From Scatterbrained to Focused: UI Support for Today's Crazed Information Worker

Mary Czerwinski

Microsoft Research
One Microsoft Way
Redmond, WA 98052, USA
marycz@microsoft.com

ABSTRACT
Today's information workers are characterized by their ability to easily handle interruptions, multi-task, switch tasks quickly, and make sense of enormous amounts of information in high-pressure situations. Current and future technologies, including various wearables and sensing devices, ensure that robust communications and information transmissions can occur almost anywhere, any time. Our ability to log, collect, and visualize event data has become more sophisticated, allowing us to analyze trends and identify patterns across many areas of individual and group behaviors. How do we use these technological trends to ensure that we are designing tools that improve productivity, insight, and an overall sense of user control? In this paper, I discuss our research approach to the user-centered design of advanced user interfaces, and describe a few of our research projects.

ACM Classification Keywords
H5.m Information interfaces and presentation (e.g., HCI): Miscellaneous.

INTRODUCTION
Information workers often balance many interruptions and tasks. In fact, a recent study found that information workers kept an average of 10 "working spheres," or basic units of work, active at once [3]. This is not surprising given the many studies on task switching showing that users spend as little as 3 minutes (on average) on each task before switching, and get interrupted at least once per task [1,3,8].

These studies point to several problems multitaskers continue to battle due to inadequate software support. First, interruptions plague longer-term tasks undermining user concentration and task progress, because users are often unable to determine which interruptions need to be handled immediately [1]. This makes it difficult for users to *maintain current task flow*. Second, successful task completion requires knowing when to step out of the current task and return to a paused task, which we call *resumption timing*. For example, Czerwinski *et al.* [1] found that people often set aside tasks while waiting for some external event (*e.g.*, an email to arrive from a co-worker), but wanted to resume as soon as the event occurred. Third, people have trouble getting back on task after shifting their attention away (*i.e.,* it is difficult to *reacquire tasks*).

We believe that providing relevant task information in a glanceable, low-attention manner is critical to solving these multitasking problems. However, little is known about *what* information is needed and how to provide it to best improve a user's ability to maintain current task flow, manage task resumption, and reacquire tasks in multitasking situations. In this paper, we explore designs in peripheral interfaces, studying the performance effects when these interfaces convey varying types of task information.

TASK MANAGEMENT
While working on their high-level tasks, users need easy access to particular sets of windows and applications that contain relevant information. Hence, we assert that an effective task management system should provide easy mechanisms for users to group relevant sets of windows, to organize the groups and windows within the groups, to switch between groups, as well as to lay out how the groups and windows are displayed on the screen. Effective multi-window task management is critical to a successful user experience on large displays.

In this section, we present two systems that explore different facets of task management for large displays: *GroupBar* and *Scalable Fabric*. GroupBar adds new semantics to the existing Microsoft Windows Taskbar for organizing and managing tasks. Scalable Fabric uses scaling and a focus-in-context metaphor to visualize groups of related windows. In this system, all tasks are scaled and located in the periphery so that they are simultaneously visible.

GroupBar

We designed GroupBar [13] with the goal of providing task management features by extending the current Windows Taskbar metaphor. GroupBar preserves basic Taskbar tile functionality, presenting one tile for each open window in the system, and showing the currently 'active' window tile in a darker, depressed-button state. Users can click on any tile to activate the corresponding window or to minimize the window if it is already active. Going beyond current Taskbar functionality to offer task management support, GroupBar allows users to drag and drop tiles that represent open windows into high-level tasks called 'Groups', represented by the green group tab shown in Figure 1. This group control allows users to switch between tasks with a single mouse click and perform window operations (minimize, maximize, close) on the entire group at once.

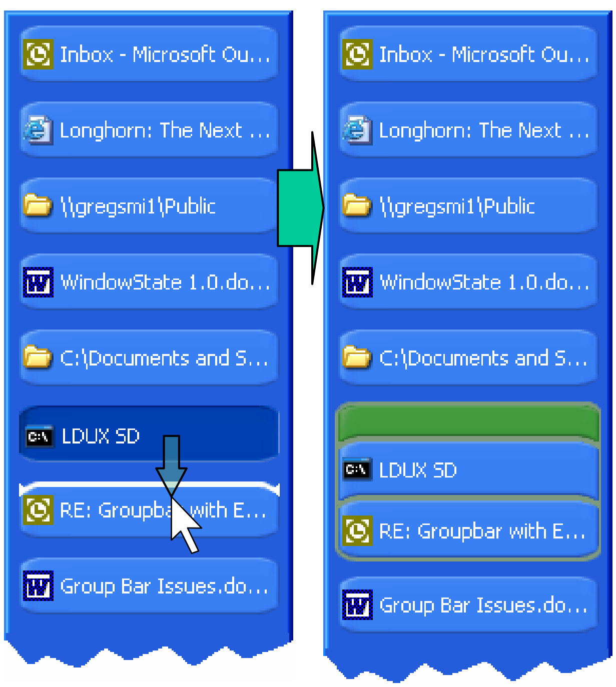

Figure 1: Dragging a window tile onto another tile combines both into a "Group".

With GroupBar we wanted to allow users to group and regroup windows easily and quickly, and then allow them to operate on groups of windows (or tasks) as though they were a single unit. We thought that by incorporating a wider array of spatial arrangement preferences, offering users a higher-level organizational structure (the group), and extending existing window manipulation functions to the group level, we could design an improved window management experience that built upon the existing Taskbar metaphor.

Scalable Fabric

Scalable Fabric [12] is a system based on managing multi-window "tasks" on the Windows desktop, this time using a focus-plus-context display to allocate screen real estate in accordance with users' attention. Scalable Fabric allows users to leave windows and clusters of windows open and visible at all times via a process of scaling down and moving the windows to the periphery. Scalable Fabric is a focus-plus-context display in the sense that users focusing their attention on a primary task are provided with the context of other work (i.e., competing or potentially related tasks) displayed in their periphery. This use of the periphery leverages both the user's spatial memory and also their visual recognition memory for images in order to facilitate task recognition and location. Our design was influenced by others working in the area of task management and large displays [4, 5, 7, 9].

In Scalable Fabric, the user defines a central focus area on the display surface by moving periphery boundary markers to desired locations (see Figure 2). The user's choice of focus area location and size is influenced by the configuration and capabilities of the physical displays. For example, on a triplemon display, users may prefer to define the central monitor as the focus area, having no upper or lower peripheral regions and using the side monitors as the only peripheral regions.

Figure 2. Scalable Fabric showing three task clusters in the periphery.

Within the focus area, windows behave as they normally do on the Windows desktop. The periphery contains windows and clusters of windows, or tasks, which are not currently in use, but may be put to use at any time. Windows in the periphery are smaller so that more tasks can be held there when the user is focusing on something else. With this metaphor, we believe users will rarely need to close or minimize windows in the traditional sense. Users can take advantage of extra screen real estate to allow the peripheral windows to always be visible.

Figure 3: Clipping Lists interface showing our study's Quiz task with a progress bar indicating the next quiz module is loading, and Web page text to help users recognize them.

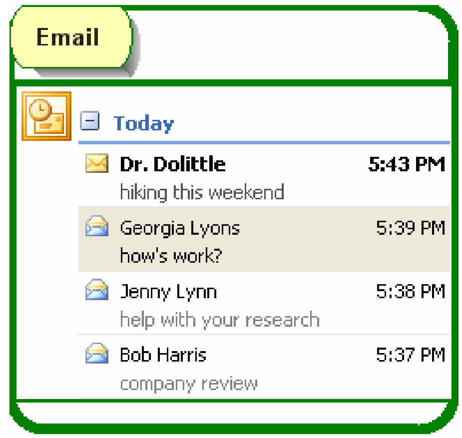

Figure 4: Clipping Lists with Borders shows an email inbox and indicates with a green border that new email arrived.

When a user moves a window into the periphery, it shrinks monotonically with the distance from the focus-periphery boundary, getting smaller as it nears the edge of the screen. When the user clicks on a window in the periphery, it returns to its last focus position; this is the new 'restore' behavior, and is accomplished with an animation of the window moving from one location to the other. When the user 'minimizes' a window in the focus area by clicking the window's 'minimize' button, it returns to its last peripheral position.

Clipping lists and change borders

Scalable Fabric was designed to be a truly peripheral display for very large display surfaces, like walls. Peripheral displays convey information in a low-attention way outside of a focal task. They often employ *abstraction* to reduce the fidelity of raw information so that is easier to read at a glance. Abstraction is a design mechanism for reducing the amount of information shown and/or transforming the information to a different form. Although several researchers have argued that abstraction improves glanceability and enables people to more quickly understand information [10,11], existing research has not established strong guidelines for *what types* of abstraction lead to easier understanding without losing too much information. In our work we explore this issue by examining two types of abstraction, semantic content extraction and change detection, in multitasking situations. We decided to explore these issues using alterations to the Scalable Fabric interace.

Semantic content extraction refers to selecting and displaying only pieces of information from an original window that are relevant to task flow, resumption timing, and task reacquisition. In our interfaces, we show in the periphery lists of rectangular window "clippings," which we call *Clipping Lists* (see Figure). Change detection provides one extra bit of information to the user, signaling when a change has occurred within a window. In our interfaces, we use *Change Borders*, colored highlights that appear around peripheral windows, to indicate changed content (see Figure). We conducted a lab study comparing four interfaces that introduced these abstraction types to the Scalable Fabric task management system [12]. In performing these comparisons, we were not interested in which form of abstraction shows *more* information, but rather which shows the most *relevant task* information in a glanceable way. We showed that Clipping Lists were significantly better at allowing a user to maintain task focus and switch when most optimal, over Scalable Fabric as a baseline. In addition, adding the Change Borders to Clipping Lists was not only most effective, but also most preferred by users.

CONCLUSION

We set out to improve multitasking efficiency, focusing on helping users maintain task flow, know when to resume tasks, and more easily reacquire tasks. We presented 3 user interfaces—groupbar, Scalable Fabric and Clipping Lists with Change Borders—all designed to help with multitasking and peripheral task awareness. All of these user interfaces were examined empirically against existing tools and practices for multitasking and information work.

The main contribution of these results is that we showed that peripheral grouping of tasks contexts and semantic content extraction is more effective than current practices in improving multitasking efficiency (e.g., over using a simple taskbar as in most windows environments today). We also showed that, with Clipping Lists, semantic content extraction significantly benefits task flow, resumption timing, and reacquisition. These findings provide a better understanding of how to design peripheral displays that aid people who multitask, so that they focus their cognitive resources on the task at hand, instead of on task and windows management.

ACKNOWLEDGMENTS
Many of the designs and ideas here are attributable to the VIBE team members of MSR, and Tara Matthews, who designed the Clipping Lists and Change Border additions to Scalable Fabric while interning with our research team.

REFERENCES

1. Czerwinski, M., Horvitz, E., Wilhite, S. "A diary study of task switching and interruptions." *Proc. of CHI*, 175-182, 2004.
2. Czerwinski, M., Smith, G., Regan, T., Meyers, B., Robertson, G. and Starkweather, G. (2003). Toward characterizing the productivity benefits of very large displays. In *Proceedings of Interact 2003*, pp. 9-16.
3. Gonzalez, V.M., Mark G. "'Constant, constant, multi-tasking craziness': managing multiple working spheres." *Proc. of CHI*, 113-120, 2004.
4. Grudin, J. (2002). Partitioning digital worlds: Focal and peripheral awareness in multiple monitor use. In *Proceedings of CHI 2002,* pp. 458-465.
5. Guimbretiere, F., Stone, M., and Winograd, T. (2001). Fluid interaction with high-resolution wall-size displays. In *Proceedings of UIST 2001,* pp. 21-30.
6. Hutchings, D., Czerwinski, M., Smith, G., Meyers, B., Robertson, G Display space usage and window management operation comparisons between single monitor and multiple monitor users. In *Proceedings of AVI 2004*.
7. MacIntyre, B., Mynatt, E., Voida, S., Hansen, K., Tullio, J., Corso, G. (2001). Support for multitasking and background awareness using interactive peripheral displays. In *Proceedings of UIST 2001,* pp. 41-50.
8. Mark, G., Gonzalez, V.M., Harris, J. "No task left behind?: examining the nature of fragmented work." *Proc. of CHI*, 321-330, 2005.
9. Mynatt, E., Igarashi, T., Edwards, W., and LaMarca, A. (1999). Flatland: new dimensions in office whiteboards. In *Proceedings of CHI 1999,* pp. 346-353.
10. Pedersen, E.R., Sokoler, T. "AROMA: abstract representation of presence supporting mutual awareness." *Proc. of CHI*, 51-58, 1997.
11. Plaue, C., Miller, T., Stasko, J. "Is a picture worth a thousand words?: an evaluation of information awareness displays." *Proc. of GI*, 117-126, 2004.
12. Robertson, G., Horvitz, E., Czerwinski, M., Baudisch, P., Hutchings, D., Meyers, B., Robbins, D., and Smith, G. (2004), Scalable Fabric: Flexible Task Management. In *Proceedings of AVI 2004* pp. 85-89.
13. Smith, G., Baudisch, P., Robertson, G., Czerwinski, M., Meyers, B., Robbins, D., and Andrews, D. (2003). GroupBar: The TaskBar Evolved. In *Proceedings of OZCHI 2003*, pp. 34-43.