

TaskZones: A Task Manager for Multiple-Monitor Systems

Dugald Ralph Hutchings[†]
GVU Center & College of Computing
Georgia Institute of Technology
Atlanta, GA 30332 USA
hutch@cc.gatech.edu

[†] This work was conducted as an intern at Microsoft Research.

Mary Czerwinski, Dan Robbins, George Robertson
Greg Smith, Brian Meyers
Microsoft Research
One Microsoft Way
Redmond, WA 98052 USA
marycz@microsoft.com

ABSTRACT

We present TaskZones, an alternative to Rooms and virtual desktops for grouping windows for multiple-monitor users. Key benefits of TaskZones are that it (1) allows a user to designate the windows of an arbitrary selection of monitors as a group, (2) allows a user to have multiple groups visible simultaneously, and (3) maintains positive properties of virtual desktops such as exploitation of spatial memory, quick keyboard-command switching and the ability to *hide* groups of potentially distracting windows quickly, which is of special interest to multiple-monitor users. Furthermore, TaskZones can be used on essentially any physical configuration of monitors. “Any configuration” includes single-monitors, which thus allows use of TaskZones while mobile and possibly limited to one screen.

ACM Classification: H5.2 [Information interfaces and presentation]: User interfaces – *GUI, Windowing systems*

General Terms: Design, Human Factors

Keywords: multiple monitors, window management, task management, TaskZones

INTRODUCTION

The rise in popularity and apparent usefulness of multiple-monitor systems has called into question the usability and utility of many interfaces [1]. In particular, standard window management techniques seem to be less effective, or at the very least less used, in the presence of increasing numbers of monitors [4]. Other techniques, such as the ability to hide distracting content in order to focus on a specific task, become more critical as increased numbers of pixels simultaneously increase the amount of visible information that can distract the user [3]. Indeed, a reason that virtual desktop (VD) users prefer VDs over a multiple-monitor system is that VDs allow them to direct their foci without being interrupted by changing information on other desktops [5].

We do not view multiple-monitors and VDs as mutually exclusive entities and have begun to develop TaskZones, an

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.

UIST '05, October 23–27, 2005, Seattle, Washington, USA.
Copyright ACM 1-59593-023-X/05/0010...\$5.00.

alternative window grouping interface to VDs for multiple-monitor users. When used on multiple-monitor systems, VDs provide both benefits and drawbacks, with the latter being closely tied to the observation that when virtual desktops were first developed [2], multiple-monitor systems were rare if not nonexistent. With TaskZones, we have attempted to maintain the advantages while avoiding the drawbacks.

Sometimes multiple monitor users will show windows from multiple tasks simultaneously while at other times they will spread the windows of a single task across all of the monitors [1]. However, VDs as applied to multiple monitors either span the entire display space or give each monitor an independent set of desktops, which forces a user to choose a specific type of use of multiple monitors. In TaskZones we have replaced the notion of a “desktop of windows” with a more general “group of windows” where each group is made up of the windows of an arbitrary selection of monitors (whether one, all, or a subgroup that does not necessarily have to be contiguous). Furthermore any number of groups (*i.e.* tasks and subtasks) can be visible simultaneously.

Advantages of a VD system are that it exploits spatial memory by arranging desktops in a physical layout and allows rapid group switch through keyboard commands. As we presently describe, TaskZones maintains these positive aspects of virtual desktops.

TASKZONES – BRIEF OVERVIEW

At the core of TaskZones is the notion of *the grid*. The grid contains a representation of all of the window groups (which are the tasks), and all task management interaction involves the grid in some way. Every task belongs to one grid location. We designed the grid so that it can easily be operated by the keyboard, inspired by the design of the ZoneZoom system [6]. The grid spans all of the monitors, but in such a way that no grid location straddles monitor bezels. Figure 1 (next page, top) shows how the grid would be displayed on 2-, 3-, and 9-monitor display systems, but note that the grid could be shown on essentially any configuration and does not have to be limited to the keypad.

There are three window grouping operations in TaskZones: create, show, and hide. Below, we describe in detail how the creation operation is conducted by a user; show and hide are very similar in their operation.

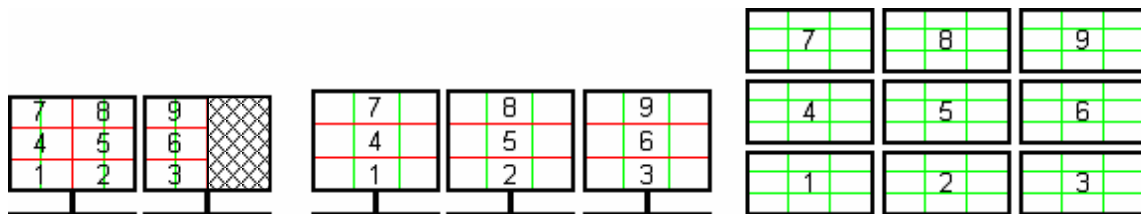


Figure 1. Three different physical monitor configurations are depicted. In each, red lines demarcate grid locations and green lines correspond to the monitor bezels in the specific configuration. The grid is placed on systems with 2, 3, and 9 monitors respectively left to right. Note that grid numbers correspond to the layout of a standard US-English keyboard keypad.

The user initiates the create operation by pressing the keyboard combination `<win>+<. >` (the decimal key on a keypad). Each monitor is assigned to and displays a number that also corresponds to the keypad. For example, in a 3-monitor system with monitors arranged left-to-right, the leftmost monitor would be assigned “1,” the middle monitor “2,” and the rightmost monitor “3” (Figure 2a). The default numerical arrangement of the desktops depends on the arrangement of the number keypad on the primary input device (cell-phone, remote control, or PC keyboard). Pressing the key that corresponds to the monitor selects all of the visible ungrouped windows on that monitor as members for the group that is about to be created. One of the interesting qualities of a keyboard keypad is that it allows three monitors in a row to be selected simultaneously if the user uses three adjacent fingers to hit the “1,” “2,” and “3” keys simultaneously.

Each operation has an *extension*, which allows a user to specify additional actions to be taken after an operation. For example, users might create a group and continue to interact with it so the windows in the newly created group are not hidden or they might desire to hide it immediately. The extension key allows the user to stow the group, and is operated by pressing the minus key (Figure 2c). As another example, suppose that a user is working with a group and creates a new window. This window may or may not belong to the group. TaskZones assumes that the window does not belong, but when a user hides the group, the extension key allows that window to be added to the group and removed from the screen. As a final example, after a group is retrieved, a user may want to use other windows that lay beneath the group. If not, the extension key for retrieve stows any groups that have windows that overlap the recently retrieved group. In essence, extensions help coordinate the several meanings that can be associated with users’ window operations by allowing to user to designate

times in which more powerful actions should be taken.

FUTURE WORK

A key component of the continued development of the TaskZones prototype will be system evaluation. We have already met the evaluation standard set by the Rooms system [2] by basing our design directly on qualities of both multiple monitor users and properties of task managers but we desire to exceed that standard by further comparing TaskZones against virtual desktop systems in laboratory studies and collecting comments about the system from users as they employ it in their everyday work.

REFERENCES

1. Grudin, J. Partitioning digital worlds: focal and peripheral awareness in multiple monitor use. In *Proc. CHI 2001*, ACM Press, 458 – 465.
2. Henderson, D. A. Jr. and Card, S. K. Rooms: The use of multiple virtual workspaces to reduce space contention in a window-based graphical user interface. *ACM Trans. on Graphics* 5, 3 (1986), 211 – 243.
3. Hutchings, D. R. and Stasko, J. Revisiting display space management: Understanding current practice to inform next-generation design. *Proc. Graphics Interface 2004*. Canadian Human-Computer Communications Society, 127 – 134.
4. Hutchings, D. R., *et al.* Display space usage and window management operation comparisons between single monitor and multiple monitor users. *Proc. Advanced Visual Interfaces 2004*, ACM Press, 32 – 39.
5. Ringel, M. When one isn’t enough: an analysis of virtual desktop usage strategies and their implications for design. *CHI Extended Abstracts 2003*, ACM Press, 762 – 763.
6. Robbins, D. C. *et al.* ZoneZoom: Map navigation for smartphones with recursive view segmentation. *Proc. AVI 2004*, ACM Press, 231 – 234.

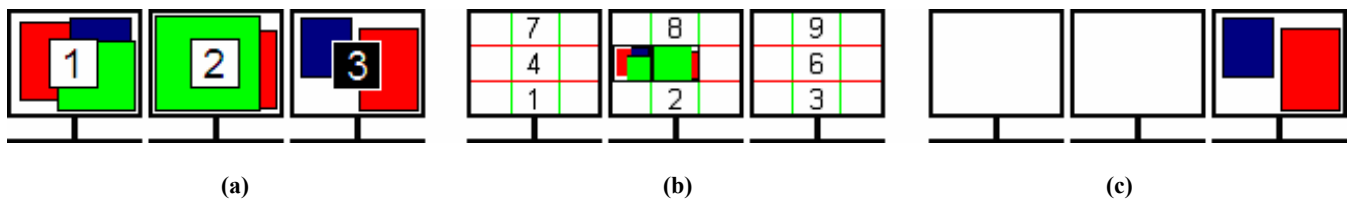


Figure 2. The creation operation in TaskZones. The user presses `<win>+<. >`, which shows the mapping of numbers to monitors. In (a), the user has selected monitors 1 and 2 but not 3. The user presses enter, which shows the grid. In (b), the user has placed the task in location 5. Notice how only windows from monitors 1 and 2 appear in the grid. After creating the task, the grid is hidden. In figure (c), the user has chosen to employ the *hide* extension of the operation, thus hiding the newly created task.