object. Examples include multiple users trying to handle the same document, or multiple users trying to select from the same menu. *Sub-element conflicts* occur when several users are editing the same item simultaneously and issue conflicting changes. Because many issues in this category have already been explored in the context of group document-editing software such as [1], we do not address them here.

An important subset of whole-element conflicts are *manipulation conflicts*. Recent advances in sensing hardware, such as DiamondTouch technology [2], SmartSkin [5], and DViT [11] have enabled a level of parallelism in face-to-face collaborative software that was not possible with previous single display groupware. These touch-sensitive technologies support multiple, simultaneous touches, leading to the potential for conflict between multiple users trying to manipulate a document. Several of our policies attempt to control a document's manipulation access rights as a means of preventing conflict – manipulation access refers to the ability of a user to interact with an element by moving it around the display.

There are also three *initiative strategies* for resolving these conflicts: *proactive*, *reactive*, or *mixed-initiative*. *Proactive policies* allow an element's owner or the initiator of a global change to control the outcome of the conflict. *Reactive policie*s produce an effect based only on the actions of the other users (e.g., the person who tries to take a document away from its owner, or the users who are affected by a proposed global change). *Mixed-initiative policies* factor in information from all parties involved in the conflict to determine the outcome.

This categorization is useful for designers considering which coordination policies would work well in their application. Our design suggestion is to include at least one proactive and one reactive policy so that users have a mechanism for choosing to share an object (via the proactive policy), and there is also a deterministic outcome when users try to take an object (via the reactive policy). Designers can provide coverage for a variety of conflicts by selecting one global policy and one or more whole-element policies for use in their application. Whole-element policies could be mixed and matched in several ways – a single policy could apply to all elements, each user could have a different policy applying to all the documents they own, different types of documents could be associated with different policies, or each individual item could have its own distinct policy.

Table 1 illustrates this design space, showing where our proposed coordination policies fit. In the following section, we describe each policy in detail. Because they are meant to be relevant to a variety of applications, we discuss them in terms of abstract "documents," which include text and images, or "elements," which include things like menus. In Section 3 we will describe

Table 1. Our proposed coordination policies, grouped along the dimensions of *conflict type* (rows) and *initiative* (columns).

	Proactive	Mixed-Initiative	Reactive
Global	privileged objects anytime	rank	no selections no touches no holding documents voting
Whole-Element	sharing explicit dialog	rank speed force	public private duplicate personalized views stalemate tear

applications that make use of these mechanisms. We anticipate that continued experimentation with and evaluation of novel coordination strategies will allow us to further articulate this design space.

The following two sections list and define global and whole-element coordination policies that we have prototyped. The specific policies were motivated by several factors, including experiences with traditional paper documents, as well as the results of a survey we presented to members of our lab in order to assess their expectations regarding conflict-resolution.

2.2 Global Coordination Policies

No Selections, No Touches, No Holding Documents: These three policies dictate conditions under which a change to global state will succeed – if none of the users have an "active" selection on the table, if none of the users are currently touching anywhere on the table, or if none of the users are "holding" documents (touching an active document with their hand).

Voting: This policy makes group coordination more explicit by soliciting feedback from all users in response to a proposed global change. Each user is presented with a voting widget that allows him to vote in favor of or against the change. Several policies (majority rules, unanimous, etc.) could determine the outcome.

Rank: This policy factors in differences in privilege among users and can be used in conjunction with other policies, such as "no holding documents," thus changing the policy to mean that a global change will succeed if the user who initiated the change outranks other users who are currently holding documents.

Privileged Objects: Under this policy the determining factor is the way a change is initiated, rather than the circumstances of other users at the time of the proposal. For instance, there might be a special menu that must be used to make global changes, rather than including these options in each user's individual menubars. This might encourage more discussion among users by necessitating that they ask someone to pass them this privileged object. Also, requiring the use of a special interface mechanism might make people more aware of the effect their interaction is going to have on other users.

Anytime: This policy allows global changes to proceed regardless of circumstance – we included it for completeness and to provide an option for designers who want to rely on social protocols.

2.3 Whole-Element Coordination Policies

Public: This policy places no limits on who can access an element, instead relying on social protocols.

Private: With the "private" policy, any attempt by a user to manipulate a document he does not own or to select from a menu invoked by another user will be unsuccessful.

Duplicate: With this policy, the contested item duplicates itself. Three variants of this policy use different semantics for duplication: (1) creating a view linked to the original (changes made to either copy are reflected in both), (2) creating a read-only copy, or (3) creating a fully independent, read-write copy.

Personalized Views: This policy allows a user to obtain a document from another user or to select from another user's menu, but it first transforms that document or menu to display content customized for the user who takes it. For instance, if user A's menu has a list of bookmarks made by user A, and user B tries to use the menu, the menu would change to show user B's bookmarks. Or, if user A had annotated a document and user B took it, the document would hide user A's annotations and display only annotations made by B.

Stalemate: This is a "nobody wins" strategy for resolving conflicts. If a user attempts to take a document from someone else, the document becomes temporarily inactive to both users. This could encourage collaborative conversation.

Tear: Inspired by paper, this strategy handles a conflict by two users over a single document by breaking the document into two pieces. This might encourage the pair to negotiate before reassembling the document so that work can continue.

Rank: A higher-ranking user can always take documents from or select from the menus of lower ranking users.

Speed, Force: These two policies are examples of policies that use a physical measurement (the speed with which each user pulls on the document, or the pressure each user applies to the document) to determine who is the "winner" of a contested item.

Sharing: "Sharing" allows users to dynamically transition an element between the "public" and "private" policies. To support sharing, we have explored four interaction techniques – *release, relocate, reorient,* and *resize*, which are described in [6].

Explicit: When using this policy, a document's owner retains explicit control over which other users can access that document. For example, the owner can grant and revoke manipulation or write permissions on the fly by interacting with tabs on the edge of the document that toggle the permissions for individual users.

Dialog: This policy offers standard WIMP semantics, responding to an attempt to "steal" a document by prompting the document's owner to allow or forbid the action via a popup dialog box.

3. APPLICATION SCENARIOS

The policies we have presented could be used individually or in combination depending on the context of the application. The following scenarios illustrate table applications that would be well-served by each of our proposed coordination policies.

3.1 Global Scenarios

A photo-browsing application which provides options such as clearing all open photos off the table, or re-orienting all open photos to face a certain user: Policies such as "no holding documents" or "no selected documents" would be suitable here, because it is probably not appropriate to remove or reorient a photo that is actively being inspected by another user.

An educational application that lets a group of students explore a topic and then answer questions on it: "Voting" might be desirable here to ensure that all the students are finished with the current topic and are ready to move on to the next part of the assignment. This policy is especially useful with larger tables and/or larger groups of people where it is harder to explicitly coordinate such actions.

A business productivity application used for interactive presentations: Here, the "rank" setting might be useful – the presenter would have a higher rank than the participants so that the participants would be able to interact with appropriate parts of the presented material but would not be able to alter any key settings or accidentally terminate the presentation.

A competitive tabletop game: A policy such as "privileged objects," which requires possession of a special object in order to make a global change, might be appropriate for a competitive game in which players earn use of the special object through gameplay. The "anytime" policy might also be suitable for a game since disrupting the other players with surprising state changes might be advantageous in competitive situations.

3.2 Whole-Element Scenarios

A walk-up-and-use table in a museum exhibit: Since none of the table elements belong to any of the users, controlling access may not be desirable. The "public" policy would be sufficient here.

A large public table in a library, where several strangers work on individual projects in parallel: The "private" policy might be appropriate, since the individuals are not working collaboratively, but are simply sharing the resource of the electronic table. Moreover, they are each working on their own information, and may not trust users they do not know with permission to access it.

A brainstorming session where several people want to illustrate their own ideas for the best way to modify a design diagram: The "duplicate" policy would facilitate this situation by allowing each person who grabs the diagram to receive their own copy of it, which they could use to illustrate their idea without taking the diagram away from a colleague. Or, they could use the "personalized views" policy – instead of overwriting a colleague's work, each user's individual annotations would be displayed on the diagram only when that user is touching it.

Several children playing an educational game meant to emphasize cooperative skills: This application might benefit from the "stalemate" conflict-resolution policy, to emphasize that when users fight over an object, nobody wins. Similarly, the "tear" policy might force cooperation in this scenario.

A table used in a classroom by a teacher and her students: The "rank" policy might be useful, so that a student cannot manipulate the teacher's documents, but the reverse is permitted.

A group of teenagers playing a competitive tabletop video game: Policies like "speed" and "force" would test the users' skills and reaction time, and add an interesting dimension to gameplay.

A group working on a joint project – there is some collective discussion and work, as well as periods where group members individually edit items: "Sharing" is appropriate in this situation, as this policy would allow group members to transition their documents between public and private modes for times of group work and times of individual modifications. The "explicit" policy is a viable alternative – a user could choose which collaborators to grant access to during the times of group work, and could revoke those permissions when they are no longer applicable.

A group working around a very large table: When an event, such as one user manipulating a document owned by somebody else, occurs outside of the document owner's field of view, the "dialog" policy might be appropriate, since it creates the permission-granting dialogue box near the user's focus of attention, thus increasing their awareness of relevant events occurring at the other end of the table.

4. CONCLUSION

Coordination mechanisms beyond social protocols can play an active role in co-located groupware – besides preventing conflicts that may arise due to confusion or malicious intent, such policies help ensure that software has deterministic, predictable responses to multi-user interactions. We have observed that social protocols do not always suffice in relatively simple situations, and we suspect that the need for coordination may increase as the number of users, the number of documents, or the size of the surface increases (although an increase in table size could reduce the number of whole-element conflicts by making it more difficult to reach other users' documents, it could potentially increase the number of global conflicts by making it more difficult to monitor and coordinate the activities of other users).

We have introduced a framework for discussing multi-user coordination in terms of conflict type (global, whole-element, or sub-element) and initiative (proactive, reactive, or mixed-initiative). We have also proposed a set of coordination policies, and presented motivating scenarios for their use.

As a next step we will explore the necessary toolkit-level support that can facilitate the use of these coordination policies in SDG applications, and we will utilize a combination of policies in various applications we are developing and studying. Additionally, we intend to examine the applicability of this framework beyond tabletops, to other SDG form-factors. We have presented an initial set of policies to address the multi-user coordination difficulties introduced by shared display groupware; however, the potential space of coordination policies is large – we also hope to expand our taxonomy of coordination strategies to reflect new insights we gain as we continue our exploration of multi-user coordination and the challenges it presents.

5. REFERENCES

[1] Bier, E. and Freeman, S. MMM: A User Interface Architecture for Shared Editors on a Single Screen. *UIST 1991,* 79-86.

[2] Dietz, P. and Leigh, D. DiamondTouch: A Multi-User Touch Technology. *UIST 2001*, 219-226.

[3] Greenberg, S. and Marwood, D. Real Time Groupware as a Distributed System: Concurrency control and its Effect on the Interface. *CSCW 1994*, 207-217.

[4] Izadi, S, Brignull, H., Rodden, T., Rogers, Y., and Underwood, M. Dynamo: A Public Interactive Surface Supporting the Cooperative Sharing and Exchange of Media. *UIST 2003*, 159-168.

[5] Rekimoto, J. SmartSkin: An Infrastructure for Freehand Manipulation on Interactive Surfaces. *CHI 2001*, 113-120.

[6] Ringel, M., Ryall, K., Shen, C., Forlines, C., and Vernier, F. Release, Relocate, Reorient, Resize: Fluid Techniques for Document Sharing on Multi-User Interactive Tables. *CHI 2004 Extended Abstracts*, 1441-1444.

[7] Scott, S.D., Grant, K., and Mandryk, R. System Guidelines for Co-located Collaborative Work on a Tabletop Display. *ECSCW 2003*, 159-178.

[8] Smith, R., Hixon, R., and Horan, B. Supporting Flexible Roles in a Shared Space. *CSCW 1998*, 197-206.

[9] Shen. C., Vernier, F., Forlines, C., and Ringel, M. DiamondSpin: An Extensible Toolkit for Around-the-Table Interaction. *CHI 2004*, 167-174.

[10] Shen, H. and Dewan, P. Access Control for Collaborative Environments. *CSCW 1992*, 51-58.

[11] SMART Technologies. "Digital Vision Touch Technology." White Paper, February 2003. http://www.smarttech.com/dvit/DViT_white_paper.pdf.

[12] Stewart, J., Bederson, B., and Druin, A. Single Display Groupware: A Model for Co-present Collaboration. *CHI 1999*, 286-293.