

GrooveUp
Step up to the next Groove

Yvonne Earnshaw
Christopher Williams
Computer Supported Cooperative Work – CS 5734
Virginia Tech, Blacksburg, VA 24061

May 10, 2004
Blacksburg, Virginia

Contents

- 1 Overview** **4**
- 1.1 The Issues 4
- 1.2 Redesign 5

- 2 Re-Designing Groove Through Scenarios** **6**
- 2.1 GrooveUp Scenarios 6
- 2.1.1 Scenario 1: Document Co-Author/Co-Edit 6
- 2.1.2 Scenario 2: Where’s my Group? 8
- 2.1.3 Scenario 3: Task Selection 10
- 2.1.4 Scenario 5: Wrong Meeting 13

- 3 Conclusions** **14**

List of Figures

2.1	Prototype of the revised Co-Author/Co-Edit interface.	7
2.2	Prototype of the revised Add Project interface.	9
2.3	Prototype of the revised Task Selection interface.	10
2.4	Prototype of the revised Project Selection interface.	12

Chapter 1

Overview

This design project is derived from the findings from a preliminary usability study conducted on Groove, an Internet-based tool that supports online and offline workgroup collaboration. The experimenters conducted an in-lab experiment with three groups of people (4 people in each group) collaborating to perform predefined tasks. Both usability and computer-supported cooperative work issues were taken into account by examining the conversation logs and pre- and post-experiment questionnaires completed by the participants. Based upon those findings and CSCW theories, we have redesigned the Groove interface and included several new functionality features for a next generation version – GrooveUp.

1.1 The Issues

In order to properly address the current usability issues within the Groove interface, two perspectives were probed, one from an administrator's perspective and the other from a generic user's perspective. In this manner, issues that mar the interface, and the potential repercussions of adjusting the interface, will be identified from the two main audiences of the application. The perspectives are drawn from tacit knowledge of the system gained through usability studies conducted for [2].

From the administrator's perspective, setting up of workgroups was an issue, due to the necessity of having to add users or invite them to the group. Another issue was that if one person made a change to the layout of the system by adding a tool, the change was delivered to all users and not just on the individual users Groove. This latter issue caused chaos for the administrator whose responsibility is to provide an organize and efficient workspace for the users.

On the opposite side, the user's perspective, two main usability issues were identified specifically relating to the tasks of "editing and chatting" [2]. With respect to the former of these two issues, the editing issues were mostly with respect to determining who had made changes at a particular point. In practice, users of the system may perform a task like this asynchronously, addressing this confusion by simply posting a message when they have made their changes to the document. Within Groove, the current document co-edit tool only provides an indication as to which group member currently has control over the document, similar to the degree of information that a filesystem would provide. Unfortunately, the Groove file manager does not provide this information, differing it from the prevalent design technique employed for a file manager, thus the co-edit tool is the only tool that provides a control mechanism.

With respect to the other issue, chatting, several participants mentioned that they were unsure which of the group members were involved in the current chat. Upon further inspection, it was noticed that there was at least five (5) different implementation of communications that utilize the banner of "chat" within in the Groove interface. The overloading of this term within the interface leads to a high degree of ambiguity and complexity that the user has to decipher every time they may want to communicate something to another

user, the group, or to establish as an artifact.

1.2 Redesign

For the purposes of this interface redesign, these two perspectives, administrator and user, were analyzed in greater detail using a scenario based design format as defined by Rosson and Carroll in [6, 7]. Scenario based design was employed to identify and address the root problems, without requiring an entire new set of users and batch of usability studies. The rest of this document presents the series of scenarios describe a new interface and some new functionality that were derived from analysis of the aforementioned usability issues. This document concludes with a summary of the work accomplish and also evaluates our proposed alterations against the next production version of Groove as designed by Groove Networks.

Chapter 2

Re-Designing Groove Through Scenarios

From the usability and CSCW issues that were raised in the aforementioned usability study, five scenarios were created according to the examples in [7]. These scenarios covered the three identified usability areas with the current Groove interface - administration, editing, and chatting. These scenarios are presented in a manner that exposes our actors utilizing a re-designed Groove application, termed GrooveUp for this document. From these scenarios, a design rationale is provided that explicitly discusses the improvements and reasons for them. All of the re-designs presented here focus strictly on the task of collaborative work, thus abstracting this discussion from HCI issues that may exist within the current Groove interface (Note: some of the HCI issues will be resolved in the process, but that is not the focus of this document).

2.1 GrooveUp Scenarios

2.1.1 Scenario 1: Document Co-Author/Co-Edit

Mary, Jane, Dick, and Tom are required to develop a design document for a project they have all four designed and developed. The design document is to include an introduction, description of the entire system, description of each piece, and finally, a conclusion. Each member was "lead" for exactly one of four pieces of the project, as well as assisting in some way to the construction of the other pieces.

The group has decided to develop the document in a manner where each person would first author the description of the piece that they were "lead" for, and act as reviewers for all of the other pieces that they assisted in development. Mary was designated to provide the introduction and conclusion sections, but the other three members have offered their support for editing and review. Dick is responsible for providing the system overview since his piece of the project is essentially the core component, thus providing him with valuable insight into the workflow of the entire project as a whole.

Also, Mary, Jane, Dick and Tom will be unable to begin working on this document until the night before it is due, so in this scenario efficiency of authoring, reviewing, and editing process needs to be at a maximum. They all go to their respective homes and each opens their GrooveUp application. Mary initiates a Document Authoring session by clicking on the "Author Document" Groove Dot, shown in Figure 2.3. This opens an interface similar to 2.1 which Mary quickly recognizes as being like Microsoft Word that she uses daily. She also notices that on the left is a white box with just her name and that behind her name is a purple background. Interestingly enough, when she begins to type in the document, behind the text is purple too. Mary quickly identifies the connection between the two is that anything she has added to the document will have her color background.

Immediately acquiesced with the working environment due to her tacit knowledge, Mary begins to author the content piece she is responsible for. However, her end product is the result of not just her work,

but the combined work of the entire group. Therefore, to accomplish the task of authoring, reviewing, and editing in the limited time constraints that they have, Mary notices, and clicks, the invite button and identifies the members of her project that she would like make co-authors for this document. As the members begin to co-author the document, Mary notices that they each have their own color, identifying each users contribution to the document.

Design Rationale

The current Groove application facilitates co-authoring/co-editing of document through its shared repository. This facilitation, however, is relatively remedial for collaborative work in that Groove only allows one person to edit the document at any given time. Groove doesn't actually handle document editing, though it lists it as a feature. Groove currently hands off the task to Microsoft Word, which has zero real-time, synchronous collaborative capabilities (note: it does offer track changes for asynchronous collaboration). This was done in order to utilize user familiarity with the Microsoft Office suite of tools, a decision which has the added burden of limiting its collaborative abilities. In this sense, Groove slightly improves the collaborative work abilities by providing a single instance of the document. However, the strict deadline that is applied in the scenario draws out the flaw in Groove in that multiple users cannot concurrently manipulate the document, thus requiring that the document be modified in succession.

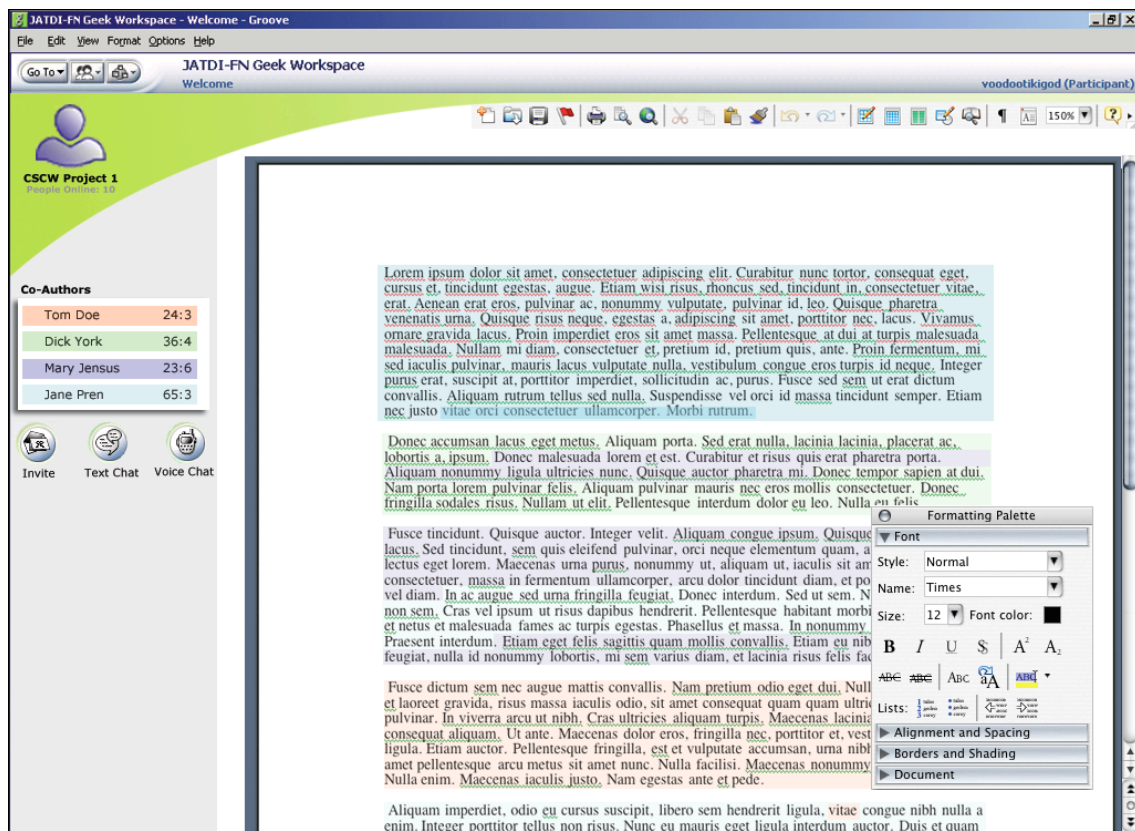


Figure 2.1: Prototype of the revised Co-Author/Co-Edit interface.

Our proposed solution, as exemplified in the scenario, integrates a Microsoft Word-like interface in order to leverage previous knowledge of the user. In this manner, collaborative authoring can be supported in an enhanced manner beyond normal face to face authoring through synchronous real-time authoring

and editing. The new interface would provide a means for multiple users to be invited to contribute to the document throughout its life-span and not just at the beginning of the authoring session. This will allow each of the users to author, review, and revise the document in their own particular manner and concurrently update the document without concern of “overwrite” errors. However, this simultaneous authoring may lead to problems of determining who has added, reviewed, or removed which piece. The resolution for this issue is that each user participating in the co-author/co-edit is assigned a color, which will background every change or addition that he/she makes.

If the addition or change is made to a section that another member has authored, the original author is identified through a notification icon. Upon pressing this notification, the original author can review the change and either accept or decline it. Users should be able to allow all changes without review in order to improve efficiency. The ability to review and revise the provided content and content changes of all users as it is being constructed assist in developing a common ground and confidence amongst the contributing authors [1, 5], which in this situation would give the document a cohesive flow, despite being developed by numerous, different minds.

This concurrent authoring and editing of the document allows the users to dynamically switch between roles within the social activity of document development. By identifying each of the users with a color, the concept of individual contribution is retained from analog or non-concurrent document editing.

With respect to the original scenario, the four group members can rapidly develop their design document and review it at the same time. Since each user is authoring the same document, it can drastically increase the efficiency of the collaboration in comparison to analog solutions and the current Groove implementation.

2.1.2 Scenario 2: Where’s my Group?

Joe is a new team member at the XYZ Corporation. He has just been promoted to a manager position that oversees several groups. Each group has its own workspace that Joe needs to join. Each workspace may include document repositories, agendas, and meeting schedules, all of which Joe will need access to and control over quite frequently. He is particularly interested in the work that Mary, Jane, Dick, and Tom have been conducting lately and needs to gain access to the group. He has attempted to get in communication with Mary to request his inclusion to the group. She has not been able to give him access to the group, since she is not a group manager. Mary thinks Dick is the manager, so Joe attempts to contact Dick, but alas Dick is unreachable. In this case, Joe would be able to search for Dick York, his project leader (as shown in Figure 2.2) or he could search for other group members. To prevent unauthorized people from joining the group, perhaps an additional password could be required.

Design Rationale

The current Groove system requires that a user be invited to join a group by the person who is the group manager. This can cause quite a delay for the person needed to join the group as well as a lot of confusion, just as Joe experienced in scenario 2. Groove distinguishes between three roles: manager, participant, and guest. A manager has all permissions, a participant can add a new tool, and a guest has read-only access. However in order to be given a manager role for the group, you first be a member or invitee to the workgroup, of which Joe would not be in this scenario. Unfortunately, Joe has to still contact Dick in order to gain access to the Groove workgroup, thus all of the same issues exemplified in the scenario remain, despite the utilization of Groove.

The problem is that one person has control over the entire group and an outside person is not able to join a group without being invited. Although this is most likely used for privacy and security reasons, we find it is a hindrance. We pose that a solution for this scenario would be to allow a person to search for a group or

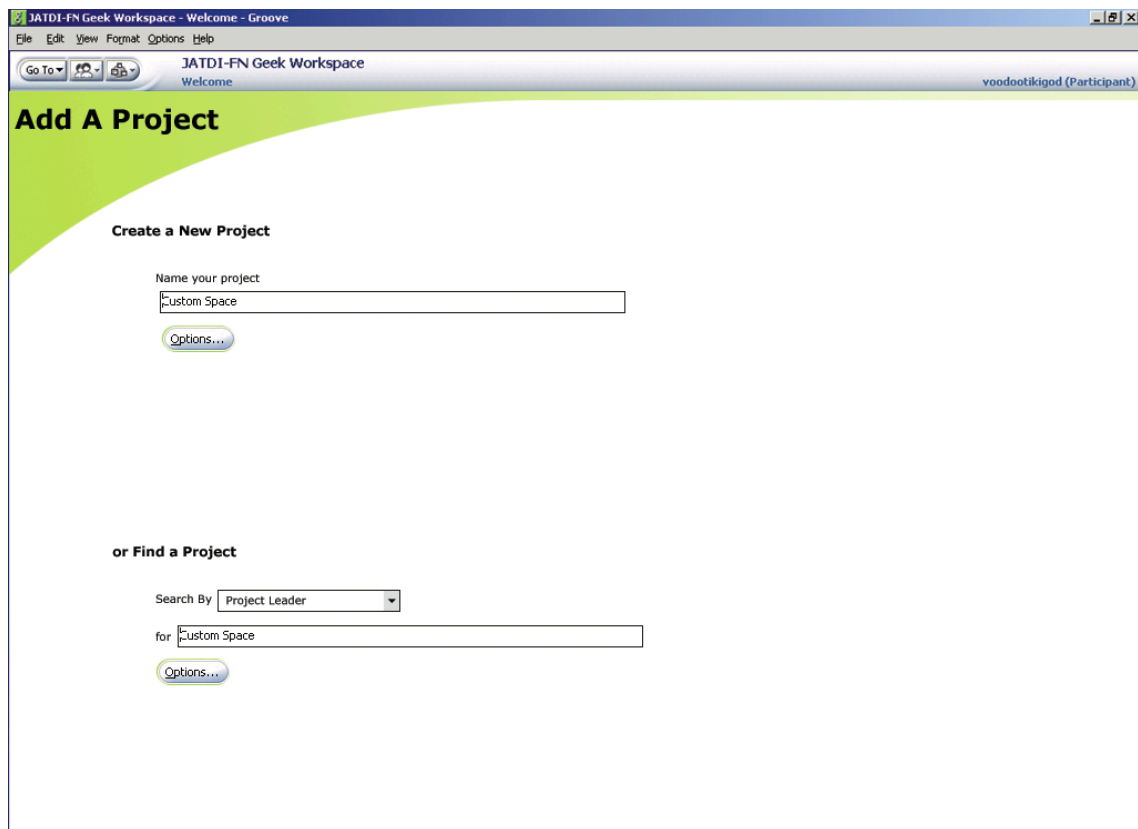


Figure 2.2: Prototype of the revised Add Project interface.

group members and be able to join the group without having to be invited (which may cause a long delay). Perhaps an additional pass code could be required to prevent unauthorized people from joining the group.

2.1.3 Scenario 3: Task Selection

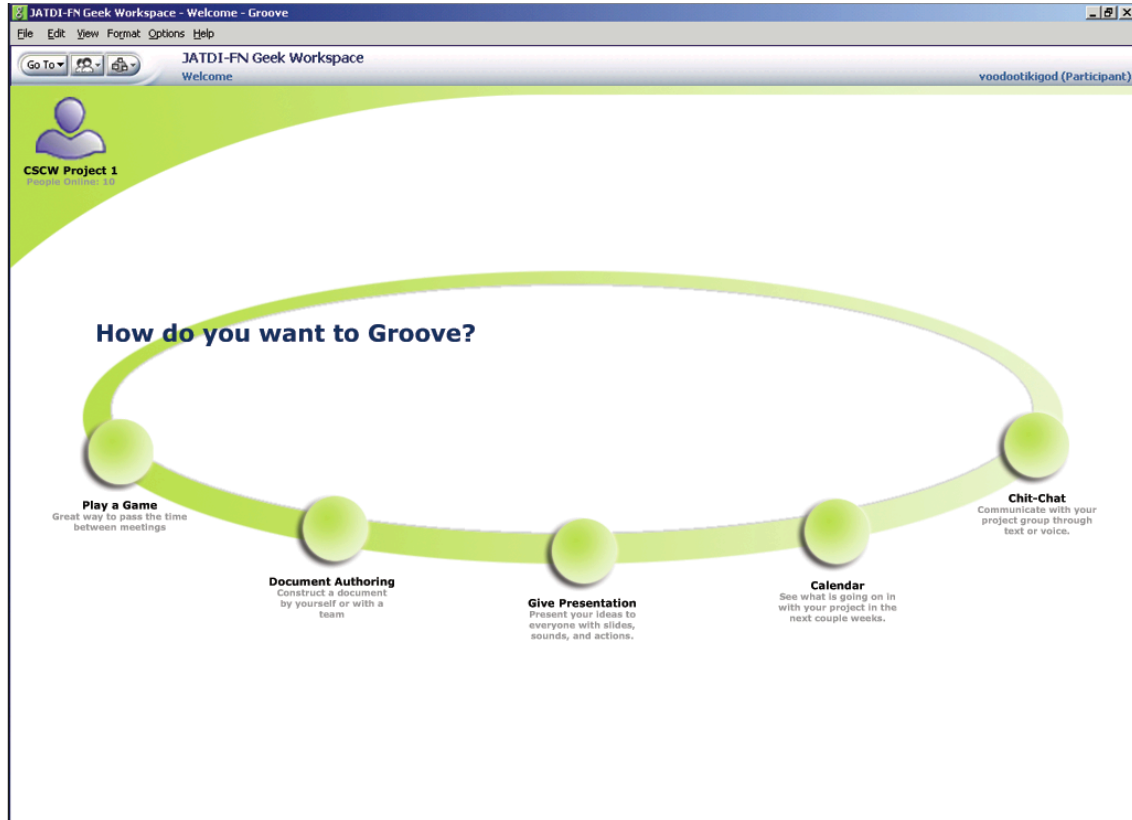


Figure 2.3: Prototype of the revised Task Selection interface.

Mary wants to present some a slide show to her group about a new technology that is pertinent to their project. The slide show contains a high degree of technical information that she believes will need to be explained vocally. Mary's main intent of presenting this information is so that Dick will become familiar with it as he will have to develop a project utilizing it. Unfortunately, Dick is out of the area for the next couple weeks on a business trip. Since Dick has many lonely hotel nights for the next couple weeks, he would like for Mary to present the information to him as soon as possible.

Mary starts up her GrooveUp application and is presented with the interface shown in Figure 2.3. She reviews each of the dots and determines the one that best describes the task that she is attempting to accomplish. Finding the "Give Presentation" Groove Dot, she immediately recognizes that clicking the dot will initiate a presentation session. She is then presented with a selection panel to enumerate the users that she would like to give the presentation to, in this case she only selects Dick as a viewer.

Mary then has to select the presentation that she wants to give using the familiar PowerPoint interface. Since it is just like PowerPoint from this point on, Mary knows exactly what to do to give Dick all of the information in an organized and structured manner.

Design Rationale

The flexible nature of the Groove application offers the ability to accomplish a high degree of tasks despite geographic distance. The current Groove application provides all of the basic units to accomplish these tasks, but does not provide the user with any direction or unity as to how to accomplish the task using those units. Upon entry to the Groove application, there is little to no affordance for the "next step" that the user might want to take. This is to some degree a usability issue, but since multiple users will be confused as to where accomplish a task (such as view/show a presentation), it decidedly falls under the domain of a CSCW issue as well.

This problem is rooted in the bottom up approach that Groove has adopted for its user interface. In providing all of the basic units, it facilitates an enormous degree of user flexibility at the cost of user confusion. In the current user interface, Groove places the processing work for determining the next step on the user, assuming that the user completely understands the scope and ability of the application. This assumption leads to a very complex and seemingly disjoint interface that even advanced users take a large degree of time to become familiar with.

To resolve this issue, we propose a top down interface for the system, a prototype of which is expressed in figures 2.4 and 2.3. In this new interface, a user would be presented with an end goal (presentation, write document, communicate, etc.). Upon selecting a goal, users would be "walked" through the necessary pieces to accomplish that goal. This new interface puts more weight on reducing user confusion than flexibility than the previous; a design decision that draws on the findings of [8] in which users' rarely to never utilize the flexibility that is inherent in an application, choosing rather to utilize default settings.

Scenario 4: What project should I work on?

Jane is attempting to coordinate four large scale projects that do not overlap. Two of the projects pertain to her academic career (CSCW Project 1 and Virtual Environments Project) and two of them pertain to clients she is working with outside of school (ACME JAV and Jaguar). She has work to do with all of the project, but needs Dick to complete her work in the ACME JAV project. Looking at the interface pictured in Figure 2.4, she notices that beneath ACME JAV a notification that zero people are online in that project. Seeing this, Jane decides to work on her CSCW project, of which ten people in her fifteen person group are online at this point.

Design Rationale

There are several design alterations drawn out by this and the previous scenario. In the previous scenario, it was assumed that Mary only has one project she is working on. From this assumption, the application conjecture that it would impede her workflow process to force her to select the only project she is working on, like the current Groove application does. This reduction of visible functionality hides the complexity of the system from Mary, allowing her to focus on her project.

Jane represents a different style of user in that she participates in multiple diverse groups. Instead of present Jane with a simple listing, as done in the current Groove application, the new interface provides her with two improvements for her work flow. The first is strictly a usability improvement in that all of the clutter and issues associated with entering the project are abstracted to the point where Jane only needs to press the project she wants to work on. The second represents a more CSCW issue in that from this new interface, Jane immediately has workspace awareness [3, 4] about all of her projects as presented through a non-invasive notification system. As expressed in the scenario, workspace awareness was critical for assisting Jane in determining which project to work. By integrating the concept of workspace awareness more so into the GrooveUp application, we can decidedly increase productivity for users with multiple, concurrent projects.

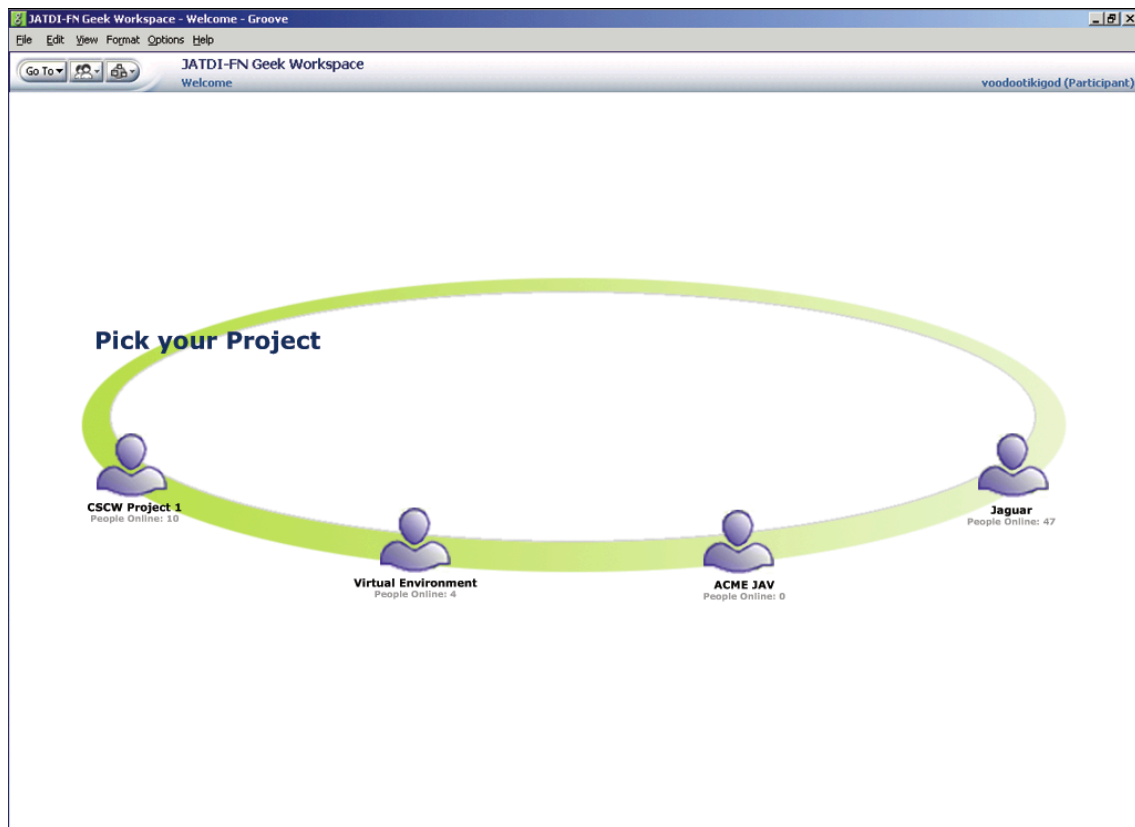


Figure 2.4: Prototype of the revised Project Selection interface.

2.1.4 Scenario 5: Wrong Meeting

Mary, Jane, Dick, and Tom are all at separate sites in various parts of the country, although they have all decided to meet for a teleconference to discuss a new project. They have set up a specific time to meet (3:30 p.m. EST) and utilize a single telephone number to dial-in to along with a pass code that connects them specifically to their meeting. Mary dials in at 3:27 p.m. She correctly types in her pass code and has to wait on the line until the rest of the group dials in (costing the company or her money for dead air).

Dick dials in at 3:28 p.m and unfortunately mistypes the pass code, which takes him to an empty room, unbeknownst to him it is the wrong meeting. Jane and Tom dial in at 3:30 pm EST and connect to the appropriate meeting room. They question each other where Dick was, but continue the meeting without him. Dick never receives the information until the end of the day, when Jane calls him to inquire if he is alright. Unfortunately, since there is no artifact of the teleconference, Jane attempts to relay the entire meeting to Dick so he has a grasp of what is going on within the company.

Our solution can be found in Figure 2.1. The chat window at the bottom of the screen has been hidden from the user. Instead, icons for “Text Chat” and “Voice Chat” are very visible to the user. When the user clicks on the “Text Chat” button, the chat window becomes visible. It is important to note that common ground has been established prior to the discussion. This new interface would remove the confusion of the user from having multiple options of chat, which dont all go to the same chat area.

Design Rationale

Groove has multiple tools for synchronously and asynchronously chatting. All of these tools, except the voice chat, provide a historical artifact, the transcript, that can be retrieved and reviewed at a later date. Unfortunately, within Groove, two of the tools are both named “chat”, although they offer completely different functionality. As one might conjecture this will lead to a lot of confusion for users, as identified in [2].

There are a few different approaches to solving this problem. The chat that may be shown or hidden could be completely removed or named something else. The intended use is to post announcements to other users who will see the message when logging onto Groove. An announcement from Dick could read something like I updated the schedule spreadsheet with Toms changes. Perhaps this window could only appear when the users logs into the system.

Another approach is to make the chat feature more obvious. The current system requires the user to click on the Contacts tab. To resolve this issue, a Chat tab could be created or to make the chat icon directly link to a chat window.

This new interface would remove the confusion of the user from having multiple options of chat, which dont all go to the same chat area.

Chapter 3

Conclusions

The employment of a top down approach represents a drastically different and more user-centric perspective for the application as a whole. As apparent through the four pictured interfaces, there is a reduction in the visible options available to the user, thus reducing clutter and distraction. Through GrooveUp's workgroup awareness notifications, users can more effectively work on multiple projects in multiple workgroups, thus increasing overall productivity of the workgroup. Finally, our revised co-author/co-edit document capabilities drastically improve the workflow necessary for collaborative document construction. The document co-author/co-edit functionality was designed using sociological and CSCW theories to replicate, and in-turn enhance, the face-to-face document co-authoring process.

In this document, we have presented a revised interface in GrooveUp for the Groove application utilizing a combination of the findings from a preliminary usability study conducted on Groove and CSCW theories. The GrooveUp interface features new functionality that is present within the current system, but is not currently made visible such as the workgroup awareness information.

Bibliography

- [1] H. H. Clark and S. E. Brennan. Grounding in communication. In *L. B. Resnick & J. Levine & S. D. Behrens (Eds.), Perspectives on Socially Shared Cognition*, pages 127 – 149, 1991.
- [2] Y. C. Earnshaw, S. Krishnamoorthy, V. Nakra, M.E. Pinney, and R. St. Clair. Is it groovy: A usability and cscw study on groove. Technical report, Virginia Tech, 2004. Unpublished document.
- [3] C. Gutwin and S. Greenberg. Workspace Awareness for Groupware. In *Conference Companion on Human Factors in Computing Systems*, pages 208–209. ACM Press, 1996.
- [4] C. Gutwin, M. Roseman, and S. Greenberg. A Usability Study of Awareness Widgets in a Shared Workspace Groupware System. In *Proceedings of the 1996 ACM conference on Computer supported cooperative work*, pages 258–267. ACM Press, 1996.
- [5] A. Monk. Common ground in electronically mediated communication: Clark’s theory of language use. Technical report, University of York, UK, 2003.
- [6] M. B. Rosson and J. M. Carroll. Scenario-Based Usability Engineering. In *Proceedings of the conference on Designing interactive systems*, pages 413–413. ACM Press, 2002.
- [7] M. B. Rosson and J. M. Carroll. *Usability Engineering: Scenario-Based Development of Human-Computer Interaction*. Morgan Kaufmann, 2002.
- [8] Lucy A. Suchman. Office procedure as practical action: Models of work and system design. *ACM Trans. Inf. Syst.*, 1(4):320–328, 1983.