# Not a Question of If, but When?

# Choosing the Right Trigger to Encourage Keyboard Shortcut Use

**Master Thesis**

**Dominik Seger, B.Sc.**
Institute of Psychology
Department for Cognitive Psychology and Methodology
University of Basel

November 2013

**Thesis Supervisors:**

**Silvia Heinz, M.Sc.**
Center for Cognitive Psychology and Methodology, University of Basel

**Prof. Dr. Klaus Opwis**
Center for Cognitive Psychology and Methodology, University of Basel

Non-plagiarism statement

I hereby declare that this thesis is my own work and that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

## Abstract

Keyboard shortcuts are efficient but underused by users. This study examined system notifications as triggers in software to increase keyboard shortcut use. Because previous studies suggested that the right timing could be crucial when promoting efficient user behavior, two different points of initiations of system notifications (early vs. delayed) were compared against each other. In a laboratory study, 76 participants performed repetitive tasks with a new personal calendar application. One third was exposed to an early notification to increase awareness of the keyboard shortcut feature, another third received the same notification delayed, and the last third received no trigger at all. Alongside performance being measured, participants rated the tools' usability. Results indicate that increased keyboard shortcut use was due to triggers. Furthermore, the early trigger was more effective than the delayed trigger. However, keyboard shortcut use could not be linked to performance or perceived usability. Further, participants who were triggered tended to be characterized by prior keyboard shortcut preferences. Implications of the results are discussed and a simple but effective method is presented to encourage keyboard shortcut use.

## Introduction

Today, general-purpose applications are often designed in a way that they are particularly easy for novices to learn. Their graphical user interfaces usually consist of drop-down menus and icon bars, showing all possible commands during software use and therefore eliminating the need to memorize commands. In the case of repetitive tasks, however, navigating through hierarchical menus is not very efficient (Lane, Napier, Peres & Sándor, 2005). This is the reason why many applications provide keyboard shortcuts to allow a transition to more efficient user behavior. However, research shows that users can be quite reluctant when it comes to learning new keyboard shortcuts, even when the demand for efficient user behavior is high, and, ironically, users themselves report efficient computer work to be important for them (Tak, 2007).

Several methods have been proposed and tested to promote keyboard shortcut use. Alternative menu designs (Grossman, Dragicevic & Balakrishnan, 2007), overlay tool tips (Malacria, Bailly, Harrison, Cockburn & Gutwin, 2013) and even a newly designed keyboard (Bailly, Pietrzak, Deber & Wigdor, 2013) have all been developed with the purpose to leverage keyboard shortcut use. System notifications, short messages reminding the user of the availability of enhanced user behavior features, have been used in a recent study to foster customization behavior (Banovic, Chevalier, Grossman & Fitzmaurice, 2012). However, whether system notifications could also be applied to promote keyboard shortcut use has yet to be examined.

The aim of this study was to evaluate system notifications as a new simple method to encourage keyboard shortcut use. Almost eighty participants were recruited and asked to perform tasks with a newly developed personal calendar application, where it was up to the participant's own choice whether to use keyboard shortcuts or traditional menu navigation. Apart from comparing system notifications against a control group, two different points of initiations of system notifications (early vs. delayed) were compared against each other in order to identify the best timing for behavior alteration. Because tracking keyboard shortcut use and measuring behavior efficiency very much depend on the computer specifications and especially the configuration of the physical input devices (keyboard and mouse), a laboratory setting was considered to be the best suitable environment for this study.

## Theoretical Background

### From Novice to Expert User Behavior

There are multiple ways to maximize user performance when interacting with software. In order to enable a transition from novice to expert behavior, the software can either (1) track individual user behavior and rearrange preferred features automatically for faster access (e.g., Gajos, Czerwinski, Tan, & Weld, 2006), (2) allow users to customize software features according to their own needs (e.g., Shneiderman, 2003), or (3) allow the user to maximize performance by learning and applying keyboard shortcuts (e.g., Lane et al., 2005). Keyboard shortcuts, also known as hotkeys or accelerators, have been available in user interfaces for decades, allowing users to perform operations quickly by pressing a key or a sequence of keys without the need to access a menu by mouse (Card, Moran & Newell, 1980). While Nielson (1992) already pointed out the importance of shortcuts twenty years ago, human-computer interaction (HCI) research has reached a consensus that using keyboard shortcuts is a timesaving alternative to navigation through traditional menus (Grossman et al., 2007; Lane et al., 2005; Malacria et al., 2013). The gain in time might be negligible in the case of rarely used actions, but rises substantially when repetitive tasks are performed over a long period of time. Keyboard shortcuts then make a big difference in *efficiency*, which is one key component in usability (Nielsen, 2012).

Despite the advantage over mouse-based techniques, keyboard shortcuts remain underused, as it was found, for example, in a laboratory experiment by Tak (2007). Keyboard shortcuts were objectively the fastest method to edit a word-document, and although participants valued efficiency in computer use and indicated keyboard shortcuts to be the fastest method, they rather preferred the icon toolbar.

**Habit Formation**

The possibility of software customization and keyboard shortcut usage have been around for quite a while, ever since though, users tended to preserve rigid patterns of use instead of striving for more efficient user behavior (Mackay, 1991). Since customization and learning keyboard shortcuts cost time and effort that could alternatively be spent working, the retention seems reasonable (Carroll & Rosson, 1987). Furthermore, because people cannot anticipate the most efficient solution in the long run due to limited time, information and cognitive capabilities (Simon, 1996), it is reasonable to pick up the just ‘good enough’ method in the first place. Once a sufficient but not necessarily ideal path (e.g., navigation through a traditional menu) has led to the desired outcome, people are likely to repeat past behavior frequently, which then results in habit formation (Ouellette & Wood, 1998). Habits enable automatic decision-making, which saves cognitive resources and does not impair performance under stable conditions. On the other hand, when conditions change (e.g., in the case of repetitive tasks) and an adoption of behavior would be more appropriate, people might preserve their used patterns even at the cost of efficiency (Zimbardo, McDermott, Jansz, & Metaal, 1995).

**Behavior Change**

Fogg (2009), however, proposed a behavior change model stating that any behavior is changeable, provided a person is (1) sufficiently motivated, (2) has the ability to perform the behavior change and (3) is triggered to perform the behavior change. Motivation, ability and triggers; all three factors must be present at the same time or a behavior change would not occur. Motivation and ability must be at least somewhat above zero-level. Furthermore, these two factors can be trade-offs; if high on ability, low motivation is sufficient for a behavior change to be triggered. Or inversely, if low on ability, high motivation is needed to increase the chance of a behavior change to be triggered. Set in the context of software, motivation could, for example, depend on the type of task a user has to perform (e.g., a lot of repetitive commands), and the ability on the difficulty level of learning a specific keyboard shortcut (e.g., easy to learn). So, what are triggers then?

**Triggers for a Behavior Change.** Generally speaking and according to Fogg,

> a trigger can take many forms – an alarm that sounds, a text message, (…) and so on. Whatever the form, successful triggers have three characteristics: First, we notice the trigger. Second, we associate the trigger with a target behavior. Third, the trigger happens when we are both motivated and able to perform the behavior. (2009, p. 3)

Mackay (1991) conducted interviews with MIT employees and identified numerous triggers that are likely to facilitate enhanced user behavior. Some of the triggers, like 'switching environment', are subject to external influences and are therefore difficult to implement in software. Other triggers though, like 'Increasing the awareness of the customization feature', are convertible into software features. In fact, Banovic et al. (2012) successfully used system notifications in a laboratory experiment to increase the awareness of customization features and to trigger more customization behavior. Yet, it remains to be clarified whether such a trigger could also encourage keyboard shortcut use.

**Timing is Critical.** Unlike Banovic et al. (2012), where system notifications were displayed between task blocks (rest time), research suggests that software features should be integrated directly into the user's task flow (Grossman et al., 2007), which is also more likely to cope with real-world scenarios. Rather than flooding the user with random triggers when the user is performing a task, the software waits for the opportune moment, which is when a user perceives a behavior alteration as appropriate. This is the philosophy of adaptive user interfaces, which assist users in customizing software based on the user's command history (Kurlander & Finer, 1992; Tsandilas & Schraefel, 2005). However, since other HCI research advocates the optimal method to be taught before rigid habits are formed (e.g., Aarts & Dijksterhuis, 2000), an early and a delayed trigger were compared against each other in this study.

**Aims and Expectations of this Study**

The aim of this study was to find whether system notifications could be used as triggers to facilitate a transition to enhanced user behavior in terms of using more keyboard shortcuts. An early and a delayed deployment of system notifications were compared against each other and against a control group without any sort of trigger.

It was expected that an early system notification will leverage more keyboard shortcuts than a delayed trigger, and a delayed trigger more than no trigger. According to this pattern, task performance and perceived usability should be best in the case of an early system notification, followed by delayed trigger and no trigger.

# Method

## Experimental Design

The between-subject independent variable was trigger (system notification) with either an early or delayed point of initiation of the trigger. A control group received no trigger at all. The dependent variables were the objective measures keyboard shortcut use, task completion time and task completion score, and the subjective measures perceived usability. A second measure was carried out in order to check for long-term effects.

## Participants

Participants were partially recruited via a faculty-internal participants database (Department of Psychology, University of Basel) and partially via invitations distributed on Facebook and the online-marketplace of the University of Basel. Participants received an equivalent of 15 US$ or course credits for compensation.

A total of 78 participants took part in both parts of the study. Two participants were excluded from analysis, as they did not interact with the application properly. One participant reported technical problems and another participant did not follow the instructions as tracking data revealed. The final sample consisted of 76 participants (49 females, $M = 25.6$ years, $SD = 8.6$, range = 18-56). According to ratings on a 4-point Likert scale (1 = no experience; 4 = expert), participants were experienced computer ($M = 2.9$, $SD = 0.5$) and internet users ($M = 3.1$, $SD = 0.4$). Across all experimental groups participants did not differ significantly regarding age, computer and internet experience.

**Materials**

A general-purpose application was used, so that no prior knowledge from participants was required. The application was a personal calendar website developed for this particular study (Figure 2). Developing a new application not only allowed us to track all sorts of user inputs, but also to focus on the essential features needed for this experiment. All computers in the computer room at the Department of Psychology Study were Mac Intel Core 2 Duo, OS 10.6 with standard Apple keyboards and standard Apple mice for data input. The application was run in Google Chrome (version 27.0).

**Personal calendar application.** The whole application consisted of nothing more than a single big screen as shown in Figure 2. All commands and information needed to complete the tasks were accessible through this screen. At the top of the screen in the header, the calendar month that the participant was currently working on was displayed. Below the header, a grid was visible, showing all days of the calendar month.

**Four calendar tasks.** All participants were expected to assign events to specific days of the months *September*, *October*, *November* and *December*. Assigning an event-box to a calendar day was as easy as double-clicking and dragging an empty event-box on the right side of the screen and dropping it onto the desired day in the grid. By accessing the menu in the upper left corner 'Termin >', the participant could add, edit or delete event-boxes.
An event-reference-table on the right side of the application screen provided all information needed (such as day, time, title and location) to fill in the calendar with multiple predetermined events (see appendix for further information). The user had to type all information into event-boxes, since it was not possible to copy text from the event-reference-table into the clipboard. In the lower right corner of the screen a completed event-box was visible during the whole task, so that the user would know how information should be entered (e.g., whether text should be in brackets or not).

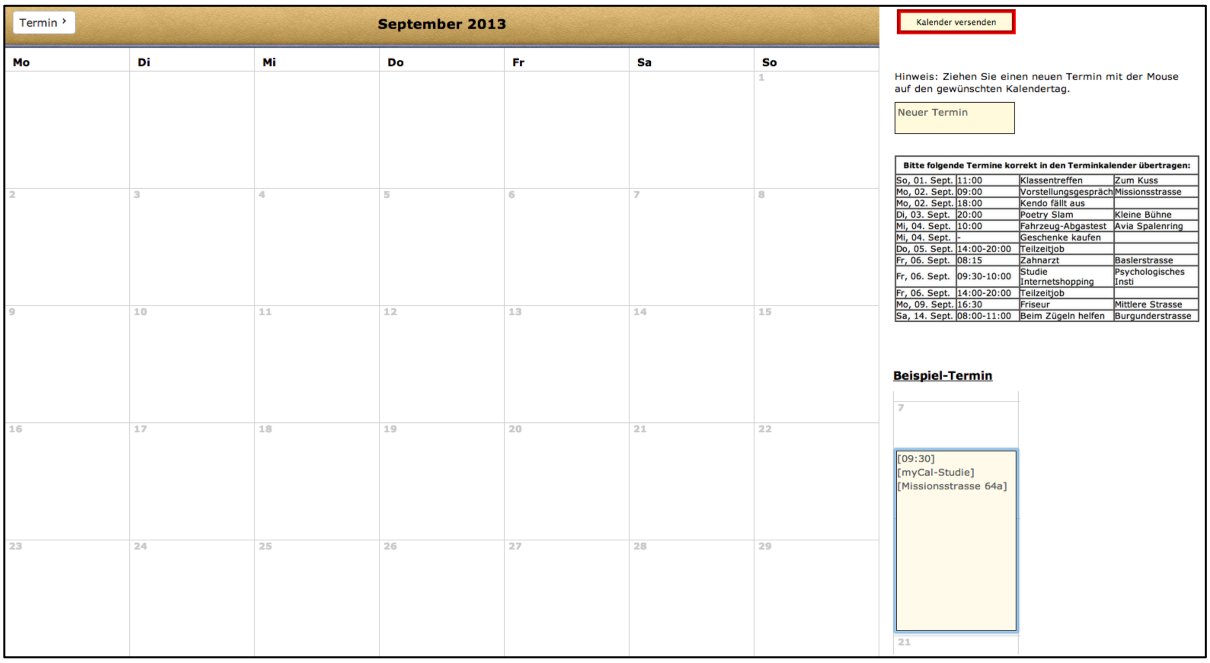


*Figure 2*. Screenshot of task 1 (September) in the personal calendar application.

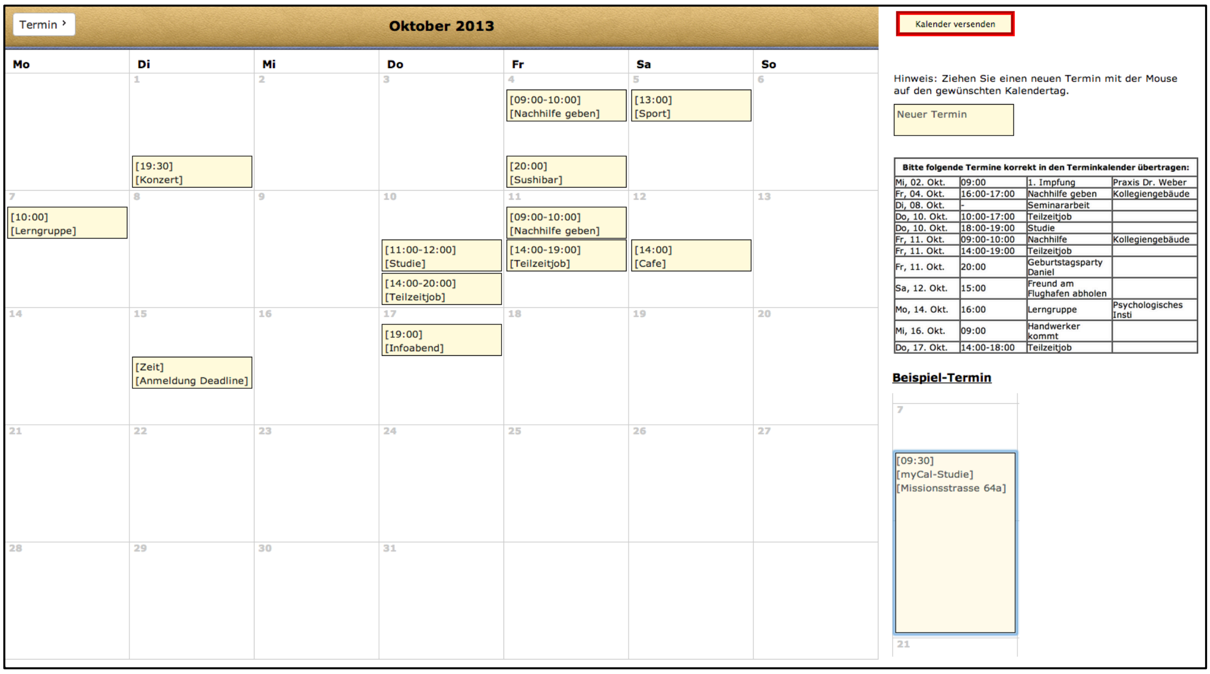


*Figure 3*. Screenshot of task 2 (October) with preloaded events.

As opposed to task 1, where the grid was empty at startup, some events were already preloaded into the grid in task 2 to simulate advanced application use (Figure 3). The user's objective was to adjust the calendar in a way that the calendar would only contain the events listed in the event-reference-table. This could either be done by editing preloaded events or by deleting and replacing preloaded events with new ones.

Similar to task 1 and 2 in the first part of the study, in the second part task 3 started off with an empty grid, followed by task 4, where the participant had to adjust preloaded events.

**Keyboard shortcuts.** Instead of frequently accessing the menu, participants from all experimental groups were free to use keyboard shortcuts, which were indicated in parentheses next to the corresponding menu commands (Figure 4).

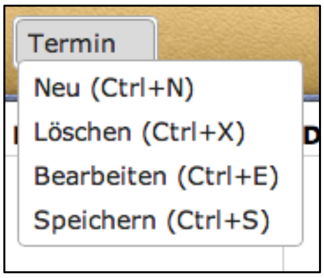


*Figure 4*. Menu commands with corresponding keyboard shortcuts in parentheses.

**Triggers and timing.** For the experimental group 'early trigger' system notifications were displayed during task 1, and for experimental group 'delayed trigger' during task 2, respectively (for an overview of experimental procedure refer to Figure 1). Once a task was launched the notifications were triggered based on the participants' command history; after the fifth time a user performed the menu command 'New (Ctrl+N)' in order to create a new event, the notification would say: "*Tip: You can also use the keyboard shortcut CTRL+N to create a new event!*" (Figure 5). After the fifth time a user performed the menu command 'Edit (Ctrl+E)' another notification would show up, but this time saying: "*Tip: You can also use the keyboard shortcut CTRL+E to edit an event!*" Both notifications stopped automatically after seven seconds. The chance of both notifications being triggered at the same time was considerably small.

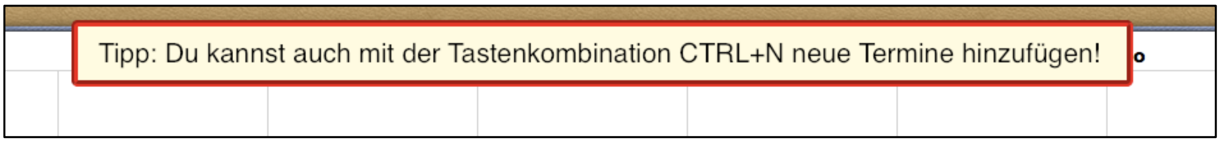


*Figure 5*. A system notification promoting the use of keyboard shortcut 'CTRL+N'.

## Measurement Of Variables

**Covariates.** Demographics, experience with computer, internet and keyboard shortcuts was assessed through non-standardized items in post-questionnaires. In order to control different levels of keyboard shortcut experience, participants were either classified as power or non-power users based upon post questionnaire data. If participants had an average rating higher than 'three' across five items with a six-point Likert scale measuring keyboard shortcut experience (1 = very rarely; 6 = almost always), they were classified as power users, otherwise they were classified as non-power users (see appendix for further information). This classification is similar to the one done by Banovic et al. (2012), who accounted for different levels of customization experience.

**User performance.** The application tracked task completion time, event-box entries and menu and keyboard shortcut commands. Event-box entries were rated by the author based on a non-standardized scoring system (see appendix for further information).

**Perceived usability.** The System Usability Scale (SUS) (Brooke, 1996) was used to assess the application's usability.

**Procedure**

All study participants were asked to come to a computer room at the Department of Psychology, where computers and peripherals were identical. The experimenter randomly assigned the participants to one of three conditions; early trigger, delayed trigger and no trigger (Figure 1). The experimenter started a personal calendar application and did not give any information about the task. Instead, the participants were asked to perform the tasks as described in the instructions on the screen. After participants completed task 1 and 2 by assigning multiple events to calendars, the first part of the study finished with an evaluation of the calendar application and questions about demographics, computer and internet experience.

After a break of one week, participants came to the computer room a second time to complete task 3 and 4, again by assigning multiple events to calendars. Because no triggers were involved in the second part, all three conditions were exactly the same. Finally, the experiment concluded with questions about keyboard shortcut use preferences. For an overview of the experimental procedure see Figure 1.

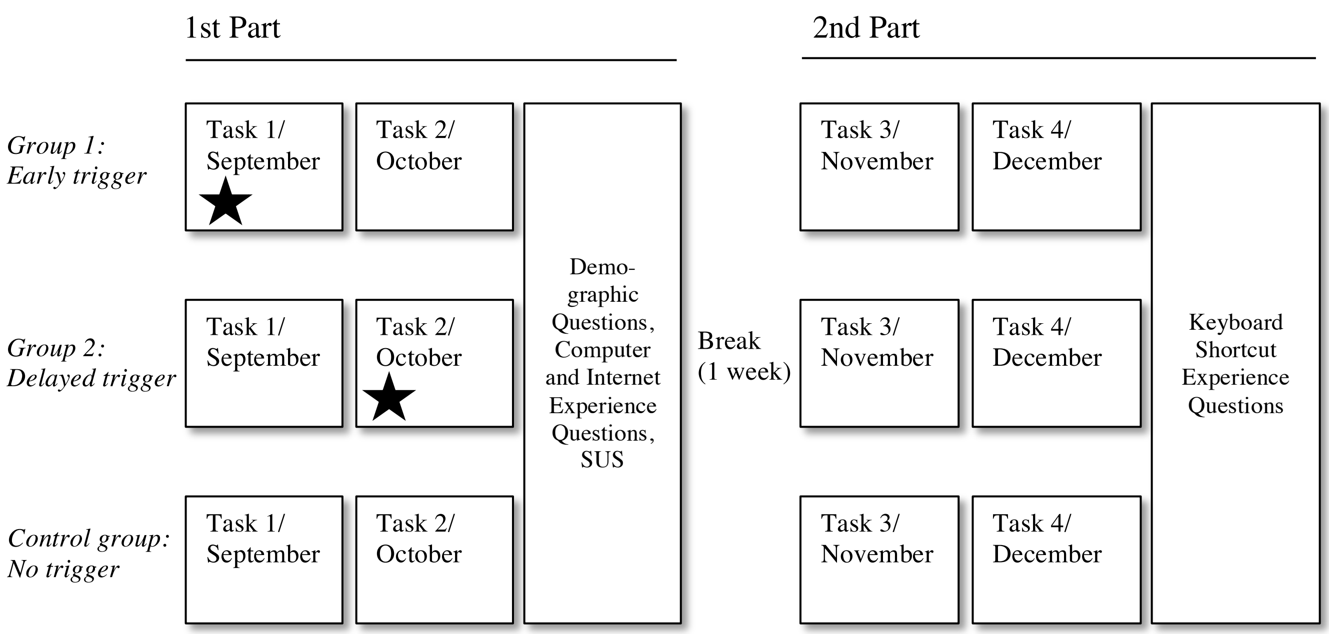


*Figure 1*. Overview of the experimental procedure. Triggers are marked with a star.

# Results

For all statistical tests an alpha level of .05 was used. All data were checked to see if they met the required conditions for the statistical tests. Differing sample sizes are due to removed outliers.

## Covariates

One-way ANOVAs revealed no significant differences between the experimental groups regarding age distribution, computer knowledge, and keyboard shortcut experience. Furthermore, the experimental groups did not differ significantly regarding task motivation and perceived difficulty of the task. The distribution of Mac and Windows users was balanced across experimental groups.

## Keyboard Shortcut Use

Descriptive data show that participants in the experimental group ‘early trigger’ used keyboard shortcuts the most, with a mean of 66.6 across all four tasks, followed by the group ‘delayed trigger’ with a mean of 40.0. The control group used the fewest shortcuts with 34.5 in average (Table 1). Because there was a wide range between participants regarding the total amount of commands (minimum = 89, maximum = 208), a more comparable keyboard shortcut ratio ‘SC ratio’ (number of keyboard shortcuts divided by total number of commands) was calculated for each participant. A Kruskal-Wallis test was conducted to evaluate differences among the three experimental groups on median change in keyboard shortcut ratio. The test was significant $H(2) = 6.60$, $p = 0.03$. Mann Whitney tests for pairwise comparison indicated that the ratio in the group ‘early trigger’ was significantly greater than the ratio in group ‘delayed trigger’ ($z = 1.91$, $p = 0.02$) and control group ($z = 2.39$, $p = 0.01$). No significant difference in ratio was found between group ‘delayed trigger’ and control group ($z = 0.61$, $p = 0.27$). Because the distributions for each group were not identically shaped and the test results were therefore questionable, a median test was also conducted, which did not reveal any group differences for ‘SC Ratio’ $\chi^2(2, N = 76) = 2.94$, $p = .27$.

Table 1

*Descriptive statistics: Keyboard shortcut use for all experimental groups, by number of SC user and number of commands*

| | | Number of SC user | | Number of SC and Menu commands | | |
|---|---|---|---|---|---|---|
| Experimental groups | *n* | *n*[a] | Percent | SC M (SD) | Menu M (SD) | SC Ratio[b] |
| Early trigger | 25 | 16 | 64% | 66.6 (51.6) | 53.7 (50.1) | 0.54 |
| Power user | 13 | 10 | 77% | 84.1 | 36.1 | 0.68 |
| Non-power user | 12 | 6 | 50% | 47.6 | 72.6 | 0.39 |
| Delayed trigger | 25 | 11 | 44% | 40.0 (50.2) | 81.3 (49.3) | 0.34 |
| Power user | 11 | 7 | 64% | 56.8 | 62.5 | 0.49 |
| Non-power user | 14 | 4 | 29% | 26.9 | 96 | 0.23 |
| Control group | 26 | 10 | 38% | 34.5 (50.3) | 86.2 (49.6) | 0.27 |
| Power user | 12 | 4 | 33% | 28 | 86.6 | 0.21 |
| Non-power user | 14 | 6 | 43% | 40 | 85.9 | 0.31 |

*Note:* No significant results found for all dependent variables. SC = Keyboard shortcut.
[a] Number of SC user with a higher SC Ratio[b] than .25.
[b] SC Ratio was calculated by dividing the total number of SC by the total number of commands.

To examine the influence of prior shortcut use preferences, users were classified as power users and non-power users based upon post-questionnaire data. In total, there were 36 power and 40 non-power users, relatively balanced across experimental groups (Table 1). Participants with a SC ratio higher than .25 were rated as keyboard shortcut users.

Within the experimental group ‘early trigger’, 10 out of 13 power users were keyboard shortcut users, whereas 7 out of 11 within the group ‘delayed trigger’ but only 4 out of 12 within the control group were keyboard shortcut users. Fisher’s exact test gave a hint about experimental group differences among power users in the number of shortcut users ($p$ = .08). No tendency of group differences was found among non-power users ($p$ = 0.6).

Further tests revealed that there were not significantly more shortcut users among power users than non-power users in the 'early trigger' group ($p$ = .02), the 'delayed trigger' group ($p$ = 0.1), or in the control group ($p$ = 0.7).

## Task Performance

**Task completion time.** Table 2 shows the descriptive data for task completion time. It took participants from all experimental groups the longest to complete task 1, which is probably due to initial efforts to become accustomed to a new application. On average, participants got gradually faster when performing the subsequent tasks 2, 3 and 4.

A Kruskal-Wallis test was performed to compare group differences, as normal distribution could not be achieved by transformation. No significant task completion time differences between experimental groups were found, $H(2) = 1.52$, $p = 0.47$.

No correlation between task completion time and keyboard shortcut use was found.

Table 2

*Descriptive statistics: Task completion time for experimental groups, by task*

| | | | | | Time in Minutes |
|---|---|---|---|---|---|
| | Task 1/ September | Task 2/ October | Task 3/ November | Task 4/ December | All tasks |
| Experimental groups | M (SD) | M (SD) | M (SD) | M (SD) | M (SD) |
| Early trigger ($n$ = 24) | 12.3 (2.7) | 8.4 (1.6) | 6.8 (1.3) | 6.1 (1.4) | 33.6 (5.6) |
| Delayed trigger ($n$ = 25) | 13.2 (3.5) | 9.0 (2.3) | 7.5 (2.1) | 6.3 (1.9) | 36.0 (7.3) |
| Control group ($n$ = 25) | 12.6 (2.6) | 8.2 (1.6) | 7.1 (1.4) | 6.2 (1.3) | 34.1 (5.2) |

**Task completion score.** As shown in Table 3, all experimental groups almost reached the total maximum score of 92 points. Since few mistakes were made when participants assigned events to calendars and a ceiling effect occurred, no link between keyboard shortcut use and task performance score could be found.

Table 3

*Descriptive statistics: Task completion score for experimental groups, by task*

| | Score | | | | |
|---|---|---|---|---|---|
| | Task 1/ September | Task 2/ October | Task 3/ November | Task 4/ December | All tasks |
| Experimental groups | M | M | M | M | M |
| Early trigger ($n$ = 25) | 23.8 | 22.8 | 21.8 | 21.6 | 90.0 |
| Delayed trigger ($n$ = 25) | 23.7 | 22 | 22 | 21.4 | 89.1 |
| Control group ($n$ = 26) | 23.4 | 23 | 21.3 | 22 | 89.8 |

*Note:* 24 points were the maximum score in task 1 & 2, 22 in task 3 & 4.

## Perceived Usability

Usability of the personal calendar application was rated similar across experimental groups; on the SUS score range of 0 to 100, group ‘early trigger’ rated the tool with a mean score of 58.4 ($SD$ = 18.4), group ‘delayed trigger’ on average with 62.6 ($SD$ = 15) and control group with 55.9 ($SD$ = 18.4). The means did not differ significantly between experimental groups, $F(2, 73) = 0.97, p = 0.38$. It was further examined whether usability ratings varied based on keyboard shortcut use, but no differences between keyboard shortcut users and non-users were found either $F(1, 74) = 2.67, p = 0.11$.

## Discussion

Despite no significant group differences, results do indicate that a simple prompt like a system notification can encourage users to use more keyboard shortcuts. System notifications, regardless of timing, caused more keyboard shortcut usage in contrast to a control group. When timing is considered, early system notifications seemed to have a greater impact than the delayed counterpart. This finding supports the position of research stating that the optimal method should be taught as early as possible (Tak, 2007; Aarts & Dijksterhuis, 2000). However, according to Fogg's behavior change model (2009), the delayed trigger being less efficient in this study could simply be linked to a lack of motivation and ability. When participants in the group 'delayed trigger' received the notification in task 2, they probably knew half of the first part of the study was already done. *Motivation* might have suffered. Additionally, confronted with a slightly new scenario in task 2, where the task objective consisted of adjusting a grid with preloaded events, *ability* might have diminished as well.

Participants who were more responsive to triggers in this study were power users rather than non-power users. Among power users, the greatest number of keyboard shortcut users was in the group 'early trigger', followed by 'delayed trigger' and the smallest in the control group. Among non-power users no systematic behavior alteration pattern related to triggers could be found. When referring to Fogg's behavior change model (2009), this finding suggests that learning new keyboard shortcuts was probably too difficult for non-power users, since no differences were found regarding task motivation in post-questionnaire data.

Keyboard shortcuts' most obvious benefit is attributed to time efficiency (Grossman et al., 2007; Lane et al., 2005; Malacria et al., 2013). Several factors in this study might have led to the outcome of no time saving for shortcut users. Since a lot of time was spent typing, typing speed might have had a large effect on task completion time. Although the amount of steps and the quantity of available commands in order to complete the tasks were channeled by the experiment, the amount of total commands varied between participants, suggesting that to some extent individual working patterns might have evolved, which is reflected in different task completion times. And finally, participants were not explicitly pushed for time.

No time pressure might also be the reason for the ceiling effect in the task completion score. Most of the participants almost reached the maximum score. Accordingly, participants rated task difficulty on a six-point Likert scale (1 = very difficult, 6 = very easy) as being rather easy ($M = 4.8$, $SD = 1.1$). When pushed for time, participants might have had favored speed at the cost of accuracy (Grossman et al., 2007), and keyboard shortcut use might have made a difference.

Perceived usability in this study was independent from keyboard shortcut use. There are other studies where improved efficiency did not lead to higher perceived usability (e.g., Nielsen & Levy, 1994). But since there was no gain in performance between groups in this study, it is possible that no gain in efficiency was perceived either. After all, software limitations reported by participants seemed to have had a much greater impact on the overall SUS rating than keyboard shortcut aspects.

**Limitations and Future Research**

Whereas in real-world scenarios the benefit of using keyboard shortcuts might emerge over a long period of time, there is not so much time in an experiment. To find a stronger effect, time pressure might have been one technique to reveal distinguished patterns in task completion time and task completion errors, as it was done in other studies (e.g., Banovic et al., 2012). However, strict time pressure or a higher amount of repetitive tasks might not reflect common real-world scenarios.

System notifications were triggered after a menu command was executed by a mouse-click and then immediately displayed below the header of the application. This was done to make sure the participant would see the trigger. It is not clear though, whether all participants really saw the notifications. In fact, some participants in the group ‘delayed trigger’ reported that they did not see any triggers. While these statements could have been caused by a cued recall, since there was one week between a short trigger and the questionnaire, future research might address this issue by using eye tracking.

Apart from ‘Increasing the awareness of a customization feature’ there are still other factors identified by Mackay (1991) that remain to be tested as triggers in software.

**Conclusions**

Keyboard shortcuts are underused, which is why several studies made suggestions on how to encourage keyboard shortcut use (e.g., Bailly et al., 2013, Grossman et al., 2007; Malacria et al., 2013). This study presented an alternative method. Participants used more keyboard shortcuts in response to a system notification, even when not explicitly prompted to work fast and efficiently. In addition to that, they interacted with a newly developed application, where a system notification was shown only once for a few seconds. The easy implementation of such a trigger makes the system notification a simple and effective tool to encourage keyboard shortcut use.

**References**

Aarts, H., & Dijksterhuis, A. (2000). Habits as knowledge structures: Automaticity in goal-directed behavior. *Journal of Personality and Social Psychology, 78*, 53-63. doi: 10.1037/0022-3514.78.1.53

Bailly, G., Pietrzak, T., Deber, J., & Wigdor, D. J. (2013). Métamorphe: Augmenting hotkey usage with actuated keys. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ACM, 563-572. doi: 10.1145/2470654.2470734

Banovic, N., Chevalier, F., Grossman, T., & Fitzmaurice, G. (2012). Triggering triggers and burying barriers to customizing software. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ACM, 2717-2726. doi: 10.1145/2207676.2208666

Brooke, J. (1996). SUS - A quick and dirty usability scale. Usability evaluation in industry, 189–194.

Card, S. K., Moran, T. P., & Newell, A. (1980). The keystroke-level model for user performance time with interactive systems. *Communications of the ACM, 23*(7), 396-410. doi: 10.1145/358886.358895

Carroll, J. M., & Rosson, M. B. (1987). *Paradox of the active user*. Cambridge: The MIT Press.

Fogg, B. J. (2009). A behavior model for persuasive design. In *Proceedings of the 4th international Conference on Persuasive Technology*, ACM, 40. doi: 10.1145/1541948.1541999

Gajos, K. Z., Czerwinski, M., Tan, D. S., & Weld, D. S. (2006). Exploring the design space for adaptive graphical user interfaces. In *Proceedings of the working conference on Advanced visual interfaces*, ACM, 201-208. doi 10.1145/1133265.1133306

Grossman, T., Dragicevic, P., & Balakrishnan, R. (2007). Strategies for accelerating on-line learning of hotkeys. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, ACM, 1591-1600. doi 10.1145/1240624.1240865

Kurlander, D., & Feiner, S. (1992). A history-based macro by example system. In *Proceedings of the 5th annual ACM symposium on User interface software and technology*, ACM, 99-106. doi: 10.1145/142621.142633

Lane, D. M., Napier, H. A., Peres, S. C., & Sándor, 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. doi: 10.1207/s15327590ijhc1802_1

Mackay, W. E. (1991). Triggers and barriers to customizing software. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, ACM, 153-160. doi: 10.1145/108844.108867

Malacria, S., Bailly, G., Harrison, J., Cockburn, A., & Gutwin, C. (2013). Promoting Hotkey use through rehearsal with ExposeHK. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ACM, 573-582. doi: 10.1145/2470654.2470735

Nielsen, J. (1992). Finding usability problems through heuristic evaluation. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, ACM, 373-380. doi: 10.1145/142750.142834

Nielsen, J. (2012, January 4). Usability 101: Introduction to usability. *Nngroup.com*. Retrieved November 15, 2013, from http://www.nngroup.com.

Nielsen, J., & Levy, J. (1994). Measuring usability: preference vs.performance. *Communications of the ACM, 37*(4), 66-75. doi: 10.1145/175276.175282

Ouellette, J. A., & Wood, W. (1998). Habit and intention in everyday life: the multiple processes by which past behavior predicts future behavior. *Psychological bulletin, 124*(1), 54-74. doi: 10.1037/0033-2909.124.1.54

Shneiderman, B. (2003). Promoting universal usability with multi-layer interface design. In *proceedings of the 2003 Conference on Universal Usability Computers and the Physically Handicapped,* 1-8. doi: 10.1145/960201.957206

Simon, H. A. (1996). *The sciences of the artificial*. Cambridge, USA: MIT press.

Tak, S. (2007). *The use of keyboard shortcuts: Optimizing versus satisficing in the use of complex technology*. Master's thesis, Eindhoven University of Technology, Netherlands.

Tsandilas, T. & Schraefel, (2005). An empirical assessment of adaptation techniques. In *CHI'05 Extended Abstracts on Human Factors in Computing Systems,* ACM, 2009-2012. doi: 10.1145/1056808.1057079

Zimbardo, McDermott, M., Jansz, J., & Metaal, N. (1995). Psychology: a european text. London: HarperCollins.

## Appendix

Four event-reference-tables. All information needed to be typed into event-boxes by the participant.

*Task 1/ September (12 events)*

| | | | |
|---|---|---|---|
| So, 01. Sept. | 11:00 | Klassentreffen | Zum Kuss |
| Mo, 02. Sept. | 09:00 | Vorstellungsgespräch | Missionsstrasse |
| Mo, 02. Sept. | 18:00 | Kendo fällt aus | |
| Di, 03. Sept. | 20:00 | Poetry Slam | Kleine Bühne |
| Mi, 04. Sept. | 10:00 | Fahrzeug-Abgastest | Avia Spalenring |
| Mi, 04. Sept. | - | Geschenke kaufen | |
| Do, 05. Sept. | 14:00-20:00 | Teilzeitjob | |
| Fr, 06. Sept. | 08:15 | Zahnarzt | Baslerstrasse |
| Fr, 06. Sept. | 09:30-10:00 | Studie Internetshopping | Psychologisches Insti |
| Fr, 06. Sept. | 14:00-20:00 | Teilzeitjob | |
| Mo, 09. Sept. | 16:30 | Friseur | Mittlere Strasse |
| Sa, 14. Sept. | 08:00-11:00 | Beim Zügeln helfen | Burgunderstrasse |

*Task 2/ October (12 events)*

| | | | |
|---|---|---|---|
| Mi, 02. Okt. | 09:00 | 1. Impfung | Praxis Dr. Weber |
| Fr, 04. Okt. | 16:00-17:00 | Nachhilfe geben | Kollegiengebäude |
| Di, 08. Okt. | - | Seminararbeit | |
| Do, 10. Okt. | 10:00-17:00 | Teilzeitjob | |
| Do, 10. Okt. | 18:00-19:00 | Studie | |
| Fr, 11. Okt. | 09:00-10:00 | Nachhilfe | Kollegiengebäude |
| Fr, 11. Okt. | 14:00-19:00 | Teilzeitjob | |
| Fr, 11. Okt. | 20:00 | Geburtstagsparty Daniel | |
| Sa, 12. Okt. | 15:00 | Freund am Flughafen abholen | |
| Mo, 14. Okt. | 16:00 | Lerngruppe | Psychologisches Insti |
| Mi, 16. Okt. | 09:00 | Handwerker kommt | |
| Do, 17. Okt. | 14:00-18:00 | Teilzeitjob | |

*Task 3/ November (11 events)*

| | | | |
|---|---|---|---|
| Sa, 02. Nov. | 16:00 | Stadtlauf Training | Barfi |
| Mo, 04. Nov. | 10:00 | Lerngruppe | UB |
| Mi, 06. Nov. | - | IKEA | Pratteln |
| Mi, 06. Nov. | 17:00 | Friseur | |
| Fr, 08. Nov. | 09:00-14:00 | Teilzeitjob | |
| Sa, 09. Nov. | 16:00 | Stadtlauf Training | Barfi |
| Mo, 11. Nov. | - | Auto mieten | |
| Mi, 13. Nov. | 16:30-17:00 | Studie | Missionsstrasse |
| Do, 14. Nov. | - | Handyvertrag | |
| Fr. 15. Nov. | 13:30-17:00 | Teilzeitjob | |
| Sa. 16. Nov. | 16:00 | Stadtlauf Training | Barfi |

*Task 4/ December (11 events)*

| | | | |
|---|---|---|---|
| Di, 03. Dez. | 11:00 | 2. Impfung | Praxis Dr. Weber |
| Di, 03. Dez. | - | Seminararbeit | |
| Mi, 04. Dez. | 17:00 | Fahrrad | |
| Do, 05. Dez. | 18:00 | Weihnachtsessen | |
| Fr, 06. Dez. | - | Schlüssel abgeben | |
| Fr, 06. Dez. | 10:00-17:00 | Teilzeitjob | |
| So, 08. Dez. | 18:00 | Date | Barfi |
| Mi, 11. Dez. | 10:30 | Computerraum | |
| Mi, 11. Dez. | 16:00 | Bandprobe fällt aus | |
| Do, 12. Dez. | 10:00 | Präsentation | |
| Fr, 13. Dez. | 16:00-17:00 | Nachhilfe geben | Kollegiengebäude |

## Appendix

Every event-box entry was rated by the author based on a scoring system as shown in Table 4. The maximum task scores were: September (24), October (24), November (22), December (22).

Table 4

*Task completion scoring system.*

| Points | Criterion |
|---|---|
| 2 | All information correct |
| 1 | Information missing |
| 0 | Event is missing |
| -1 | Preloaded event was not modified or deleted |

# Appendix

Five post-questionnaire items used to classify participants as power or non-power users. All items had the same six-point rating scale (1 = very rarely; 6 = almost always).

*Question*: “Bitte schätzen Sie, in welchem Masse Sie in den folgenden Gebieten Keyboard Shortcuts anwenden”

- Desktopanwendungen (z.B. Microsoft Word)

  *1 = in sehr geringem Masse; 6 = in höchstem Masse*

- Surfen im Internet

  *1 = in sehr geringem Masse; 6 = in höchstem Masse*

- Social Network Systems (z.B. Facebook)

  *1 = in sehr geringem Masse; 6 = in höchstem Masse*

- E-Mail

  *1 = in sehr geringem Masse; 6 = in höchstem Masse*

- Computerspiele

  *1 = in sehr geringem Masse; 6 = in höchstem Masse*