Snookered by an interruption? Use a cue

Stuart A Jones
UCL Interaction Centre,
8th Floor MPEB, Malet Place
UCL, London, WC1E 6BT
stu@gingerbbm.com

Sandy J J Gould
UCL Interaction Centre
8th Floor MPEB, Malet Place
UCL, London WC1E 6BT
s.gould@cs.ucl.ac.uk

Anna L Cox
UCL Interaction Centre,
8th Floor MPEB, Malet Place
UCL, London, WC1E 6BT
anna.cox@ucl.ac.uk

When routine tasks are interrupted, erroneous slips become more likely. Expertise is no defence against these kinds of errors but visual hints can alleviate such negative effects in computer interfaces. We compared previous-action cueing with next-action cueing, measuring the effects on error rate, and found that both approaches were statistically equivalent in helping to mitigate the disruptive effects of interruptions. Following an interruption, a cue should be displayed highlighting the last action performed by the user – a trivial operation for software applications.

Interruptions, Cueing, Disruption mitigation

1. INTRODUCTION

This study investigates a method of mitigating the effects of interruptions on the performance of computer-based routine procedural tasks. It is an important field of work: a secondary activity may be trivial in nature but its intrusiveness could have significant negative effects on the primary task being undertaken. In some cases the consequences can be catastrophic. Wickens and McCarley (2008) cite two aviation examples where interruptions led to a loss of life. In one, an air-traffic controller is distracted after having positioned an aeroplane on a runway. He fails to return his attention to the primary task, leading to a fatal crash with another aeroplane that had been cleared to land. In another example, a pilot is interrupted during the take-off procedure. Upon resuming the primary task he commits an anticipation error, skipping a vital step that results in over a hundred deaths. Many other examples have been documented in the literature (see Latorella, 1999).

In addition to safety-critical industries (such as nuclear power generation or healthcare) where the deleterious effects of interruptions can be severe, interruptions also have an effect in the office environment. People are using computers for long periods of time on a daily basis and are finding themselves increasingly disrupted by incoming emails, updates and tweets – what Bailey and Konstan (2006) term a "burgeoning epidemic of interruption at the user interface". Juggling tasks is something people tend to do well, but interruptions have been shown adversely to reduce performance and should therefore be subject to investigation. In so doing, we can better understand their impact, develop cognitive models to predict their consequences, and refine strategies to mitigate their effects (Bailey and Konstan, 2006).

Various approaches for mitigating the problems caused by interruptions have been investigated. For post-completion errors (PCEs) – a particular class of error where a sub-task that should occur after the completion of a main goal is missed – motivation and training have both been found to be ineffective (Back, Cheng, Dann, Curzon, and Blandford, 2006; Byrne and Davis, 2006). However, Chung and Byrne (2008) found that providing a prominent hint in the user interface immediately prior to the post-completion step completely eliminated PCEs. Trafton, Altmann and Brock (2005) studied the effect of cues on the time taken to resume a primary task following an interruption, finding that a highly salient cue had a significant positive effect.

This study expands upon these investigations into the effects of cueing. The work by Trafton et al. (2005) uses *previous-action cueing* (PAC) and concentrates on resumption lag while ignoring sequence errors. In contrast, Chung and Byrne's (2008) investigation uses *next-action cueing* (NAC) and focuses solely on PCEs. Determining the relative efficacy of different types of cueing is an important issue for designers of interrupted systems. Whereas it is non-trivial for a computer program to anticipate the next action in a series of tasks, it is simple to recall the action taken prior to an interruption by using a native loss-of-focus event in computer systems.

Our hypothesis is that PAC results in a significant improvement in error rate over the absence of a cue. Additionally, we hypothesise that NAC is an improvement over PAC. We base this hypothesis on the exceptional results demonstrated by Chung and Byrne (2008) with post-completion errors. We measure error rate by counting the number and type of sequence errors committed following the interruptions.

2. METHOD

The experiment compares three conditions: the control condition (NC) during which no cueing occurs; previous-action cueing (PAC); and next-action cueing (NAC).

2.1 Materials

For this experiment we utilised a software application called the Prescription Machine. It simulates the routine procedural task of compiling various types of medication in a pharmacy. It was developed in Python and was deployed on a PC running Microsoft Windows XP. Experiments were carried out in cubicle rooms containing a single computer to reduce the possibility of external distractions. Figure 1 depicts the main user interface of the Prescription Machine.

The Prescription Machine is similar to the Wicket Doughnut Machine, which has been used extensively in other studies into memory load and human error (e.g. Li, Blandford, Cairns and Young, 2008; Ament, Cox, Blandford and Brumby, 2010; Back, Brumby and Cox, 2010; and Hiltz, Back and Blandford, 2010). Users are required to fulfil an order presented in the screen's central *Prescription sheet* pane by entering the quantities into the various satellite panes. These sub-tasks must be performed in a specific order. To complete a sub-task, the appropriate values must be entered into the sub-task, after which the *OK* button must be clicked. At this point the information entered is reset to zero to prevent the presence of numbers acting as an implicit cue.

Before working on a sub-task participants had to activate it by clicking the corresponding button in the *Selector* pane. The spatial mapping between sub-task pane and sub-task button is purposefully defective in order to increase the effort required to complete a trial successfully.

For this investigation, the Prescription Machine was configured to be able to interrupt a trial between the click of a sub-task's *OK* button and the click of the subsequent sub-task's selector button (or the *Process* button in the case of the last sub-task). There were four opportunities for interruption: after

the *Shape*, *Colour*, *Packaging* and *Label* sub-tasks. Participants were interrupted so as to encourage erroneous behaviour when resuming the trial.
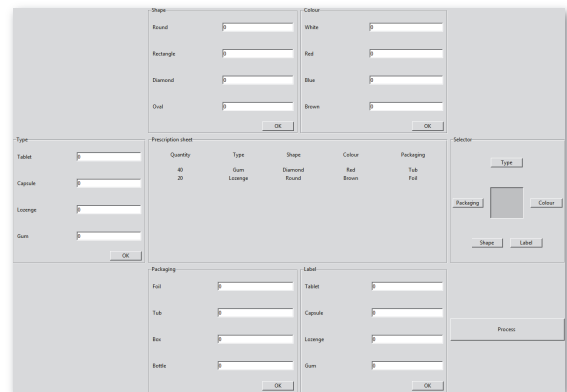

Figure 1: The main interface of the Prescription Machine

An interruption consisted of a modal dialog box occluding the interface of the Prescription Machine for 45 seconds. The interface was hidden to prevent participants using their own visual cues to aid resumption. The enforced interruption period was employed to make it harder for participants to remember what they were doing prior to an interruption. The value of 45 seconds was chosen based on the success of a similar study (Altmann and Trafton, 2004). The dialog box presented a contextual arithmetic question related to packaging. Upon submission of an answer, participants were prompted to answer another question, and so on, in order to keep the load on working memory high whilst being distracted from the primary task. Wrong answers were ignored unless two consecutive questions were answered incorrectly, in which case a warning was provided and the same question was reiterated. This was to prevent participants entering a series of thoughtless responses, without being over-strict.
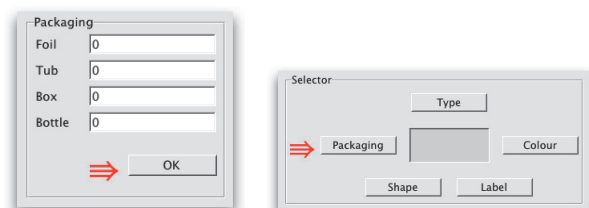

Figure 2: Example PAC (left) and NAC (right) cues

After 45 seconds the interruption dialog box would close. Participants in the NC condition would then proceed unaided to resume the trial. For participants in the other two conditions, a red arrow would be visible adjacent to and pointing at either the previous button clicked (PAC) or the next button to click (NAC). In the former case, this would always be an *OK* button in one of the sub-task panes; in the latter, this would be one of the

buttons in the *Selector* pane or the *Process* button (see Figure 2 for examples). Once the next action had been taken, this cue would disappear from the interface.

The Prescription Machine recorded the number and type of erroneous actions made after interruptions. It also stored the resumption lag—the elapsed time between an interruption dialog closing and a subsequent action being taken.

2.2 Design

A mixed design was chosen in order to allow easier interrogation of the results and to avoid training and fatigue effects in participants. The between-subjects independent variable was the type of cue used, resulting in the three conditions NC, PAC and NAC.

The within-subjects independent variable was the number of interruptions per trial. This was randomised such that, within each batch of three trials, a participant would encounter zero, one or two interruptions per trial. This randomisation was introduced to reduce the possibility of confounds affecting the findings in the form of training effects: the number of interruptions was varied to ensure that once a participant had experienced an interruption within a trial they could not know whether to expect further interruptions.

Each participant performed 21 trials with the Prescription Machine.

When a participant committed an error, a panel in the interface was coloured red and labelled "Error". Participants were instructed that, on experiencing an error, they must determine and perform the correct step to get back into sequence (thus clearing the error) and proceed with the trial. We opted to show an error rather than instruct participants of the next correct step in order to accentuate the cost of an incorrect action, motivating users to concentrate both to avoid mistakes and maintain a high load on working memory.

2.3 Measures

In accordance with our hypothesis we measured the post-interruption dependent variable of error rate. Consistent with Trafton, Altmann and Ratwani (2011) error rates were percentages, calculated by dividing the number of errors by the number of opportunities for error. As with Byrne and Bovair (1997) we used a score of 5% as the threshold for systematicity: an error rate above this value is considered systematic. Following an interruption, only the initial error was counted.

2.4 Participants

A total of 45 participants (26 female) took part in the trials, the majority recruited from the Psychology Subject Pool at University College London. The ages ranged from 18 to 64; the mean age was 27.9 years (*SD* = 9.3 years). Fifteen students participated in each condition, giving their time for the chance of winning a prize of £50, £30 or £20. A certain level of exposure to Microsoft Windows was assumed and no matching was done between conditions. Given that the chosen visual cue used in the Prescription Machine was red in colour, to reduce possible effects of red-green colour blindness only participants without this condition were recruited.

2.5 Procedure

Participants were told that the study was an investigation into the effects of repetition on the performance of routine procedural tasks. This deception was necessary to avoid drawing attention to the real interest of performance related to interruptions.

A demonstration introduced participants to the Prescription Machine. Following this they were invited to carry out several training trials themselves, during which no interruptions occurred.

To introduce the secondary task, participants completed a trial that included two interruptions and a resumption cue corresponding to the condition to which they had been assigned. The instructions given to the participants in the PAC and NAC groups stated that on resuming the primary task after interruption, a visual cue in the form of a large red arrow would be present. The meaning and purpose of this cue was described on an instruction sheet.

Participants were told that they would be completing 21 trials in total, that during the trials they could encounter any number of interruptions, and that there would be an optional two-minute break roughly halfway through the experiment. After the trials had been completed, participants were thanked for their time and debriefed in accordance with The British Psychological Society's guidelines (BPS, 2009, pp. 20-21) as to the true nature of the investigation. At this point, the reason for the subterfuge was made clear.

3 RESULTS

3.1 Baseline error rate

To ascertain that participants could execute the primary task of the Prescription Machine we counted errors in the zero-interruption trials at six occasions: the five selector button steps, plus the *Process* button. That is, if the correct action at each of these occasions was not performed, an error was counted. (Any subsequent errors made while the participant attempted to get back into sequence were ignored.) There were seven such trials per participant, so 42 chances in total to commit such an error. A participant's baseline error rate was therefore defined as: number of errors ÷ 42 × 100.

3.2 Outliers

To identify outliers for removal from the analyses we considered the mean error rate across all participants in the zero-interruption trials. We excluded those whose error rate was greater than the mean plus the product of the standard deviation and 1.96 (Table 1). Data from four participants were excluded in this way, suggesting an inability to follow instructions or learn the task to an acceptable standard. Two outliers were removed from the NC condition; one outlier was removed from each from the PAC and NAC conditions.

Table 1: Error rate information in zero-interruption trials

Mean error rate $\bar{x}$	Standard deviation SD	Upper limit of error rate $\bar{x} + (SD \times 1.96)$
5.50%	5.56%	16.39%

Next, we analysed the average post-interruption error rates in each condition. It was clear that some participants performed particularly poorly so a second pass for outliers was conducted using the same approach as above. Table 2 shows the information which resulted in three further outliers being removed, one from each condition.

Table 2: Post-interruption error rate information

Cue	Mean error rate $\bar{x}$	Standard deviation SD	Upper limit of error rate $\bar{x} + (SD \times 1.96)$
NC	51.28%	22.43%	95.25%
PAC	6.80%	10.68%	27.74%
NAC	3.06%	7.84%	18.42%

3.3 Baseline task performance

Each participant performed a subset of trials when no cueing was encountered (the zero-interruption trials). With the outliers removed, and analysing the data across all conditions, errors were on average made 4.20% (*SD*=4.05%) of the time. Inspecting these data by cue condition shows similar means of 4.76% (*SD*=5.37%), 4.58% (*SD*=4.18%) and 3.30% (*SD*=2.29%) for NC, PAC and NAC respectively. Furthermore, a one-way between-subjects ANOVA demonstrated no significant main effect (F(2, 35) = 0.483, p=0.621), illustrating consistency in performance between the three conditions, as expected.

3.5 Resumption error rate

Focusing on the NC condition we recalculated the error rates according to the opportunities for interruption within one trial. There were four such opportunities as described previously: when the clicking of the *Colour, Packaging* and *Label* selector buttons, or the *Process* button, was not carried out as expected. Thus, there were four chances for sequence errors to be made upon resumption. There were 28 possible occasions in total per participant so the error rate was therefore defined as: number of errors ÷ 28 × 100.

3.7 The effect of cue on error rate

Our hypothesis relates to the impact upon error rates when different types of cueing is utilised. The average error rates in each of the three conditions NC, PAC and NAC were 47.22% (*SD*=17.75%), 4.76% (*SD*=7.78%) and 1.10% (*SD*=2.85%) respectively (see Figure 3). A one-way between-subjects ANOVA on these error rates demonstrated a significant main effect of cue type (F(2, 35) = 66.048, p<0.001). Tukey post-hoc comparisons were used to determine that both PAC and NAC conditions were significantly different to the NC condition (p<0.001 in both cases). No significant difference was found between the PAC and NAC conditions (p=0.697).
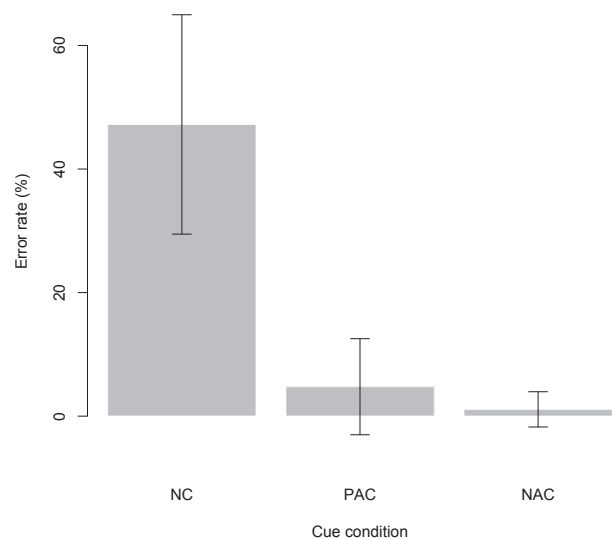

Figure 3: Mean error rates by cue condition

4. DISCUSSION

This experiment investigated how the introduction of a salient, meaningful, just-in-time cue might affect resumption following an interruption to a primary activity. Introducing interruptions into a routine procedural task resulted in a substantial increase in the error rates of participants who had previously gained a satisfactory level of expertise with the Prescription Machine application. By providing a visual hint either to the last action performed or the next step to take, the error rates dropped considerably.

Previous- and next-action cueing both reduced the average error rate to below the 5% systematicity level. Neither cue type completely eliminated errors, so the remarkable results of Chung and Byrne (2008) for post-completion errors—a 0% error rate when the next action was cued—were not wholly replicated for sequence errors. The mean error rates reported support our hypothesis: PAC and NAC both resulted in a dramatic improvement in error rate over the NC condition. Since the error rates for the cue conditions both fell below the systematicity level of 5% the results can be considered equivalent—and hence previous-action cueing can be considered as effective as next-action cueing in reducing error. Statistical tests showed no significant difference between the two cue conditions, strengthening the notion of cue equivalence. In other words, for routine procedural tasks, showing a user what they have just done is as beneficial as telling them the next thing to do when resuming after interruption.

In general, a user's next action in an interface is not necessarily clearly defined. The direct manipulation paradigm of graphical user interfaces encourages tasks to be constructed in a variety of ways using many granular sub-tasks. Since these steps may not be strictly ordered, software applications are unable to second-guess the next action that will be taken. A task that is composed of a sequence of steps to be performed in a specific order could be automated, completely avoiding the negative impacts caused by interruptions. For this reason, Burmistrov and Leonova (1997) suggested that common compound tasks could be encapsulated into single commands, such as combining the sub-tasks for moving a paragraph of text: select, cut, position cursor, paste.

In contrast, it is trivial for a computer program to know the last action taken by a user. Since we have shown cueing both before and after interruptions to be equivalent in their effects, we can recommend that software applications, when regaining the focus after an interruption, add a dynamic cue pointing to the previous step completed by the user. As an example, programs written for the Windows operating system have the ability to detect when they lose and gain focus; assuming an interruption is caused by another application (rather than an external disruption) the implementation of such functionality is straightforward. Norman (2010) echoes the suggestion of the use of cueing, stating that software applications ought to recognise that a user's attention has switched away, and that upon resumption users "will need a quick and easy way to remember just what has been done [and] what is now required".

People exploit cues to reduce cognitive effort (Kool, McGuire, Rosen and Botvinick, 2010) but some evidence suggests that repeated exposure to cues can reduce their effectiveness, and moreover irritate users (Ratwani, McCurry and Trafton, 2008). It has been postulated that users might become over-reliant on cues and thus susceptible to error should a cue fail (Byrne, 2008). But a study that focused specifically on the effects of repeated exposure found no evidence that participants became dependent upon cues (Ament, Lai and Cox, 2011). Cueing, then, can be effective for existing software applications, but more work is required to understand the full implications.

5. CONCLUSION

Reducing the frequency of human error is an important endeavour in the realm of interface design, especially in safety-critical domains. The findings of this study are useful to designers of interrupted systems because they suggest that given the equivalence of next- and previous-action cues, software applications should introduce a salient, meaningful, just-in-time cue pointing towards a user's last action.

6. REFERENCES

Altmann, E. M., & Trafton, J. G. (2004). Task Interruption: Resumption Lag and the Role of Cues. *Proceedings of the 26th Annual Conference of the Cognitive Science Society,* Chicago, IL, 05/08/2004, 42–47. Austin, TX.

Ament, M. G. A., Cox, A. L., Blandford, A., & Brumby, D. (2010). Working Memory Load Affects Device-Specific but Not Task-Specific Error Rates. *Proceedings of the 32nd Annual Conference of the Cognitive Science Society*, Portland, OR, 11/08/2010, 91–96. Austin, TX.

Ament, M. G. A., Lai, A. Y. T. and Cox, A. L. (2011). The Effect of Repeated Cue Exposure on Post-Completion Errors. *Proceedings of the 33rd Annual Conference of the Cognitive Science*

Society, Boston, MA, 20/07/2011, 850–855. Austin, TX.

Back, J., Cheng, W., Dann, R., Curzon, P., & Blandford, A. (2006). Does being motivated to avoid procedural errors influence their systematicity? *People and Computers XX— Engage: Proceedings of HCI 2006*, London, 15/09/2006, 151–157. Swindon, UK.

Back, J., Brumby, D. P., & Cox, A. L. (2010). Locked-out: investigating the effectiveness of system lockouts to reduce errors in routine tasks. *Proceedings of the 28th of the international conference extended abstracts on Human factors in computing systems*, Atlanta, GA, 10/04/2010, 3775–3780. New York, NY.

Bailey, B. P., & Konstan, J. A. (2006). On the need for attention-aware systems: Measuring effects of interruption on task performance, error rate, and affective state. *Computers in Human Behavior*, *22*(4), 685–708.

Burmistrov I. and Leonova A. (1997). Interruptions in the computer aided office work: Implications to user interface design. *Proceedings of the 13th Triennial Congress of the International Ergonomics Association 1997*, Tampere, Finland, 29/07/1997, 77–79. Finnish Institute of Occupational Health, Finland.

BPS. (2009). *Code of Ethics and Conduct: Guidance Published by the Ethics Committee of the British Psychological Society*. British Psychological Society, Leicester.

Byrne, M. D. (2008). Preventing Postcompletion Errors: How Much Cue Is Enough? *Proceedings of the 30th Annual Conference of the Cognitive Science Society*, Austin, TX, 23/07/2008, 351–356. Austin, TX.

Byrne, M. D., & Bovair, S. (1997). A working memory model of a common procedural error. *Cognitive Science*, 21(1), 31–61.

Byrne, M. D., & Davis, E. M. (2006). Task structure and postcompletion error in the execution of a routine procedure. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 48(4), 627.

Chung, P. H., & Byrne, M. D. (2008). Cue effectiveness in mitigating postcompletion errors in a routine procedural task. *International Journal of Human-Computer Studies*, 66(4), 217–232.

Hiltz, K., Back, J., & Blandford, A. (2010). The roles of conceptual device models and user goals in avoiding device initialization errors. *Interacting with Computers*, 22(5), 363–374.

Kool, W., McGuire, J. T., Rosen, Z. B., & Botvinick, M. M. (2010). Decision making and the avoidance of cognitive demand. *Journal of Experimental Psychology: General*, 139(4), 665.

Latorella, K. A. (1999). *Investigating interruptions: Implications for flightdeck performance (NASA/TM-1999-209707)*. NASA Langley Research Center, Hampton, VA.

Li, S. Y. W., Blandford, A., Cairns, P., & Young, R. M. (2008). The effect of interruptions on postcompletion and other procedural errors: An account based on the activation-based goal memory model. *Journal of Experimental Psychology: Applied*, 14(4), 314.

Norman, D. A. (2010). *Living with Complexity*. MIT Press, Cambridge, MA.

Ratwani, R. M., McCurry, J. M., & Trafton, J. G. (2008). Predicting postcompletion errors using eye movements. *Proceeding of the 26th annual SIGCHI conference on Human factors in computing systems*, where, where, when, 539–542. ACM, New York, NY.

Trafton, J. G., Altmann, E. M., & Brock, D. P. (2005). Huh, what was I doing? How people use environmental cues after an interruption. *Human Factors and Ergonomics Society Annual Meeting Proceedings*, 49, 468–472.

Trafton, J. G., Altmann, E. M., & Ratwani, R. M. (2011). A memory for goals model of sequence errors. *Cognitive Systems Research*, 12, 134–143.

Wickens, C. D., & McCarley, J. S. (2008). *Applied attention theory*. CRC Press, Boca-Raton, FL.