Window Shopping: A Study of Desktop Window Switching

Andrew Warr, Ed H. Chi, Helen Harris, Alexander Kucher, Jenn Chen, Robert Flack,
Nicholas Jitkoff

Google Inc.

1600 Amphitheatre Parkway, Mountain View, CA 94043

{andywarr, edchi, helenharris, kucher, jennschen, flackr, alcor}@google.com

ABSTRACT

Desktop users frequently open and switch between multiple windows. Here we present an experiment comparing 3 window switching interfaces: *the Cards interface* spreads windows out like a vertical stack of cards with the most recent window at the front; *the Exposé interface* provides an map-like overview based on the relative size and position of windows; and, *the Mosaic interface* places each window in a grid ordered by recency. Experimental results suggest that the Mosaic interface scales, enabling faster window selection than the Cards interface and less erroneous window selection than the Exposé interface.

Author Keywords

Window Management, User Interfaces

ACM Classification Keywords

H.5.2 User Interfaces

INTRODUCTION AND RELATED WORK

Multiple application windows are a ubiquitous feature of modern desktop operating systems, yet users often find switching between windows problematic. Since windows frequently obstruct other windows, and users often forget the location of windows, users often end up hunting for a specific window and the problem exacerbates as the number of windows increases. For example, Hutchings *et al.* [3] showed that participants in their study had 8 or more windows open 78.1% of the time, 3.5 of which were visible at any time on a single display. Highlighting the frequency of window switching, their results show windows were active for an average of 20.9 seconds and the median activation time took 3.8 seconds.

The 1980s was a golden age researching windowing systems in HCI. Early work showed that tiled windowing systems are quicker than overlapping windows [1]. Later research showed tiled windowing systems were also faster when there are large numbers of windows [4]. This is

intuitive since there are no overlapping windows to manage. Despite this, users generally prefer overlapping windows [1]. Therefore, it is imperative to create windowing systems that are both efficient for the user and meet their preferences.

In one early work, Card & Henderson [2] argued for designs that reduce the time and cognitive load of switching tasks and context. In their system called Rooms, a virtual window manager keeps track of active sets of windows relating to tasks and users could easily switch between different ‘rooms’ to complete their associated tasks.

Later researchers investigated providing the user with an overview of open windows, as well as user-defined sets of windows that might be associated with some task [5, 6, 7]. The Task Gallery [6] allowed windows to be grouped as virtual artworks representing tasks and arranged them in a virtual gallery, thereby leveraging human spatial cognition and perception. Scalable Fabric [5] used the periphery of the screen to hold scaled down groups of live windows that represented tasks. Adhering to the ‘window sets’ idea, a system called GroupBar [7], allowed users to organize windows into groups in Windows’ TaskBar. More recently, SCOTZ [8] presents windows in spatially consistent layouts, such as a spiral treemap, squarified treemap and a grid.

Inspired by these prior works, here we investigate the efficiency of three window switching interfaces (two of which we were considering for a new operating system). It is not the purpose of this paper to be a comprehensive design exploratory of window switching interfaces. Instead, our focus is to evaluate the window switching interfaces being considered for a new operating system, as well as a popular window switching interface, with focus on understanding the switching costs [2].

Cards: Warr and Chi [9] examined the Cards interface for mobile browser tab switching. Cards show a vertical stack of tabs/windows for users to touch and browse through before selecting one as the main window. We considered this design for desktop window management, where windows would be spread out like a stack of cards with the most *recent* window at the front, as depicted in Figure 1a. Since it performed well on mobile devices [9], a consistent experience across two devices might be preferable to users.

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. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from Permissions@acm.org.

CHI'16, May 07-12, 2016, San Jose, CA, USA

© 2016 ACM. ISBN 978-1-4503-3362-7/16/05...\$15.00

DOI: <http://dx.doi.org/10.1145/2858036.2858526>

Figure 1. The (a) Cards, (b) Exposé and (c) Mosaic interfaces

Exposé: The Exposé interface — a version of the Exposé window manager used in Mac OS X, depicted in Figure 1b — arranged windows into a map based on their relative size and original position.

Mosaic: We were also inspired by window switching interfaces that were spatially consistent [8], in which each window is placed in a grid ordered by recency (left-to-right, top-to-bottom). Figure 1c depicts our implementation called the Mosaic interface, which not only arranged windows in a grid but also scales the window to the size of the grid cell.

Inspired by Card & Henderson [2] our research goal is to characterize these different interfaces' effects on user performance, in terms of selection time and errors. To our knowledge, this is the first comparative study of these specific window overview schemes.

EXPERIMENTAL DESIGN

We used a 3 x 3 mixed factorial design. The between-factor independent variable was the window switching interface (Cards, Exposé and Mosaic interfaces), and the within-factor independent variable was the number of open windows (3, 6 and 9 windows). While previous research [3], showed participants in their study had 8 or more windows open 78.1% of the time, usage data of an operating system showed that the 90th percentile of users (N=2240) have 9 or fewer windows open. To mitigate ordering effects, we balanced the levels of the within-factor variable using latin-squares. The dependent variables were response time to select a window and the number of erroneous selections.

We surmised that participants would be faster and less error prone selecting windows with the Mosaic interface compared to the other interfaces, because the Mosaic design is more structured and therefore more predictable compared to the Exposé interface. Moreover, the Mosaic interface did not require scrolling, unlike the Cards interface. Formally, we hypothesized that:

- H1: The Mosaic interface would allow for faster selection times of windows than the Cards and the Exposé interfaces;
- H2: The Mosaic interface would result in fewer erroneous window selection than the Cards and the Exposé interfaces.

Subjects: 42 participants (27 females & 15 males) took part in this experiment, with ages ranging 18 to 60 (median range is 31-40). We recruited from a database of participants who had signed up to participate in user research studies. We pre-screened participants for (1) using a laptop as their primary computing device; (2) not having used any of the window switching interfaces previously; and (3) on average having more than one window open at a time. Participants received a \$100 perks.com gift card for participating in the experiment.

Material: We brought participants into a controlled lab setting to use a laptop with a mouse attached. In a pilot, we found many participants were not familiar using a touchpad. Furthermore, while the window switching interfaces supported alt-tab it was disabled for this experiment to avoid a confounding variable.

We developed a full-screen web application, which allowed us to prototype the window switching interfaces, control the experimental procedure and record experiment data.

The web application presented a desktop with a taskbar at the bottom of the screen and a number of application windows. The taskbar presented instructions indicating the application window to switch to next. The participant could press a designated key on the keyboard to enter the window switching interface.

Procedure: The experiment consisted of trials in which subjects were instructed to enter the window switching interface and switch to a specified window as quickly as possible by selecting that window using the cursor. The experimenter first demonstrated the Card, Exposé or Mosaic interface depending on the assigned condition for the subject. Participants were then asked to practice selecting windows until they were comfortable with the procedure. Participants then completed all the trials for a given condition. In a single trial, the application recorded the time from pressing the key on the keyboard to enter the window switching interfaces to selecting the specified window. If the participant did not select the correct window, the timer continued until the correct window was selected and erroneous selections were recorded.

Once a trial ended, the participant was asked to select another window. To mitigate learning effects, between trials, we changed the order of the windows in the Cards

Figure 2. Mean total selection time for the Cards, Exposé and Mosaic interfaces for the 3, 6 and 9 window conditions

and Mosaic interfaces; and the location of the windows in the Exposé interface. Participants traverse every possible distance for a given condition (2 through $N-1$, where N is the number of windows in the condition). The distances to traverse were chosen randomly without replacement from the set. This was done to mitigate possible learning effects between trials. To ensure a valid comparison across participants' changes remained the same for a given number of windows.

Participants repeated the above procedure for the 3, 6 and 9 window conditions based on a predetermined latin-squares order. The order of windows to be selected was randomized between the 3, 6, and 9 window conditions to mitigate learning effects within conditions.

RESULTS

Selection Time: Table 1 and Figure 2 shows the mean total selection time for the interfaces. Mauchly's test showed no significant difference ($W=0.91$, $p > 0.05$) and therefore sphericity was not violated. A mixed-factorial ANOVA on natural logarithm of the total selection time found significant differences between interfaces ($F(2,39)=16.5$, $p<0.001$, $d=0.37$) and number of windows ($F(2,78)=468.3$, $p<0.001$, $d=0.79$), as well as an interaction effect ($F(4,78)=2.87$, $p<0.05$, $d=0.04$).

	3 windows	6 windows	9 windows
Cards	6143.8 (4220.7)	17992.6 (9611.4)	31975.7 (8869.1)
Exposé	3774.3 (3076.5)	9611.4 (2503.6)	18919.3 (7710.5)
Mosaic	3998.0 (1278.7)	9099.9 (2575.8)	15148.6 (3430.8)

Table 1. The mean selection time (ms) for the Cards, Exposé and Mosaic interfaces for the 3, 6 and 9 window conditions

Figure 3. Mean total number of errors for the Cards, Exposé and Mosaic interfaces for the 3, 6 and 9 window conditions

Post-Hoc TukeyHSD tests showed a significant differences between the Cards and Exposé interfaces for the 3 ($p<0.05$), 6 ($p<0.001$) and 9 ($p<0.001$) window conditions; and the Cards and Mosaic interfaces for the 3 ($p<0.1$), 6 ($p<0.001$) and 9 ($p<0.001$) window conditions. Table 1 and these results suggest that we can partially accept hypothesis H1 that it is faster to select windows using the Mosaic interface compared to the Cards interface.

Error Rates: Table 2 and Figure 3 shows a chart for the total mean number of errors for the Cards, Exposé and Mosaic interfaces. Mauchly's test showed a significant difference ($W=0.08$, $p<0.001$), therefore degrees of freedom were corrected using Greenhouse-Geisser estimates of sphericity ($\epsilon=0.52$). A mixed-factorial ANOVA on the mean total number of errors found significant differences between interfaces ($F(2,39)=3.73$, $p<0.05$, $d=0.07$) and number of windows ($F(2,78)=4.65$, $p<0.05$, $d=0.06$), as well as an interaction effect between interfaces and number of windows ($F(4,78)=4.29$, $p<0.01$, $d=0.12$).

	3 windows	6 windows	9 windows
Cards	0.0 (0.0)	0.1 (0.5)	0.1 (0.4)
Exposé	0.0 (0.0)	0.2 (0.8)	3.0 (5.4)
Mosaic	0.0 (0.0)	0.0 (0.0)	0.0 (0.0)

Table 2. The mean total errors for the Cards, Exposé and Mosaic interfaces for the 3, 6 and 9 window conditions

Post-Hoc TukeyHSD tests showed a significant difference between the Cards and Exposé interfaces ($p<0.05$); and the Exposé and Mosaic interfaces ($p<0.05$) for 9 window condition. Table 2 and these results suggest we can partially accept hypothesis H2 that there would be fewer erroneous selections with the Mosaic interface than the Exposé interface, particularly for the 9 window condition.

DISCUSSION

Our experiment showed that it is generally faster to switch windows using the Mosaic interface compared to the Cards interface. The question is *why*?

First, the Cards interface sometimes required the user to scroll when moving to another window further away in the stack. As the number of windows increased, so did the selection time, since the users have to scroll further. However, in the 3 window condition, participants generally did not scroll since all windows were partially visible, so participants were able to immediately select a window without scrolling. This implies that Cards performed reasonably for low numbers of windows, while inefficient for large numbers of windows. As one participant mentioned: *“Vertical scrolling is very inefficient.”* Interestingly, this is exactly the opposite of the findings on mobile devices [9], where Cards is more efficient for large number of windows, perhaps due to the differences between screen size, as well as mouse and touchscreen interactions.

Second, Exposé also seems to have trouble scaling to large numbers of windows. Exposé interface arranged windows based on their relative size and position, allowing all open windows to be seen at once: *“It is nice to have an overview of everything you are looking at.”* However, as the number of windows increased, it became more difficult to parse the Exposé interface since the window positions in the interface often lacked apparent structure [8]. Moreover, the sizes of windows in the Exposé interface was determined by the number of windows and their sizes. As the number of windows increased the windows size in the interface became smaller, which was particularly problematic for smaller windows: *“The only issue is that some things are pretty small, like the sticky notes.”*

Finally, the Mosaic interface did not suffer from some of the design problems the Cards and Exposé interfaces did. Participants did not have to scroll in the Mosaic interface and the layout of windows was structured and spatially consistent making them easier to parse, because the Mosaic interface arranged windows in a grid, allowing all open windows to be seen at once. While hunting for the window in a grid could still be problematic, it is less problematic than Exposé: *“It was very easy to see all the apps together and see the one I want.”*

Our experiment did not show any difference in the number of erroneous selections between the window switching interfaces, with the exception of the Exposé interface in the 9 window condition. We surmise that significant differences exist when there are an even larger number of windows (>9), where the thumbnails become smaller, and harder to select, according to Fitts' law.

LIMITATIONS

Like many experiments, the realism of the tasks can be improved. For example, when people are working with

windows in a natural setting, they may remember where they are. The randomization that we applied mitigated these learning effects for the purpose of a controlled experiment. As such, these results should be considered a worst-case scenario.

CONCLUSION

Our experimental results suggest that the Mosaic interface enables fast, less error prone window selection. While window management on the desktop is a classic HCI problem, our research suggests there are still room for understanding and improvement, especially for new window switching interfaces invented in the last decade. These findings could also generalize to other task management and information space interfaces on the desktop, such as web browsers and multi-screens. It is our hope that we have pointed researchers toward directions that could make further improvements.

REFERENCES

1. Bly, S. A. and Rosenberg, J. K. 1986. A comparison of tiled and overlapping windows. Proc. CHI'86, ACM Press, 101-106.
2. Card S. K. and Henderson, A. 1987. A Multiple, Virtual-Workspace Interface to Support User Task Switching. Proc CHI'87, ACM Press, 53-59.
3. Hutchings, D. R., Smith, G., Meyers, B., Czerwinski, M., Robertson, G. 2004. Display space usage and window management operation comparisons between single monitor and multiple monitor users. Proc. AVI'04, ACM Press, 32-39.
4. Kandogan, E. and Shneiderman, B. Elastic windows: Evaluation of multi-window operations. Proc. CHI'97, ACM Press, 250-257.
5. Robertson, G., Horvitz, E., Czerwinski, M., Baudisch, P., Hutchings, D., Meyers, B., Robbins, D. and Smith, G. 2004. Scalable Fabric: Flexible task management. Proc AVI'04, ACM Press, 85-89.
6. Robertson, G., van Dantzich, M., Robbins, D., Czerwinski, M., Hinckley, K., Ridsen, K., Thiel, D. and Gorokhovskiy, V. 2000. The task gallery: a 3D window manager. Proc CHI'00, ACM Press, 494-501.
7. Smith, G., Baudisch, P., Robertson, G., Czerwinski, M., Meyers, B., Robbins, D. and Andrews, D. 2003. GroupBar: the taskbar evolved. Proc OZCHI'03, ACM Press, 34-43.
8. Tak, S., Scarr, J., Gutwin, C. and Cockburn, A. 2011. Supporting Window Switching with Spatially Consistent Thumbnail Zones: Design and Evaluation. Proc INTERACT'11, LNCS 6946, 331-347.
9. Warr, A. and Chi, E. H. 2013. Swipe vs. scroll: web page switching on mobile browsers. Proc CHI'13, ACM Press, 2171-2174.