ShoCons: Effective Display of Shortcuts in Icon Toolbars

Vidya Setlur[1] and Benjamin Watson[1][a]

[1] *Tableau Research, Palo Alto, USA*

[2] *Dept. Computer Science, North Carolina State Univ., Raleigh, USA*

vsetlur@tableau.com, bwatson@ncsu.edu

Keywords: Keyboard Shortcuts, Menus, Toolbars, Icons

Abstract: Users often do not use keyboard shortcuts in applications as recalling and choosing the correct shortcut is a higher-order cognitive task. Mouse driven menus, toolbars, and icons are easier for a user to learn because they present hints and make visible what operations are possible, drawing on the power of recognition rather than recall. How can we better support the usage of shortcuts with such menus? Two existing methods are text in the icons, and popups with mouse hover. While the first is space inefficient; the second limits exposure and imposes an interaction cost. We propose a third method, ShoCons, that is spatially more efficient and neither limits user exposure nor imposes an interaction cost. To achieve this, ShoCons use a succinct iconic display of meta keys, limiting textual display to one character. We examine these alternatives in a controlled study, and find that when used with a high-level task, ShoCons enable faster task performance and an immediate increase in the accuracy of shortcut use.

1 INTRODUCTION

Making functionality easier to access for users is a problem common to all interactive software. Toolbars are prevalent in graphical user interfaces because they make visible what operations are possible, substituting the ease of recognition for the difficulty of recall, making the interface easier to learn. Icons are commonly used in toolbars as they are visually distinctive, and can succinctly convey the semantics of the underlying information (Familant and Detweiler, 1993).

A keyboard shortcut is a combination of keystrokes used to carry out some operation that would otherwise be executed with the mouse. In desktop applications, most shortcuts use one or more metakeys (*i.e.*, the Ctrl, Alt, and Shift keys) in combination with another key, creating a chord. Because they allow users to perform operations without deviating significantly from their tasks, they are generally viewed as more efficient than using icons in toolbars (Lane et al., 2005). As users transition from being novices to experts, one would expect them to employ shortcuts more often. Unfortunately this is not the case. Shortcuts can be hard to learn and remember, because unlike menus and toolbars, they often have no visual representation (Lane et al., 2005).

[a] https://orcid.org/0000-0002-3758-7357

2 RELATED WORK

2.1 Shortcut Display in Icons

There are two prevailing methods for displaying keyboard shortcuts. Pop-ups display on demand a textual representation of the shortcut next to the corresponding icon, and are often used with toolbars. For example, Microsoft Office displays a tooltip when the cursor hovers over an icon. Pop-ups preserve the visibility of icons, but hide the keyboard end of the shortcut mapping, requiring user action before making it completely visible. This limits the both the exposure to and incidental learning of mappings that is important to effective integration of shortcuts in interfaces (Grossman et al., 2007). Static or continuous hints always display a textual representation of the shortcut, and are regularly used in text menus such as OS X. While they can display both ends of the shortcut mapping, the display space they dedicate to key display can reduce the visibility of the mapping to application functionality.

2.2 Measuring Usage and Efficacy of Keyboard Shortcuts

Lane et al. explored the efficiency of shortcuts (2005), finding that once learned, drop-down menus are less

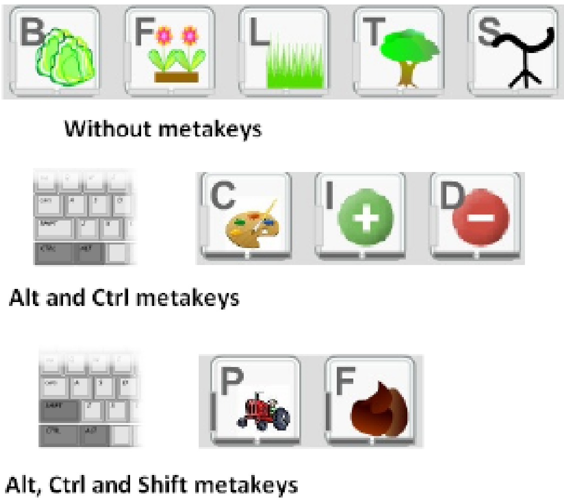

Figure 1: *ShoCons* encodings for various shortcut metakey combinations. Hash marks in the border of the *ShoCons* correspond to the positioning of the Alt, Control and Shift metakeys on a QUERTY keyboard.

efficient than icon toolbars, and toolbars are less efficient than shortcuts. In another study, Odell et al. examined the efficiency of toolbars, shortcuts physically grouped by related functionality, and shortcuts mapped to keys with a lexical relationship to the function's name (Odell et al., 2004). They found that physically grouped shortcuts were fastest, while toolbars were slowest. The literature on flow applied to user interfaces provides a good motivation for why users should make the transition from icon toolbars to keyboard shortcuts (Bederson, 2004). Bederson describes different levels of skill acquisition, where in the final autonomous stage, the command is almost exclusively facilitated by keyboard shortcuts.

Several researchers worked toward easing the transition to shortcuts. Grossman et al. (2007) found that audio display of shortcut mappings and disabling command access through menus was quite effective. In the *Blur* system (Scarr et al., 2011), when the user executes a command using a toolbar, *Blur* displays a transient, visual "calm notification" reminding them of the equivalent shortcut. When arranging several objects in Powerpoint, *Blur* users made the expert transition more easily than with traditional shortcuts. *ExposeHK* overlays key combinations onto existing UI widgets when a metakey is pressed (Malacria et al., 2013), making shortcuts visible on demand.

3 Contributions

We propose a new method of shortcut display called *ShoCons*, which uses a succinct iconic representation of shortcut metakeys (Shift, Ctrl and Alt) that mirrors the layout of those keys on the QWERTY keyboard (Figure 1). Our work:

- *Demonstrates the potential of continuous shortcut display.* Rather than making display of shortcut mappings conditional on mouse (Grossman et al., 2007) or keyboard interaction (Malacria et al., 2013) (Scarr et al., 2011), *ShoCons* enables continuous display of shortcuts, increasing the visibility of shortcut keyboard mappings. This simplifies their discovery, eases their confirmation, and better supports the incidental and associative learning described by Grossman et al. Our experiments show that *ShoCons*' continuous shortcut display can significantly improve task performance and accuracy of shortcut use.
- *Proposes and verifies a visually efficient shortcut display.* Continuous display of shortcut mappings might overload the interface, which is already quite busy in complex applications, and may introduce a tradeoff that obscures the very functionality mappings it hopes to reveal. Our experiments confirm that *ShoCons* offers significant improvements in task performance and accuracy of shortcut use over a less concise, continuous shortcut display.
- *Examines shortcut usage in a higher level context.* Most previous work on shortcuts evaluated their effectiveness using disjoint, short, repetitive tasks (*e.g.*, "press 'A' now") to accelerate and enable the study of learning in the lab. Inspired by Scarr et al.'s (2011) use of a higher-level Powerpoint layout task to improve the external validity of their evaluation, we use a high-level garden building task to evaluate *ShoCons*. Our experiments suggest that even in this challenging context, *ShoCons* can enable rapid improvements in task performance and accuracy of shortcut use.

4 EVALUATING SHOCONS

To test whether *ShoCons*' continuous display of shortcut mappings could improve task performance and shortcut learning, we performed an experiment comparing our continuous, succinct *ShoCons* display to a a traditional continuous shortcut display (*verbose*, Figure 3(a)), and a popup display that shows map-

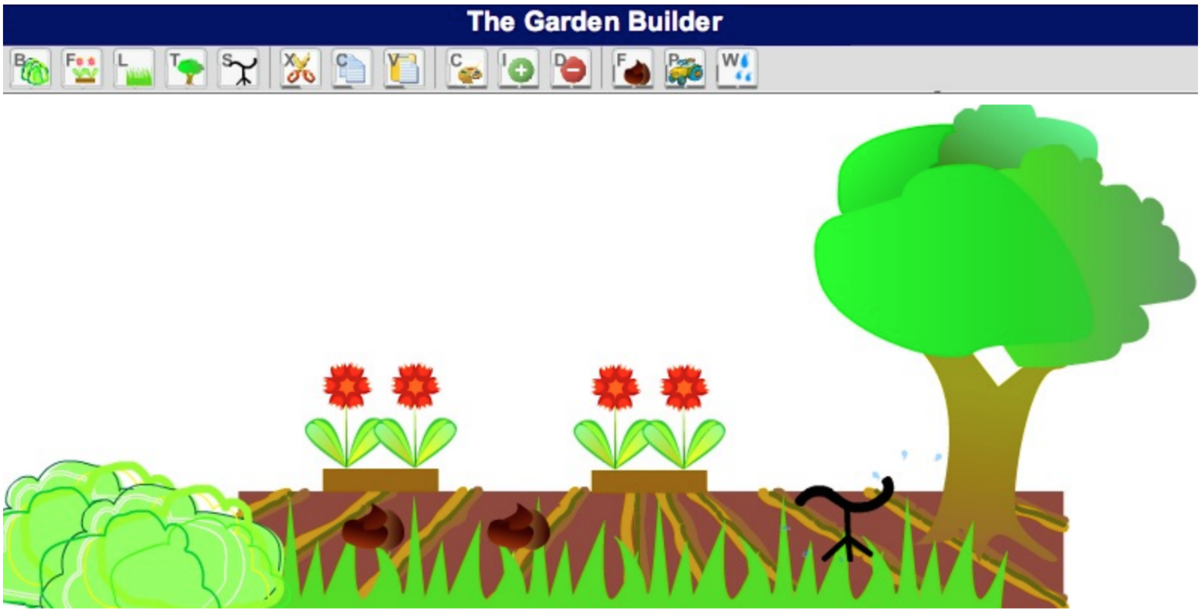

Figure 2: Experimental interface with *ShoCons* (continuous succinct) display in the toolbar and a participant's garden. We do not show the very similar target garden at the bottom.

(a) Text in the icon (continuous verbose)

(b) Popups (conditional)

Figure 3: The other two different experimental shortcut displays in addition to *ShoCons*. There are no popups in (b) because the user has not just pressed the Alt key.

pings only when the Alt key is pressed (*conditional*, Figure 3(a)). We hypothesized that:

- **Continuous is better:** Continuous display (*verbose* & *ShoCons*) enables better task performance and learning than conditional display.
- **Succinct is better:** Succinct continuous display (*ShoCons*) enables better performance and learning than verbose display.

4.1 Method

4.1.1 Participants and Apparatus

18 participants aged from 25 to 45 took part in our experiment, all with normal or corrected-to-normal vision. They used a Windows 7 2.0GHz 4GB PC with a 1280 × 1024 LCD monitor, mouse, and keyboard.

4.1.2 Task and Stimuli

In each trial, participants used a computerized gardening interface to recreate a target garden (Figure 2). A successful recreation matched the target garden by having the same number, types and sizes of objects, placed in the same vertical and horizontal orders. The gardening interface showed a toolbar at the top with an icon for each possible garden operation, a display below of the participant's current garden, and the target garden at the bottom. Gardens were made up of five object types including grass, flowers, bushes, trees and sprinklers, each of which could vary in location and size (except fixed size sprinklers). Target gardens varied in both the complexity of the target, and of the shortcut key combinations that could be used. All participants used the same gardens and shortcut key combinations.

There were five operations for inserting objects (one per type), six editing operations (cut, copy, paste, color, insert and delete), and four maintenance operations (irrigating, pruning, plowing and fertilizing). Maintenance operations either changed object state (e.g. making sprinklers shoot water) or appearance (e.g. fertilizing made objects larger). Maintenance operations were present only if they could be used in the current garden. Each icon had a matching keyboard shortcut. Icons in the toolbar displayed shortcuts in one of the three ways discussed above: continuous succinct display (*ShoCons*, Figure 1 and the top of Figure 2); continuous verbose display, with text in the icon (Figure 3(a)); or conditional display, with higher-contrast icon text appearing only when the Alt key was pressed and disappearing with the next action (Figure 3(b)). To interact with the interface, participants could click on an icon, or use the matching keyboard shortcut. To select an object, participants could click on it, or use the tab key to cycle through them. To move an object, participants could either drag it with the mouse, or use arrow keys.

4.1.3 Procedure

We provided each participant with written instructions explaining their task and illustrating each of the shortcut displays. The instructions asked participants to complete each task as quickly as possible, use shortcuts as often as they were able, and take breaks as needed between any two trials. To gain some familiarity with the gardening interface and the different shortcut displays without also beginning to learn about their combination, participants practiced each in isolation: they built simple gardens using a purely textual interface without shortcuts until successful twice, and then practiced with each shortcut display without gardening functionality until they input the matching keystrokes twice. The interface displayed a visual message informing participants of when they successfuly completed the garden. However, participants were free to continue to the next garden before this message appeared, causing an error. On average, only 3 of a participant's 108 gardens were errors.

4.1.4 Design

We used a two-factor (3 shortcut displays x 3 blocks) design. Both variables were within subject. Shortcut display showed shortcut key mappings to participants using verbose, *ShoCons* or conditional display. Blocks formed trials within each shortcut display into three sequential groups, letting us sample the learning process. To control for learning interference between different shortcut displays, the order in which participants worked with different shortcut displays was completely counterbalanced across subjects. Moreover, as in Grossman et al. (2007), each participant used three different key mappings, so that learning of key mappings would not continue as shortcut display varied. All participants experienced the same key mappings, but their order and pairing with shortcut display was completely counterbalanced across subjects.

As dependent measures of task performance, we used *time*, the average number of seconds elapsed between target garden display and trial completion; and *error*, the percentage of trials completed incorrectly. To measure shortcut usage we recorded *achievement*, the proportion of all operations executed that were shortcuts; and *accuracy*, the proportion of the number of all key combinations that successfully executed a shortcut. There was no reason to use key combinations unless one was attempting a shortcut, so accuracy measured the success of participants when attempting to use shortcuts.

Participants performed 108 trials: 36 with each display, with each block containing 12 trials. Average trial time was 27.67 seconds, meaning that participants finished their trials in just under 50 minutes. The number of commands performed per garden varied between 2 and 11, with a median of 5.

4.2 Results

For all of our analyses, we averaged the times, errors, achievement and accuracy within each block. Each block contained 12 trials, producing 9 aggregated trials from the original 108 for each participant: one for each combination of shortcut display and block.

To confirm that key mapping and shortcut display order did not cause a confound in our results, we performed a four-way ANOVA (*key mapping*, *display order*, *shortcut display* and *block*) that included these experimental parameters as independent variables. Neither had any significant main effects, nor any meaningful interactions with *display* or *block*.

We therefore performed a 2-way ANOVA on shortcut display and block alone. We found no effects on achievement (overall achievement was 51%), and while block significantly affected error ($F(2,34) = 4.35$, $p < 0.01$), its effect size was quite limited, changing error from 1.8% to 4.6%, an increase of roughly 1 in 36 gardens per display. We therefore confine our remaining discussion of results to *time* and *accuracy*.

In a 2-way ANOVA, shortcut display had significant main effects on both time (F(2,34)=2833.5,

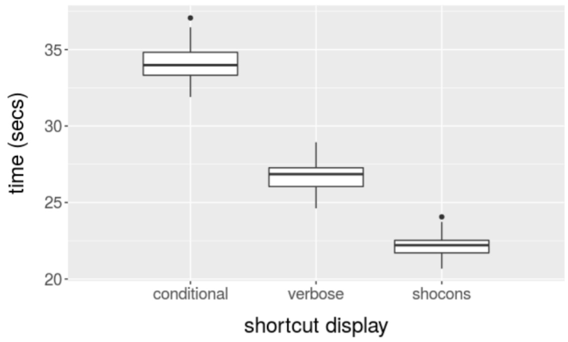

Figure 4: Effects of shortcut display on time. The box center is at median, the hinges are quartiles, and the whiskers extend to the last data point less than 1.5 IHQ from the hinge, where IHQ is the distance across the quartiles.

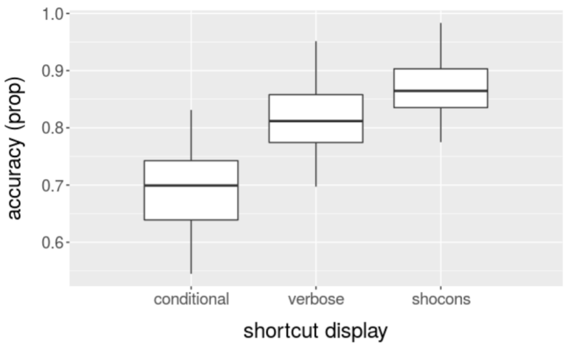

Figure 5: Effects of shortcut display on accuracy. Box plots are configured as in Figure 4.

$p < .0001$) and accuracy ($F(2,34)=84.5$, $p < .0001$). Block had no effect, and did not interact significantly with display.

Figure 4 shows the effects on time. Significant pairwise comparisons using contrasts showed that all three mean times (34.16, 26.67 and 22.17 seconds; σ = 10.91, 11.96 and 8.43) differed from one another significantly. Both the verbose and *ShoCons* displays enabled task performance that was meaningfully (22% and 35%) faster than performance with the conditional display. *ShoCons* communicated the shortcut key mappings more clearly than the verbose continuous display, permitting a performance speedup of over four seconds.

Figure 5 shows the effects of shortcut display on accuracy. Again significant pairwise comparisons using contrasts showed that all three means were significantly different. While only two-thirds of shortcut attempts with the conditional shortcut display were accurate, 81% of the attempts with the verbose display were, and 87% of the attempts with *ShoCons* were (σ = 27%, 23% 18%). This difference in accuracy of shortcut use likely explains much of the difference in time between the conditional and continuous displays.

5 DISCUSSION

We begin our discussion with a review of our hypotheses, and continue with two research questions.

Continuous is better: This hypothesis was partially confirmed. Continuous shortcut display did indeed permit faster task performance than conditional display. It also enabled much more immediately accurate use of shortcuts than conditional display. However, despite some promising trends across blocks, we found no concrete evidence that continuous display improves learning over time. Instead, the improvements in task speed and shortcut accuracy were almost immediate.

Succinct is better: This hypothesis was also partially confirmed. The succinct *ShoCons* display permitted faster task performance and immediate shortcut accuracy than verbose display, even though its shortcut hints were smaller than the verbose display's. However, succinct display also did not clearly accelerate shortcut learning over time. Rather, these improvements were again almost immediate.

Why did patterns of shortcut usage and learning differ from previous work? In previous research (Grossman et al., 2007) (Scarr et al., 2011) (Malacria et al., 2013), improved learning consistently showed itself in significant differences in measured achievement (proportion of shortcut usage), and faster improvement in achievement over time. Why are these more gradual patterns absent in our results? First, recall that we asked participants to maximize shortcut usage. For this reason shortcut usage may have rapidly reached a ceiling. Perhaps more importantly, like Scarr et al. (2011), we used a high-level experimental task with many components that may have slowed shortcut learning, and certainly reduced the rate at which we could experimentally sample shortcut learning (with repetitious requests to execute shortcuts). For example, selecting and positioning objects was a central component of the experimental task, and that *ShoCons* did not display the shortcuts for those operations. Users may have relied heavily on positioning with the mouse, limiting achievement.

Will ShoCons *scale to larger interfaces?* The obvious objection to continuous display of shortcut key mappings is that it adds additional information to already busy interfaces. While it is fortunate that *ShoCons* is more effective in communicating shortcuts than the less succinct verbose display, the objection still holds. If shortcut display is indeed to be continuous, it may prove impossible to eliminate this concern entirely. However, it may be possible to mitigate it by throttling continuous display to match the specific learning needs of a user.

6 LIMITATIONS

While our experimental task had many similarities to Scarr et al.'s (2011), it was unusual, giving rise to a unique pattern of results. Future work might replicate or extend our evaluation of *ShoCons* and continuous display by using a high-level task with a more complete use of shortcut display, or a low-level task enabling a more direct comparison to prior work.

Even when succinct, continuous shortcut display introduces tradeoffs between display of functionality and shortcut mappings — the difference we found between verbose and ShoCon task performance may be evidence of this. Future work might explore this tradeoff, e.g. by varying the complexity of the user interface, or adjusting shortcut display salience to match shortcut familiarity.

7 CONCLUSION

In this paper, we presented *ShoCons*, a visual technique that allows succinct and continuous display of shortcut keyboard mappings in toolbars. We found that when compared to standard alternatives for shortcut display, *ShoCons* significantly speeds task performance and improves accuracy during shortcut use; even when that task contained significant components (e.g cognitive planning and object positioning) that did not benefit greatly from shortcuts.

Future work should examine the questions raised by our work more closely. Does the improvement in shortcut learning grow over the longer term in these more complex task settings? Might an improvement in learning as measured by achievement eventually appear? Finally, what sort of information load does continuous shortcut display place on user interfaces?

8 ACKNOWLEDGEMENTS

Thanks to our anonymous reviewers, who helped improve this paper. We are also grateful to our psychology colleague Anne McLaughlin for her statistical advice, and to the many Nokia Research employees who participated in our experimental evaluation.

REFERENCES

Bederson, B. B. (2004). Interfaces for staying in the flow. *Ubiquity*, 2004(September):1–1.

Familant, M. E. and Detweiler, M. C. (1993). Iconic reference: Evolving perspectives and an organizing framework. *Int. J. Man-Mach. Stud.*, 39(5):705–728.

Grossman, T., Dragicevic, P., and Balakrishnan, R. (2007). Strategies for accelerating on-line learning of hotkeys. In *Proc. ACM CHI*, pages 1591–1600, New York, NY, USA. ACM.

Lane, D. M., Napier, H. A., Peres, S. C., and Sandor, A. (2005). Hidden costs of graphical user interfaces: Failure to make the transition from menus and icon toolbars to keyboard shortcuts. *International Journal of Human-Computer Interaction*, 18(2):133–144.

Malacria, S., Bailly, G., Harrison, J., Cockburn, A., and Gutwin, C. (2013). Promoting hotkey use through rehearsal with exposehk. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 573–582, New York, NY, USA. ACM.

Odell, D. L., Davis, R. C., Smith, A., and Wright, P. K. (2004). Toolglasses, marking menus, and hotkeys: a comparison of one and two-handed command selection techniques. In *Proc. GI*, pages 17–24, Canada. Canadian Human-Computer Communications Society.

Scarr, J., Cockburn, A., Gutwin, C., and Quinn, P. (2011). Dips and ceilings: Understanding and supporting transitions to expertise in user interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 2741–2750, New York, NY, USA. ACM.