

Working memory updating involves item-specific removal

Ullrich K. H. Ecker^{a,*}, Klaus Oberauer^b, Stephan Lewandowsky^{c,a}

^a*School of Psychology, University of Western Australia, 35 Stirling Hwy, Crawley 6009, Australia*

^b*Department of Psychology, University of Zurich, Binzmühlestrasse 14/22, 8050 Zurich, Switzerland*

^c*School of Experimental Psychology, University of Bristol, 12a Priory Road, Clifton, Bristol BS8 1TU, United Kingdom*

Abstract

The ability to keep working memory content up to date is vital for a number of higher cognitive functions such as navigation and reasoning, but it is also crucial for the effective operation of working memory itself. Removing outdated or irrelevant information allows focused processing of relevant information, and minimizes interference. We present evidence from three experiments that (1) people utilize an active removal process to update working memory, (2) that this removal process is an item-specific operation, and (3) that updating subsets of information held in working memory involves switching between maintenance and updating modes of working memory.

Keywords: working memory, updating, removal, executive functions

This paper was prepared using the recommended Elsevier L^AT_EXtemplate for the Journal of Memory and Language. Word count: 8,746.

*Corresponding author. Phone: +61-8-6488-3257.

Email addresses: ullrich.ecker@uwa.edu.au (Ullrich K. H. Ecker),
k.oberauer@psychologie.uzh.ch (Klaus Oberauer),
stephan.lewandowsky@bristol.ac.uk (Stephan Lewandowsky)

1. Introduction

Working memory (WM) is a severely capacity-limited system serving the configuration, manipulation, and maintenance of mental representations in support of ongoing cognition (Baddeley, 1986; Cowan, 1999; Oberauer, 2009). The flexible configuration of representations relies on ad-hoc integration of item and context information. For example, memorizing a telephone number requires the binding of digits to serial positions, and maintaining an accurate representation of one's environment while navigating traffic requires binding of shape, color, and dynamic location information over time (Oberauer, 2005; Wilhelm et al., 2013, also see Ecker et al., 2013).

Because of WM's capacity limitations, a fundamental requirement for its smooth operation is a mechanism that ensures outdated and no-longer-relevant information is discarded. Without such a mechanism, clutter from irrelevant information would prevent access to relevant information and would thus ultimately render efficient cognitive processing impossible.

Traditionally, a temporal decay mechanism has been proposed to serve this 'housekeeping' function (Baddeley, 2000; Barrouillet et al., 2007). However, supported by the growing evidence that WM representations do not, in fact, decay (Berman et al., 2009; Jalbert et al., 2011; Lewandowsky et al., 2009; Oberauer & Lewandowsky, 2008, 2013), we recently suggested that no-longer-relevant information is discarded by an active removal process (Ecker et al., in press).

In Ecker et al. (in press), we argued that this removal process is the central

component of WM updating, as for example when the operands “8” and “3” must be replaced by their product “24” during mental arithmetic. Updating WM is a process that is essential for maintaining a focus on relevant information and replacing outdated with current information, and is thus crucial for mental arithmetic, reading comprehension, navigation, and reasoning (Carretti et al., 2005; Chen & Li, 2007; Garavan, 1998; Gugerty, 1997). By definition, WM updating involves the substitution of outdated by new information and may thus be described as a two-component process: outdated information is removed and updated information is encoded to replace it (Ecker et al., 2010, also see Artuso & Palladino, 2011).

To investigate this proposed decomposition, Ecker et al. (in press) introduced a novel updating task that allowed the experimental separation of removal and encoding processes. Previous investigations of WM updating, using traditional WM updating tasks, focused either on the effectiveness of WM updating (e.g., Friedman et al., 2006) or its overall duration (Artuso & Palladino, 2011; Kessler & Meiran, 2008). Traditional WM updating tasks present a set of to-be-remembered items (e.g., the letters B-K-F) and then repeatedly replace one or more items (e.g., B-H-F . . . W-H-J, and so on) for participants to keep track of before they recall the most recent set at the end of a trial. In tasks of that type, removal of outdated items can only begin when the new items are presented. The time to update WM will thus include both time for removal and time for encoding. The novel task used by Ecker et al. (in press) presented a cue, indicating which item was about to be replaced, before presenting the new to-be-encoded stimulus. This allowed partic-

ipants to use the cue-target interval (CTI) to selectively remove an outdated item from the memory set. Varying this CTI thus varied the time available for removal. Ecker et al. (in press) found that longer CTIs led to faster updating, and argued that people used those long CTIs for removal, thus supporting the notion of a two-stage updating process comprising removal and encoding operations. Additional evidence for a removal process came from trials in which the to-be-replaced item in WM was similar, or even identical, to the new item replacing it. Ecker et al. (in press) found that with a short CTI, similarity or identity between the old and the new item reduced updating RTs, but these RT benefits were strongly reduced with a long CTI. This finding provides additional evidence that the long CTI was used for removing the old item, and not just to search for the to-be-updated item in the list (e.g., cf. Lange et al., 2011).

In the present paper, we present further evidence that an active removal process is central to WM updating, and expand on the previous work by demonstrating that (1) this removal operation is item-specific and goes beyond finding a to-be-updated item, and (2) how the removal process fits into a recent computational model of WM updating, which assumes that updating subsets of information held in working memory involves switching between maintenance and updating modes of working memory (cf. Kessler & Oberauer, in press).

We begin by revisiting a finding from the literature that suggests the involvement of multiple process in WM updating. Kessler & Meiran (2008) used an updating task that presented n items in a set of frames, and substituted between 1 and n items repeatedly with new items. The authors measured the time it took their

participants to complete this updating process, and found (in their Experiment 3) that updating RTs increased with the number of to-be-updated items up to $n - 1$ items, but that updating was much faster again when all n items were replaced on a given updating step. Thus, updating latencies depended in a non-monotonic fashion on the number of to-be-updated items.

Kessler & Meiran (2008) explained this non-monotonicity by proposing a distinction between local and global updating processes. Local updating refers to changes made to individual items, whereas global updating refers to the integration of all items in the current memory set after individual items were updated. The authors argued that partial updates require a complex sequence of (1) “un-binding” of the integrated representation of the previous memory set, (2) substitution of some but not all items (i.e., the actual local updating), followed by (3) re-binding the new set as part of the global updating process. In contrast, when the entire set is updated, steps (1) and (2) can be omitted, the old set is simply discarded and a new memory set is encoded and bound.

Our interpretation of the non-monotonicity of updating latencies is a specific instantiation of the ideas of Kessler & Meiran (2008), in light of the results of Ecker et al. (in press) and in the context of the SOB (serial order in a box) model of WM (Farrell & Lewandowsky, 2002; Lewandowsky & Farrell, 2008; Oberauer et al., 2012). SOB computationally implements the informational integration that is key to WM processing in a two-layered neural network that associates items (represented in one layer) with positional context markers (represented in the second layer) via Hebbian learning. To this end, the two layers are connected by

a weight matrix that is continuously adjusted by the learning algorithm. For instance, in an updating task such as the one by Kessler & Meiran (2008), SOB would associate each item to its frame through Hebbian learning. Forgetting in SOB occurs purely because of interference (there is no temporal decay mechanism). Interference from outdated information is prevented by an active removal mechanism (see in particular Oberauer et al., 2012). Removal of a specific item in SOB involves cuing the item with its position marker to retrieve it and then “unlearning” the association between the item and its position via Hebbian anti-learning (cf. Anderson, 1991; Lewandowsky, 1999). Thus, the removal of outdated and the encoding of new information is accomplished by two separate processes in SOB.

Kessler & Meiran (2008)’s notion of unbinding refers in SOB to the unlearning of selected items from their position markers. Like encoding of an item into the network (Jolicoeur & Dell’Acqua, 1998), the act of removing an individual item takes time (cf. Fawcett & Taylor, 2008; Oberauer, 2001). By contrast, wholesale removal of an entire memory set can be achieved in SOB by simply resetting the entire weight matrix, which we assume to be a very rapid process. It follows that updating an entire memory set does not require active, item-wise removal, and updating RT will mainly comprise the time needed for encoding the new memory set. This explains why updating an entire memory set is faster than partial updating of just one or two items within a larger set. This distinction between partial, item-wise updating and complete updating of an entire memory set is supported by neuro-cognitive evidence for selective recruitment of a neural network

by partial updating as opposed to total set replacement (Murty et al., 2011).

The active removal process described by Ecker et al. (in press) is therefore only utilized during partial updates, not complete updates, which can be achieved more efficiently by ‘wiping’ memory (i.e., in SOB terms, by resetting the weight matrix). It follows that bringing forward the removal process by providing a cue with a sufficiently large CTI should reduce updating RTs for partial updates—because the time can be used for unbinding of those representations—but not for updates of the entire set—because ‘wiping’ takes very little time and hence there is little to be gained by a longer CTI. Our notion of active item-wise removal thus predicts that the non-monotonicity of updating RTs observed by Kessler & Meiran (2008) should replicate with a short CTI, but should disappear with a long CTI. With a long CTI we assume that for both partial and complete updates removal is largely complete before the new stimuli are presented, so that the updating RTs reflect primarily the time for encoding the new stimuli. To test this prediction, we used the updating paradigm introduced by Ecker et al. (in press), but modified it such that on each updating step, either 1, 2, or the total set of 3 items was updated.

2. Experiment 1

Based on the distinction between slow selective removal and fast wiping of memory, we predicted that longer CTIs should lead to a substantial time gain in partial updating, but little gain on updating steps replacing the entire memory set.

The updating task consisted of (1) an encoding stage, during which participants studied a set of three letters presented in individual frames, (2) an updating

stage with multiple successive updating steps, each replacing 1, 2, or 3 letters of the set, and (3) a final recall stage targeting all 3 items, the only purpose of which was to ensure that participants were consistently updating the memory set during stage 2. Before each updating step, a cue signaled which items were about to be updated. The duration of the cue (i.e., the CTI) was either long—presumably allowing for removal to take place—or short—allowing only enough time to focus attention on the to-be-updated frames without providing sufficient time to initiate the removal process.

2.1. Methods

2.1.1. Participants

Seventy-four University of Western Australia undergraduates participated for partial course credit. Three participants were removed because of missing data ($n = 1$) or misunderstanding of task instructions ($n = 2$); two further participants were removed from the RT analyses because of outlying recall performance (see Results for details). The final sample for the main RT analysis thus comprised 69 participants (51 females, 18 males; mean age 20.8 years; age range 18-34 years).

2.1.2. Apparatus and Procedure

The experiment was run with the aid of Matlab and the Psychophysics toolbox (Brainard, 1997). Participants were tested individually in booths, sitting about 70 cm from a 17-in. thin-film transistor monitor.

2.1.3. Stimuli, Design, and Procedure

The letter updating task involved a single row of three black, rectangular frames. Following a fixation cross presented for one second, each trial began with the simultaneous 2-second presentation of 3 black consonants (ranging from B to Z) in the frames. The minimum alphabetic distance between the to-be-encoded letters was 2, to avoid sequences such as X, Y, Z.

A series of updating steps followed. On each step, new letter(s) were presented in 1, 2, or 3 frames. The to-be-updated frames at each step turned bold and red before the new content was displayed, potentially allowing participants to remove outdated memory content before encoding the new memoranda. The time between cue onset and onset of the new content (i.e., the CTI) was either short (i.e., 200 ms before presentation of the new items) or long (i.e., 1500 ms).

Participants pressed the space bar once they had encoded the new content, that is, as soon as they had updated their WM. The new memoranda remained on the screen until a response was made or the maximum updating time of 5 seconds was reached. After each updating step, the frames were blanked for 500 ms or 1800 ms in the long and short CTI condition, respectively, ensuring equal retention intervals in both conditions (i.e., 500 ms + 1500 ms in the long CTI condition and 1800 ms + 200 ms in the short CTI condition).

The number of updating steps varied from 1 to 21; the sequence finished with a constant probability of 10% after each updating step. This resulted in an unpredictable number of updating steps, and because each step was equally likely to be the last, participants had an equal incentive to carry out each updating step

independent of the duration of the sequence. Sequences had a mean number of about 9 updating steps. The number of to-be-updated frames and CTI duration were chosen randomly at each step.

After the updating phase of each trial, participants recalled the current contents of all frames. Recall was prompted by blue question marks appearing one-by-one in each frame in random order. Probes remained on the screen until a response was given, or until the maximum response time of 5 seconds per frame was reached. After recall of all three frames, feedback (“x out of 3 correct”) was given. The blank-screen inter-trial interval was 2.5 seconds. A representative (albeit short) trial sequence is shown in Figure 1.

In sum, the experiment comprised a total of 3 (number of updated frames: 1 vs. 2 vs. 3) \times 2 (cue-target interval: 200 vs. 1500 ms) conditions. There were 28 trials in total (plus 4 practice trials), yielding approximately 252 updating steps, or 42 per condition. Each trial took approximately 40 seconds, and the experiment took about 20 minutes.

[Figure 1 about here.]

2.2. Results

2.2.1. Recall accuracy analysis

Recall accuracy was very high, as expected, with $M = .96$ ($SD = .04$; $N = 71$). Two outliers, falling more than 1.5 interquartile ranges below the first quartile, were removed for the RT analysis.

2.2.2. Updating RT analysis

Updating response time data are shown in Figure 2.

A two-way repeated measures ANOVA on updating response times yielded a significant main effect of the number of updated frames, $F(2, 136) = 64.30$, $MSE = 0.03$, $p < .001$, $\eta_p^2 = 0.49$, a significant main effect of CTI, $F(1, 68) = 281.68$, $MSE = 0.02$, $p < .001$, $\eta_p^2 = 0.81$, as well as a significant interaction, $F(2, 136) = 109.55$, $MSE = 0.01$, $p < .001$, $\eta_p^2 = 0.62$. Planned contrasts showed that on average it took significantly longer to update two frames compared to one (1.44 seconds vs. 1.26 seconds; $F(1, 68) = 146.29$, $MSE = 0.02$, $p < .001$) and that with 1 or 2 frames, updating took significantly longer with a short as compared to a long CTI (1.53 vs. 1.17 seconds; $F(1, 68) = 320.14$, $MSE = 0.03$, $p < .001$). However, updating three frames was relatively quick (1.23 seconds), and CTI duration had a negligible impact on updating RT when all three frames were updated. While the CTI effect with three updated frames was statistically significant (arguably due to the large sample size and power; 1.26 vs. 1.20 seconds; $F(1, 68) = 13.59$, $MSE = 0.01$, $p < .001$), it was much smaller with three frames (53 ms) than with one or two frames (362 ms).

[Figure 2 about here.]

2.3. Discussion

The results of Experiment 1 support our hypothesis that partial updating of a memory set involves a process of active removal. Bringing forward this removal process by cueing the to-be-updated frames 1500 ms before presentation of the

new items sped up partial updating substantially. This time saving of roughly 300-400 ms can be interpreted as the time that is required to initiate item-wise removal, and at least partially remove information from WM. Based on previous research it is likely that near-complete removal takes a little more time (500-600 ms per item; cf. LaRocque et al., 2013; Oberauer, 2001).

In contrast, the opportunity to remove items from memory before presentation of new memoranda brought no substantial advantage when the entire memory set was updated. This finding supports our notion that memory can be cleared almost instantly, and that updating of an entire memory set does not require the time-consuming selective removal process.

The speed-up induced by the long CTI did not increase with the number of to-be-updated items. This unexpected observation seems to imply that removing one item from WM takes as long as removing two items. There are two possible explanations. One is that removal of multiple items can occur in parallel. While this is a theoretical possibility, it is at odds with SOB's notion that items must be retrieved individually to be removed selectively through anti-learning.

The second explanation is that people use the CTI only to remove one item, even when two items are about to be updated. This might be an efficient strategy because item-specific removal is likely to require the focus of attention, and switching the focus of attention to a new item takes time (cf. Garavan, 1998; Oberauer, 2003). Thus, the most efficient use of the CTI might be to focus on the first to-be-removed item and remove it, then wait. As soon as the new item(s) are presented, one of them can immediately be encoded in the currently focused frame.

Only then would the focus move on to the second to-be-updated frame (if there is one), remove the old item, and encode the new item.

This interpretation is in line with a recent proposal by Kessler & Oberauer (in press), which suggests that partial updating involves not only switching of the focus of attention between list positions, but also switching between two operating modes of WM, viz. maintenance and updating. Because this proposal is explored in our next experiment, we present it in some detail, considering first the situation in which there is no (or very little) pre-cued time for removal. In that situation participants are assumed to scan the items from the beginning of the list to the end (i.e., moving the focus of attention from frame to frame; cf. Farrell & Lelièvre, 2009) in the default maintenance mode (cf. O'Reilly, 2006). Hence, if the item in the first (i.e., left-most) frame is not updated, people stay in maintenance mode (M), perhaps refreshing the first item, before moving on to the next frame. As soon as a frame is encountered that requires updating, people switch to updating mode (U). Updating mode entails the active removal of the current item from WM (i.e., unbinding it from the currently-focused frame) and the encoding of its replacement by binding it to the same frame. Once updating is complete, people move on to the next frame. If the next frame is not updated, a switch back into maintenance (M) mode is required, whereas if a further update is required people remain in updating mode. It follows that updating frames 1 and 3 would require 3 mode switches from M to U and back; the sequence of modes while scanning the list from left to right would be (M)UMU, where the initial M in parentheses refers to the default mode at the outset. Updating frames 1 and 2 requires two

mode switches, (M)UUM; and updating frames 2 and 3 requires only one mode switch, (M)MUU. In the case of 1-frame updates, updating frame 1, (M)UMM, and updating frame 2, (M)MUM, would both require two mode switches, but updating frame 3 would only require one switch, (M)MMU. Kessler & Oberauer (in press) implemented this notion as a regression model with two predictors, the number of mode switches needed when scanning from left to right, and the number of new items to encode. This model gave a satisfactory account of the mean RTs across conditions differing in the number of updated items and their position in the list.

What would this “scan-switch-removal” model predict for the situation in which there is a long CTI that permits removal ahead of presentation of the replacement item? The assumptions of the model, applied to the present paradigm, are illustrated in Figure 3. We would expect participants to use the long CTI to scan the list up to the first to-be-updated frame, switch to updating mode, remove the item in that frame, and then wait—rather than move on to the next frame—because this strategy would minimize the number of both focus and mode switches. For instance, consider an updating step in which the first and the third item need to be updated. If people used the CTI to remove both items, they would have to carry out three focus switches—to the first item, then on to the second, then to the third—and three mode switches—(M)UMU—during the CTI. Once the new content is presented, they would have to shift the focus of attention back to the first and then to the third frame to encode the new items, with a possible intermediate switch to the second frame to refresh its content, including additional

mode switches. In total, people would do a minimum of five focus switches and a minimum of three mode switches. In contrast, with the ‘remove-one-and-wait’ strategy, people would only perform one focus switch (to the first frame) and one mode switch (from M to U) during the CTI. When the new content is presented, they would immediately encode the first new item in the first frame, then continue scanning through the list to complete the updating process, including two more focus switches and two more mode switches. Hence, this strategy would result in a net saving of at least two focus switches across the entire updating operation.

[Figure 3 about here.]

To test the assumptions just presented, we applied an extended version of the regression model of Kessler & Oberauer (in press) to the mean RTs of Experiment 1. We coded the hypothetical processes required at each updating step with the following predictors: The number of new items to encode (E^1 ; 1-3), the number of items still to be removed after the new items are presented (R ; 0-2), and the number of mode switches required to complete a full forward scan of the memory set (MS ; 0-3), as well as the number of focus switches (FS ; 0-2). We also included a refresh (RF) predictor to code the number of non-updated, to-be-refreshed items.² As explained in the Introduction, we assume that when all items

¹The E predictor is called item-position in Kessler & Oberauer (in press) because it reflects the number of new item-position bindings that need to be established.

²We note that the parameters are partially redundant; for example, for all partial updates, the number of focus switches in the long CTI condition is redundant with the sum of refresh and removal operations, while in the short CTI condition it is redundant with the sum of refresh and removal operations minus 1. This caused some singularities and hence model convergence failures. We nonetheless retained all parameters in order to test all identified model versions.

are to be updated (UUU), people rapidly wipe out the entire content of WM, rather than removing items one by one. Therefore, the number of removals (R) was set to 0 for the UUU condition, and instead we introduced a separate ‘wipe’ predictor (W). The wipe predictor codes the operation of rapidly wiping the entire content of WM; this predictor was set to 1 for each updating step in which all three letters were to be updated, and to 0 for all other cases.

Importantly, the predictors code only the processes left to be done after the CTI, because all processes that can be completed within the CTI do not count towards the measured updating RT. Therefore, the predictors differed between the short and long CTI conditions. For example, with a long CTI, a double update of frames 2 and 3 would involve no refreshing or mode switching, and only one focus switch: Participants could use the CTI to refresh the first item, switch to updating mode, remove the letter in frame 2, then wait. Once the new items are presented, what is left to do is to encode the new letter in frame 2, perform a focus switch to frame 3, remove the old letter in frame 3, and encode the new letter in frame 3. Hence the predictor values are $RF = MS = W = 0; FS = R = 1; E = 2$. In contrast, with a short CTI, participants could do nothing but switching the focus of attention to the first list item during the CTI. Once the new items are presented, they would have to refresh the first item, switch into updating mode, perform two focus switches, remove two items, and encode the two replacement items ($W = 0; RF = SW = 1; R = E = 2$). The complete set of model predictors for all conditions are given in Table 1.

[Table 1 about here.]

Linear models combining these predictors were fit through multi-level mixed-effects regression to the data of Experiment 1 (more specifically, to the individual RTs from each trial). Mixed-effects modeling is a flexible analysis tool that can simultaneously account for predictor effects and individual differences in these effects; by entering all predictors simultaneously as both fixed and random effects, the model estimates the intercept and main effects—in this case, the effects of the number of removals, the number of mode switches, etc.—while at the same time allowing these to vary across subjects. We first tested all possible additive models, and then added singular interaction terms to the best-fitting candidates. Table 2 presents the model fits for the five best-fitting models³, based on *BIC*.

The modeling showed that both the mode switch and the removal predictor were included in the best fitting models. Based on *BIC*, the model achieving the best fit was (*SEs* in parentheses): Updating RT = 704(40) + 156(14) × *E* + 87(7) × *R* + 393(18) × *MS* − 95(7) × *E* × *MS* ms, with a coefficient of determination *COD* = 0.437 and a Bayes Information Criterion *BIC* = 23403. The best-fitting model without a removal parameter achieved a *BIC* = 23437; which is 34 *BIC* units worse than the best-fitting model including removal. According to Raftery (1996) this difference in *BIC* values provides very strong evidence in favor of the removal model. Figure 4 shows the best-fitting model’s fit to the data.

[Table 2 about here.]

³We limit the number of model fits presented due to space considerations; results from the full set of 80+ model tests are available from the first author on request. The fact that none of the reported models involve a focus switch predictor is due to the partial redundancy of parameters and should not be taken as evidence against the relevance of focus switching.

[Figure 4 about here.]

The best-fitting model suggests that mode-switching, item-removal, and encoding of new items all contributed significantly to updating latencies. The regression weights of the predictors provide estimates for the contribution of the corresponding processes to updating RTs. For instance, encoding of each new item is estimated to take 156 ms, removing an item is estimated to take 87 ms, and switching between maintenance and updating mode is estimated to take 393 ms.⁴ The interaction reflects a reduction of mode switch cost by 95 ms for every to-be-encoded item. This could mean that switching from maintenance to updating mode (in order to encode a new item) is quicker than switching from updating to maintenance mode. In the task switching literature, such asymmetric switch costs are typically interpreted such that switching to the easier task takes longer than switching to the harder task (cf. Gilbert & Shallice, 2002; Schneider & Anderson, 2010). If maintenance is easier than updating, it would make sense to have maintenance as the default mode, and to have larger switch cost from updating to maintenance than vice versa. The fact that the wiping and refreshing parameters were not necessary for a good model fit implies that either participants did not wipe or refresh, or that the time required for these processes was so short that it was not measurably different from zero in our data, which seems plausible at least for the wiping process. The finding that the wiping predictor had no signifi-

⁴This interpretation of the fixed effects estimates assumes discrete sequential processing stages. In reality, processing stages might partially overlap, and thus the duration of the processes might be longer than indicated here.

cant regression weight should not be taken as evidence against the proposed rapid wiping process. This is because the rapid-wiping proposal is incorporated not only in the wiping predictor itself but also in how we coded the removal predictor. Specifically, we assumed no item-wise removal for the UUU conditions—without a quick wiping process, however, those conditions would require two (with a long CTI) or three (with a short CTI) removal operations.

A critic could argue that, if wiping memory is so much faster than removing individual items, it might be more efficient to wipe memory whenever updating is required, instead of relying on a slower, item-wise removal process. In response we note that in our paradigm, only new items are presented at each update step, whereas memorized items that are not being updated are not re-presented. Hence the task cannot be performed with indiscriminate wiping when fewer than 3 updates are required, because wiping would render information inaccessible that is still required for accurate retention. This situation mimics the real world where sometimes only certain aspects of an event are updated, while the constant aspects should still be maintained in memory.

To corroborate the assumption that only one item is removed during a long CTI, we additionally tested alternative versions of the best-fitting removal models, assuming that people can utilize the long CTI to remove multiple items. These models still assumed that participants completed a full scan through the list in both the short and long CTI conditions, including the associated mode and focus switching operations, but the removal predictor was set to 0 for all long-CTI conditions to reflect the assumption that all to-be-updated items were already re-

moved during the CTI. These models achieved worse fits (with $BICs > 23551$) than the removal models that assumed only one item to be removed during the long CTI.

3. Experiment 2

Experiment 1 suggested that people only remove one item during a long CTI, even when two items are marked as to-be updated. We argued this is efficient behavior as it minimizes focus switch and mode switch costs. Yet, an alternative hypothesis is that people would have removed more information had they had more time. Experiment 2 tested the idea that, with a sufficiently long CTI, people might remove more than one item, and thus show a larger gain in RTs from the long CTI in conditions that would benefit from the removal of more than one item. To this end, we added a third condition with a very long CTI to the design of Experiment 1, while omitting the full update condition, which according to our account does not involve removal. In addition, Experiment 2 provided a further opportunity to observe the robustness of our account of the intricacies of Experiment 1.

3.1. Methods

3.1.1. Participants

We tested 37 undergraduate students from the University of Western Australia, who received course credit. Three participants were removed from the main RT analyses because of outlying recall performance (see Results for details). The

final sample for the main RT analysis thus comprised 34 participants (20 females, 14 males; mean age 20.1 years; age range 17-31 years).

3.1.2. Apparatus and Procedure

Apparatus and procedure were identical to Experiment 1 except for the addition of a third, very-long (3000 ms) CTI condition, and the omission of the full update condition. Thus, the experiment comprised a total of 2 (number of updated frames: 1 vs. 2) \times 3 (CTI: 200 vs. 1500 vs. 3000 ms) conditions. There were 15 trials in total (plus 4 practice trials), with an average of 9 updating steps per trial, yielding approximately 135 updating steps, or 22.5 per condition. Each trial took approximately 40 seconds, and the experiment took about 15 minutes.

3.2. Results

3.2.1. Recall accuracy analysis

Recall accuracy was again very high, with $M = .94$ ($SD = .07$; $N = 37$). Three outliers, falling more than 1.5 interquartile ranges below the first quartile, were removed for the RT analysis.

3.2.2. Updating RT analysis

Updating response time data are shown in Figure 5.

[Figure 5 about here.]

A 2×3 repeated measures ANOVA on updating RTs yielded a main effect of the number of updated frames, $F(1,33) = 71.66$, $MSE = 0.03$, $p < .001$,

$\eta_p^2 = 0.68$, a main effect of CTI, $F(2,66) = 167.88$, $MSE = 0.02$, $p < .001$, $\eta_p^2 = 0.84$, but no significant interaction, $F(2,66) = 2.06$, $MSE = 0.02$, $p > .10$, $\eta_p^2 = 0.06$. Results showed that it took longer to update two frames than one, and that a longer CTI led to faster updating RTs. The effect of CTI was strong when comparing the short (200 ms) and long (1500 ms) conditions, but negligible when comparing long and very-long (3000 ms) conditions. The lack of interaction means that doubling the CTI still did not lead to a larger RT gain in the 2-frame updates compared to the 1-frame updates. Thus, it appears that even when given a 3000 ms headstart for removal, participants do not remove more than one item.

We applied the mixed effects regression model as specified in Experiment 1, but without the wiping parameter due to the absence of the full-set updating condition. As in Experiment 1, models featuring both a mode switch and a removal parameter achieved the best fits. As the long and very-long CTI conditions were coded identically, we additionally introduced a parameter c to account for the small observed RT difference between long and very-long CTI conditions (c was coded 1 in the very-long CTI conditions and 0 otherwise, and was added as a fixed effect only); this had only marginal impact. Table 3 presents the model fits for the five best-fitting models, based on BIC , plus the best-fitting model without removal, and the preferred model of Experiment 1.⁵ We found that a simple model provided the best fit (SE s in parentheses): Updating RT = $901(53) + 131(15) \times R + 255(18) \times MS$ ms, with $COD = 0.380$, $BIC = 6865$.

⁵We limit the number of model fits presented due to space considerations; results from the full set of 60+ model tests are available from the first author on request.

The best removal-free model achieved $BIC = 6905$; again the loss of fit relative to the best model with removal (BIC difference = 40) is very strong evidence in favor of a model featuring a removal predictor. Figure 6 shows the best-fitting model's fit to the data.

[Table 3 about here.]

[Figure 6 about here.]

To again corroborate the notion of one-item removal during a long CTI, we additionally tested alternative versions of the best-fitting removal models, assuming that people can utilize the long CTI to remove multiple items. As in Experiment 1, these models achieved worse fits (with $BICs > 6951$) than the removal models assuming one-item removal during the long CTI.

3.3. Discussion

The results from Experiment 2 again suggested that in the present task people only removed one item in anticipation of an update, even when this update concerned more than one item, and even when participants arguably had sufficient time (3 seconds) to perform two removal operations. This supports our notion that people avoid focus switching and mode switching when removing information from WM during updating. The modeling of the Experiment 2 data was broadly consistent with the modeling in Experiment 1, in that again models featuring both removal and mode-switching parameters achieved the best fits to the data. Differences between the modeling in Experiment 1 and the modeling in

Experiment 2 resulted mainly from the differences related to experimental design differences, viz. the full-set updating condition in Experiment 1 and the very-long CTI condition in Experiment 2. We highlight the fact, however, that when ignoring the experiment-specific model versions including W and c predictors, the modeling was remarkably consistent across experiments, with variants of a core model featuring removal and mode-switching supported by both BIC and AIC fit indices. The only notable difference between experiments was that the encoding predictor carried somewhat less weight in Experiment 2.

Our emphasis on the similarities in the modeling of Experiments 1 and 2 was buttressed by the results of an across-experiment modeling analysis. To this end, we restricted the modeling to the conditions shared by both experiments (i.e., we excluded the full-update and very-long CTI conditions from the data of Experiments 1 and 2, respectively). The data from both experiments were captured well by the same simple model with removal and mode-switching predictors. This model achieved the best fit (based on BIC) out of all possible models for both Experiments 1 and 2 (SEs in parentheses; Experiment 1: Updating $RT = 914(34) + 114(12) \times R + 247(13) \times MS$ ms, with $COD = 0.433, BIC = 16921$; Experiment 2: Updating $RT = 915(54) + 103(19) \times R + 282(20) \times MS$ ms, with $COD = 0.382, BIC = 5084$).

To further corroborate the somewhat counterintuitive conclusion that people remove only one old item before receiving the new content, even if they have 3 seconds for removal, we designed another experiment. This final experiment was based on Ecker et al. (in press)'s Experiment 1. That experiment investigated the

effects of occasional item repetitions in an updating task equivalent to the task used here. In repetition trials the new letter presented for updating matched the old letter in the same frame. In that case, there is no item substitution, and hence no need for actual updating. Accordingly, Ecker et al. (in press) found that updating steps with an item repetition were markedly faster than those presenting an actually new item. More importantly, Ecker et al. (in press) demonstrated that the opportunity to remove a to-be-updated item during a long CTI virtually eliminated this item repetition benefit in updating RTs. In other words, updating RTs are short when an item is repeated in the same frame (as replacing an item with itself does not actually require any updating), but this repetition benefit is eliminated if the to-be-replaced item is removed from WM before the new (identical) item is presented.

In the present context, this observation leads to the following prediction: If people use a long CTI to only remove one item even when two items are marked as to-be updated, then a long CTI should abolish the item-repetition benefit for the removed item but not for the non-removed item. Further assuming that in the present paradigm, people scan their memory set in forward order (i.e., starting with frame-1, ending with frame-3) and remove only the first—that is, the left-most—to-be-substituted item, the repetition benefit should be present for the right-most item in both short and long CTI conditions, but should occur for the left-most item only in the short but not the long CTI condition. For instance, if the current memory set is B-C-D, and the first and third frame are cued as to-be-updated, then a long CTI would be used to remove B but not D. If the new items

presented after the CTI are B-()-N, there would be no repetition benefit for the repeated B, compared to a condition where the new items are G-()-N. In contrast, if the new items are F-()-D, there would still be a repetition benefit for the repeated D, compared to a condition where the new items are F-()-G. The purpose of Experiment 3 was to test this precise prediction of a highly specific interaction between CTI duration and repetition condition.

4. Experiment 3

Experiment 3 modified the updating task used in the present Experiment 1 in the following ways: (1) Each updating step updated two frames; (2) occasionally (20% of updating steps), a single item repeated across two successive updating steps involving its specific frame (as in Experiment 1 of Ecker et al., in press). As two frames were updated but only one item (occasionally) repeated, the repetition could occur either in the left or the right updated frame. To reiterate, if only the content of the first (i.e., left-most) frame is removed during a long CTI, then a long CTI should only lead to a reduction of the expected repetition benefit for the left but not the right updated frame.

4.1. Methods

4.1.1. Participants

We tested 43 undergraduate students from the University of Western Australia, who received course credit. One participant was removed for not following task instructions; three participants were additionally removed from the main RT anal-

yses because of outlying recall performance (see Results for details). One participant had missing data, but the number of correctly completed trials (36) was deemed sufficient to retain the participant's data.⁶ The final sample for the main RT analysis thus comprised 39 participants (29 females, 14 males; mean age 20.3 years; age range 18-29 years).

4.1.2. Apparatus and Procedure

Apparatus and procedure were identical to Experiment 1 except for the constant updating of two frames, and the repetition of a single item on a random 20% of the updating steps. The experiment comprised a total of 3 (repetition: no-repetition vs. repetition-left vs. repetition-right) \times 2 (CTI: 200 vs. 1500 ms) conditions. There were 70 trials in total (plus 4 practice trials), with an average of 9 updating steps per trial, yielding approximately 630 updating steps, or 252 per no-repetition condition, and 31.5 per repetition condition. Each trial took approximately 40 seconds, and the experiment took about 50 minutes.

4.2. Results

4.2.1. Recall accuracy analysis

Recall accuracy was again very high, with $M = .96$ ($SD = .07$; $N = 42$). Three outliers falling more than 1.5 interquartile ranges below the first quartile were removed for the RT analysis.

⁶Excluding this participant did not affect the outcome.

4.2.2. Updating RT analysis

Updating response time data are shown in Figure 7.

[Figure 7 about here.]

A 3×2 repeated measures ANOVA on updating RTs yielded a main effect of repetition, $F(2, 76) = 3.44$, $MSE = 0.01$, $p = .04$, $\eta_p^2 = 0.08$, a main effect of CTI, $F(1, 38) = 147.59$, $MSE = 0.03$, $p < .001$, $\eta_p^2 = 0.80$, as well as a significant interaction, $F(2, 76) = 4.60$, $MSE = 0.01$, $p = .01$, $\eta_p^2 = 0.11$. The observed interaction exactly mirrored the predicted pattern: Repetitions on the left incurred an RT benefit at the short CTI, but that repetition benefit was entirely abolished at the long CTI. Contrast analysis confirmed that the difference between no-repetition and repetition-left conditions was greater in the short compared to the long CTI condition (significant interaction contrast, $F(1, 38) = 7.47$, $MSE = 0.005$, $p = .01$). Repetitions on the right, by contrast, incurred a benefit regardless of CTI. As a consequence, the repetition benefits on the left and on the right were equally large with a short CTI, whereas at a long CTI there was a repetition benefit only on the right. Statistically, the difference between repetition-left and repetition-right conditions was greater in the long compared to the short CTI condition (significant interaction contrast, $F(1, 38) = 7.74$, $MSE = 0.01$, $p = .01$), whereas the difference between no-repetition and repetition-right conditions was not affected by CTI ($F < 1$ for the interaction contrast).

We again applied our mixed-effects regression modeling approach. One challenge in the modeling of Experiment 3 was how to deal with the item repetitions,

that is, with situations where the cue prompts updating but the item prompts maintenance. Modeling of the long-CTI condition was straightforward, because we assumed that the first to-be-updated item encountered during scanning is removed, regardless of whether it was later to be replaced by an identical item or not. For the short-CTI condition, however, there were two options: (1) The cognitive system detects the repetition before deciding upon the operating mode (updating or maintenance), and hence remains in, or shifts into, maintenance mode. (2) The system first decides on the operation mode, hence remaining in, or shifting into, updating mode (as indicated by the cue), and only then detects the repetition, which obviates the removal of the item. We tried both options, in each case adding a further parameter D to capture the additional time for detecting the repetition. Initial modeling showed that possibility (1) invariably led to serious misfits of the data, and these models were thus discarded. By contrast, models of variety (2) achieved good fits, and Table 4 summarizes the model predictors for all conditions.

As in Experiments 1 and 2, the best-fitting models of Experiment 3 included both removal and mode-switch parameters; they are summarized in Table 5. We again found that a simple model provided the best fit (SEs in parentheses): Updating $RT = 862(94) + 71(51) \times R + 36(51) \times D$ ms, with $COD = 0.430$, $BIC = 37890$. The best removal-free model achieved $BIC = 37950$; again the loss of fit relative to the best model with removal (BIC difference = 60) is very strong evidence in favor of a model featuring a removal predictor. Figure 8 shows the best-fitting model's fit to the data.

[Table 4 about here.]

[Table 5 about here.]

[Figure 8 about here.]

4.3. Discussion

The results of Experiment 3 again showed that a longer CTI led to faster updating RTs, corroborating our claim of an active removal process contributing to WM updating. Results also showed that item repetition led to an updating RT benefit in all conditions, except for the long CTI condition that featured a repetition in the left-most frame. This supports our notion that item-specific removal took place only in the long CTI condition, and that even when two frames were updated, only the left-most to-be-updated item was in fact removed before presentation of the new items. We interpret this as corroborating evidence for our proposal that during WM updating in the current paradigm, people scan the memory set in forward order until encountering a to-be-substituted item, at which point WM switches from maintenance to updating mode, and the item is actively removed. To avoid further switches of WM mode and the focus of attention, the focus then stands by until a new item can be encoded by associating it to the cleared context (cf. also Kessler & Oberauer, in press), even if there would be enough time to perform a further removal.

5. General Discussion

Across three experiments, we provided evidence that giving people sufficient time in an updating task to remove to-be-substituted items before the presentation of new items substantially speeds subsequent updating. This supports Ecker

et al. (in press)'s proposal that active removal is a core process involved in WM updating.

As alluded to in the Introduction, an alternative view might suggest that in the present set of experiments, people could have used the long CTI just to *find* the to-be-removed item, without actually removing it. We endorse the view that part of the long CTI is used to find the first to-be-removed item, and we note that this search process is reflected in our modeling in the assumption of sequential scanning and the time cost for a frame switch. However, assuming that the long CTI is used *only* for finding the to-be-replaced item but not for removing it cannot explain why, with a long CTI, item-repetition benefits disappear, as shown by (Ecker et al., in press) and again in the present Experiment 3. Therefore, the assumption that people use the long CTI to both find and remove the first to-be-updated item is in better agreement with the data than the alternative explanation that the CTI is used only for finding that item.

The present evidence for an active removal process in WM converges with other findings supporting active removal. Using a short-term variant of the DRM false-memory paradigm, Festini & Reuter-Lorenz (2013) showed that only to-be-remembered items, but not to-be-forgotten items, produced false memories for semantic associates of those items. Hence, directed-forgetting instructions reduced semantic-associative processing of the to-be-forgotten items (Lin et al., 2013). Festini & Reuter-Lorenz (in press) used a two-list modified Sternberg paradigm, with directed forgetting of one of the lists (Oberauer, 2001). On some trials, study items from the previous trial served as the recognition probe. False alarm rates

and correct rejection RTs to recent to-be-forgotten items were not different from new items, showing that directed forgetting can virtually eliminate proactive interference in WM, suggesting complete removal. Likewise, Williams et al. (2013) showed that removal of irrelevant items from WM in a change-detection task can be highly effective: the authors provided evidence that items cued for removal were no longer available in WM subsequently, suggesting complete removal. The authors also demonstrated that removal of irrelevant items improved the representational quality of the remaining items. Finally, Fawcett & Taylor (2008, 2012) showed that the act of intentional forgetting interfered with an unrelated secondary task and reduced incidental memory for distractor stimuli.

The question remains what specific mechanisms underlie these effects. Festini & Reuter-Lorenz (2013, in press) attributed their findings to an active forgetting process that attenuated the memory representation of to-be-forgotten items. They speculated that this might be an inhibitory process, as suggested for example by Hasher & Zacks (1988). Conceptually, our proposed removal process can be considered an inhibitory process, although the item-context “unlearning” implementation of removal in SOB is computationally different from traditional implementations of inhibition such as lateral inhibition (for a recent application in a WM model, see Chuderski et al., in press). Future research may investigate the relation between removal ability and other inhibitory processes such as response inhibition. For now, we argue that removal and inhibition, even though they serve similar cognitive-control functions, should be conceptually distinguished because they work differently in computational models (cf. Chuderski et al., 2012; Dave-

laar & Cooper, 2010).

Festini & Reuter-Lorenz (2013, in press) further speculated that selective rehearsal (i.e., preferential processing of non-removed information) might have contributed to their results, but emphasized that active forgetting seems to differ from simply blocking rehearsal of the to-be-forgotten information. We agree that active removal is not inherently related to (although it may result in) selective rehearsal.

An account based purely on selective rehearsal would implicitly require a process to achieve the kind of representational weakening described by Festini & Reuter-Lorenz (in press), that is, a process by which non-rehearsed WM contents disappear—they could be assumed to decay, or to be crowded out by interference from the stronger rehearsed contents. Decay is a particularly unlikely explanation for the finding of Festini & Reuter-Lorenz (in press) that directed forgetting eliminates proactive interference from recent negative probes, because Berman et al. (2009) found that proactive interference from recent negative probes remains largely undiminished across a large range of inter-trial intervals. Thus, the mere passage of time does not do much to get rid of outdated information in WM; doing so requires an active process of forgetting.

To further support this argument, we argue that the differential pattern of repetition benefits for left-repetitions and right-repetitions at long CTIs in Experiment 3 presents a strong case against both decay and interference-based crowding out as the ‘housekeeping’ mechanism for clearing out outdated information from WM (adding to the already existing evidence against decay in verbal WM (Berman et al., 2009; Jalbert et al., 2011; Lewandowsky et al., 2009; Oberauer &

Lewandowsky, 2008, 2013)). To specify: Selective rehearsal of to-be-remembered items, combined with decay or interference-based crowding out of to-be-forgotten items, implies that all non-rehearsed items disappear from WM at the same rate. This account cannot be reconciled with the fact that, after a long CTI, repetition benefits differed for to-be-removed items on the left and on the right side of the memory set. Therefore, the results of Experiment 3 strengthen the hypothesis of active removal and discriminate it from passive decay or crowding out of outdated information in WM.

Williams et al. (2013) took their results to imply that there are two ways for information to leave WM—an item can be replaced with a new item or it can be removed in a top-down fashion. We argue, in contrast, that it is the same removal process that is involved in both the “replacement” and “top-down removal” scenarios described by Williams et al. (2013), just accompanied or not accompanied by the encoding of a new item. A process of replacement separate from removal and encoding, as envisioned by Williams et al. (2013), does not agree with the mechanisms in the SOB model: If encoding involves binding an item to its context, then without removal (i.e., unbinding) the new item-context binding would be simply superimposed onto the existing one, rather than actually replacing it.

The distinction between removal of old information and encoding of new information in our theoretical framework implies that removal can also operate without immediate replacement of the removed content by new content. Whereas in the present paper we conceptualized removal and encoding as twin processes jointly accomplishing WM updating, removal can also be useful in situations that

are not typically considered to involve updating. For example, when a list item has been recalled and participants expect no further recall demand for that item, the recalled item is removed to avoid repetition errors. Likewise, if information encoded into WM distracts from the main task at hand, as the distractor items do in complex span tasks, the distracting information can be removed. We argue that there is now considerable evidence for the generality of a process of removal in cognition that can be formalized as Hebbian anti-learning (Anderson, 1991). Removal by anti-learning has been used to explain response suppression in standard (serial) recall (cf. Lewandowsky & Li, 1994; Lewandowsky, 1999; Lewandowsky & Farrell, 2000), removal of distractors in complex-span tasks (cf. Oberauer et al., 2012), and the removal of outdated items during updating.

Taken together, a number of recent studies support the notion that active removal is vital for WM updating, and can also act as the “housekeeper” of WM, serving to minimize interference from irrelevant information and thus allowing more efficient WM processing. It follows that the present results also support the implementation of a removal mechanism into contemporary computational models of WM, such as SOB (Oberauer et al., 2012).

In this context, however, one conundrum remains. In two individual-differences studies, we found that the efficiency of information removal and substitution in WM was unrelated to WM capacity (Ecker et al., 2010, in press). While these studies provide evidence that removal and substitution processes can be differentiated from more generic WM functions (such as retrieval), it is unclear why higher removal efficiency does not lead to higher WM capacity; this question needs to be

addressed by future research.

Besides providing additional evidence for active removal of irrelevant information from WM, the present study sheds more light on the general workings of WM. We have provided evidence that circumstances exist in which people seem to only remove one item even though they know that two items will have to be updated. On the one hand, this demonstrates that removal is an item-specific process, and thus supports the assumption, derived from SOB, that removal operates via unlearning of a specific item-context association. On the other hand, this finding supports the notion that people try to avoid the costs associated with switching their focus of attention (Basak & Verhaeghen, 2011; Oberauer, 2002) and also their operational mode of WM (see below and Kessler & Oberauer, in press). A precedent for the avoidance of switch costs can be found in the literature on problem solving and categorization, where people may continue to use a suboptimal strategy to avoid strategy switch costs (Kalish et al., 2005; Lemaire & Lecacheur, 2010; Lewandowsky et al., 2000).

Finally, our modeling confirms some of the conclusions advanced by Kessler & Oberauer (in press): First, people carry out selective updating of a set of verbal memory items by scanning through the set in forward order (here: from left to right; we acknowledge that this may not apply to non-verbal stimuli, in particular if items are presented in a less-structured spatial array). The selective abolishment of the repetition benefit for the left-most but not the right-most updated item in the set (Experiment 3) provides a direct confirmation for that assumption. Second, selective updating of WM contents entails switching between maintenance and up-

dating modes of WM, and these switches incur a time cost that is reflected in the regression weight of the mode-switch predictor. Our consistent finding of a significant regression weight for the mode-switch predictor thus supports dual-mode theories of WM, with a default maintenance mode on the one hand—providing the stability expected of any memory system—and an updating mode on the other—providing the flexibility that a working memory system in service of higher cognition requires (Artuso & Palladino, 2011; Kessler & Oberauer, in press). We argue that this conceptualization of dual WM modes provides a compelling explanation for how WM can achieve both stability and flexibility. The notion of dual modes of operation is also supported by recent findings from cognitive neuroscience. For example, Roth et al. (2006) identified a frontoparietal network specifically involved in WM updating, and Murty et al. (2011) provided evidence for selective recruitment of a neural network by partial updating as opposed to maintenance of WM content, thus supporting a distinction between updating and maintenance modes of WM that has also been incorporated into contemporary neuro-cognitive frameworks of WM (cf. Bledowski et al., 2010; O'Reilly, 2006).

6. Acknowledgments

Preparation of this paper was facilitated by Discovery Grants and Research Fellowships from the Australian Research Council to Ullrich Ecker and Stephan Lewandowsky, and a grant from the Swiss National Science Foundation to Klaus Oberauer. We thank Charles Hanich and Devon Spaapen for research assistance, and Yoav Kessler and Nelson Cowan for their comments on earlier drafts.

References

- Anderson, J. (1991). Why, having so many neurons, do we have so few thoughts? In *Relating theory and data: Essays on human memory in honor of Bennet B. Murdock* (pp. 477–507). Hillsdale, NJ: Lawrence Erlbaum.
- Artuso, C., & Palladino, P. (2011). Content-context binding in verbal working memory updating: On-line and off-line effects. *Acta Psychologica, 136*, 363–369.
- Baddeley, A. (2000). The episodic buffer: A new component of working memory? *Trends in Cognitive Sciences, 4*, 417–423.
- Baddeley, A. D. (1986). *Working memory*. New York: Oxford University Press.
- Barrouillet, P., Bernardin, S., Portrat, S., Vergauwe, E., & Camos, V. (2007). Time and cognitive load in working memory. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 33*, 570–585.
- Basak, C., & Verhaeghen, P. (2011). Three layers of working memory: Focus-switch costs and retrieval dynamics as revealed by the n-count task. *Journal of Cognitive Psychology, 23*, 204–219.
- Berman, M. G., Jonides, J., & Lewis, R. L. (2009). In search of decay in verbal short-term memory. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 35*, 317–333.

- Bledowski, C., Kaiser, J., & Rahm, B. (2010). Basic operations in working memory: Contributions from functional imaging studies. *Behavioural Brain Research*, *214*, 172–179.
- Brainard, D. H. (1997). The Psychophysics Toolbox. *Spatial Vision*, *10*, 433–436.
- Carretti, B., Cornoldi, C., De Beni, R., & Romano, M. (2005). Updating in working memory: A comparison of good and poor comprehenders. *Journal of Experimental Child Psychology*, *91*, 45–66.
- Chen, T., & Li, D. (2007). The roles of working memory updating and processing speed in mediating age-related differences in fluid intelligence. *Aging, Neuropsychology, and Cognition*, *14*, 631–646.
- Chuderski, A., Andrelczyk, K., & Smolen, T. (in press). An oscillatory model of individual differences in working memory capacity and relational integration. *Cognitive Systems Research*, .
- Chuderski, A., Taraday, M., Necka, E., & Smolen, T. (2012). Storage capacity explains fluid intelligence but executive control does not. *Intelligence*, *40*, 278–295.
- Cowan, N. (1999). An embedded-processes model of working memory. In A. Miyake, & P. Shah (Eds.), *Models of working memory: Mechanisms of active maintenance and executive control* (pp. 62–101). New York: Cambridge University Press.

- Davelaar, E., & Cooper, R. P. (2010). Modelling the correlation between two putative inhibition tasks: An analytic approach. In S. Ohlsson, & R. Catrambone (Eds.), *Proceedings of the 32nd Annual Conference of the Cognitive Science Society* (pp. 937–942). Austin, TX: Cognitive Science Society.
- Ecker, U. K. H., Lewandowsky, S., & Oberauer, K. (in press). Removal of information from working memory: A specific updating process. *Journal of Memory and Language*, .
- Ecker, U. K. H., Lewandowsky, S., Oberauer, K., & Chee, A. E. H. (2010). The components of working memory updating: An experimental decomposition and individual differences. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *36*, 170–189.
- Ecker, U. K. H., Maybery, M., & Zimmer, H. D. (2013). Binding of intrinsic and extrinsic features in working memory. *Journal of Experimental Psychology: General*, *142*, 218–234.
- Farrell, S., & Lelièvre, A. (2009). Is scanning in probed order recall articulatory? *Quarterly Journal of Experimental Psychology*, *62*, 1843–1858.
- Farrell, S., & Lewandowsky, S. (2002). An endogenous distributed model of ordering in serial recall. *Psychonomic Bulletin & Review*, *9*, 59–79.
- Fawcett, J. M., & Taylor, T. L. (2008). Forgetting is effortful: Evidence from reaction time probes in an item-method directed forgetting task. *Memory & Cognition*, *36*, 1168–1181.

- Fawcett, J. M., & Taylor, T. L. (2012). The control of working memory resources in intentional forgetting: Evidence from incidental probe word recognition. *Acta Psychologica, 139*, 84–90.
- Festini, S. B., & Reuter-Lorenz, P. A. (2013). The short- and long-term consequences of directed forgetting in a working memory task. *Memory, 21*, 763–777.
- Festini, S. B., & Reuter-Lorenz, P. A. (in press). Cognitive control of familiarity: Directed forgetting reduces proactive interference in working memory. *Cognitive, Affective and Behavioral Neuroscience*, .
- Friedman, N. P., Miyake, A., Corley, R. P., Young, S. E., DeFries, J. C., & Hewitt, J. K. (2006). Not all executive functions are related to intelligence. *Psychological Science, 17*, 172–179.
- Garavan, H. (1998). Serial attention within working memory. *Memory & Cognition, 26*, 263–276.
- Gilbert, S. J., & Shallice, T. (2002). Task switching: A PDP model. *Cognitive Psychology, 44*, 297–337.
- Gugerty, L. (1997). Situation awareness during driving: Explicit and implicit knowledge in dynamic spatial memory. *Journal of Experimental Psychology: Applied, 3*, 42–66.
- Hasher, L., & Zacks, R. (1988). Working memory, comprehension, and aging: A

- review and a new view. In G. H. Bower (Ed.), *The Psychology of Learning and Motivation*, 22 (pp. 193–225). London, UK: Elsevier.
- Jalbert, A., Neath, I., Bireta, T. J., & Surprenant, A. M. (2011). When does length cause the word length effect? *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 37, 338–353.
- Jolicoeur, P., & Dell’Acqua, R. (1998). The demonstration of short-term consolidation. *Cognitive Psychology*, 36, 138–202.
- Kalish, M. L., Lewandowsky, S., & Davies, M. (2005). Error-driven knowledge restructuring in categorization. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 31, 846–861.
- Kessler, Y., & Meiran, N. (2008). Two dissociable updating processes in working memory. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 34, 1339–1348.
- Kessler, Y., & Oberauer, K. (in press). Working memory updating latency reflects the cost of switching between maintenance and updating modes of operation. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, .
- Lange, E. B., Cerella, J., & Verhaeghen, P. (2011). Ease of access to list items in short-term memory depends on the order of the recognition probes. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 37, 608–620.
- LaRocque, J. J., Lewis-Peacock, J. A., Drysdale, A. T., Oberauer, K., & Postle,

- B. R. (2013). Decoding attended information in short-term memory: An EEG study. *Journal of Cognitive Neuroscience*, *25*, 127–142.
- Lemaire, P., & Lecacheur, M. (2010). Strategy switch costs in arithmetic problem solving. *Memory & Cognition*, *38*, 322–332.
- Lewandowsky, S. (1999). Redintegration and response suppression in serial recall: A dynamic network model. *International Journal of Psychology*, *34*, 434–446.
- Lewandowsky, S., & Farrell, S. (2000). A redintegration account of the effects of speech rate, lexicality, and word frequency in immediate serial recall. *Psychological Research*, *63*, 163–73.
- Lewandowsky, S., & Farrell, S. (2008). Short-term memory: New data and a model. In B. H. Ross (Ed.), *The Psychology of Learning and Motivation*, *49* (pp. 1–48). London, UK: Elsevier.
- Lewandowsky, S., Kalish, M., & Griffiths, T. (2000). Competing strategies in categorization: Expediency and resistance to knowledge restructuring. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *26*, 1666–1684.
- Lewandowsky, S., & Li, S.-C. (1994). Memory for serial order revisited. *Psychological Review*, *101*, 539–543.
- Lewandowsky, S., Oberauer, K., & Brown, G. D. A. (2009). No temporal decay in verbal short-term memory. *Trends in Cognitive Sciences*, *13*, 120–126.

- Lin, W.-J., Kuo, Y.-C., Liu, T.-L., Han, Y.-J., & Cheng, S.-K. (2013). Intentional forgetting reduces the semantic processing of to-be-forgotten items: An ERP study of item-method directed forgetting. *Psychophysiology*, *50*, 1120–1132.
- Murty, V. P., Sambataro, F., Radulescu, E., Altamura, M., Iudicello, J., Zolnick, B., Weinberger, D. R., Goldberg, T. E., & Mattay, V. S. (2011). Selective updating of working memory content modulates meso-cortico-striatal activity. *NeuroImage*, *57*, 1264–1272.
- Oberauer, K. (2001). Removing irrelevant information from working memory: A cognitive aging study with the modified Sternberg task. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *27*, 948–957.
- Oberauer, K. (2002). Access to information in working memory: Exploring the focus of attention. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *28*, 411–421.
- Oberauer, K. (2003). Understanding serial position curves in short-term recognition and recall. *Journal of Memory and Language*, *49*, 469–483.
- Oberauer, K. (2005). Binding and inhibition in working memory: Individual and age differences in short-term recognition. *Journal of Experimental Psychology: General*, *134*, 368–387.
- Oberauer, K. (2009). Design for a working memory. In B. H. Ross (Ed.), *The Psychology of Learning and Motivation*, *51* (pp. 45–100). San Diego: Academic Press.

- Oberauer, K., & Lewandowsky, S. (2008). Forgetting in immediate serial recall: Decay, temporal distinctiveness, or interference? *Psychological Review*, *115*, 544–576.
- Oberauer, K., & Lewandowsky, S. (2013). Evidence against decay in verbal working memory. *Journal of Experimental Psychology: General*, *142*, 380–411.
- Oberauer, K., Lewandowsky, S., Farrell, S., Jarrold, C., & Greaves, M. (2012). Modeling working memory: An interference model of complex span. *Psychonomic Bulletin & Review*, *19*, 779–819.
- O'Reilly, R. C. (2006). Biologically based computational models of high-level cognition. *Science*, *314*, 91–94.
- Raftery, A. E. (1996). Approximate Bayes factors and accounting for model uncertainty in generalised linear models. *Biometrika*, *83*, 251–266.
- Roth, J., Serences, J., & Courtney, S. (2006). Neural system for controlling the contents of object working memory in humans. *Cerebral Cortex*, *16*, 1595–1603.
- Schneider, D. W., & Anderson, J. R. (2010). Asymmetric switch costs as sequential difficulty effects. *Quarterly Journal of Experimental Psychology*, *63*, 1873–1894.
- Wilhelm, O., Hildebrandt, A., & Oberauer, K. (2013). What is working memory capacity, and how can we measure it? *Frontiers in Psychology*, *4*, article 433.

Williams, M., Hong, S., Kang, M.-S., Carlisle, N., & Woodman, G. (2013). The benefit of forgetting. *Psychonomic Bulletin and Review*, *20*, 348–355.

Table 1: Model Predictor Values for each Updating Condition in Experiment 1

Condition	<i>E</i>	<i>R</i>	<i>MS</i>	<i>FS</i>	<i>W</i>	<i>RF</i>
UMM-S	1	1	2	2	0	2
MUM-S	1	1	2	2	0	2
MMU-S	1	1	1	2	0	2
UUM-S	2	2	2	2	0	1
UMU-S	2	2	3	2	0	1
MUU-S	2	2	1	2	0	1
UUU-S	3	0	1	2	1	0
UMM-L	1	0	1	2	0	2
MUM-L	1	0	1	1	0	1
MMU-L	1	0	0	0	0	0
UUM-L	2	1	1	2	0	1
UMU-L	2	1	2	2	0	1
MUU-L	2	1	0	1	0	0
UUU-L	3	0	0	2	0	0

Note. Updating conditions specified by U and M, updating and maintenance requirements for each of the three frames, and S and L, short and long CTI conditions; Model predictors coded as *E*, Encoding; *R*, Removal; *MS*, Mode Switch; *FS*, Focus Switch; *W*, Wiping; *RF*, Refreshing. See text for details.

Table 2: Model Fits in Experiment 1

Model	<i>BIC</i>	<i>AIC</i>	<i>deviance</i>	<i>COD</i>
E + MS + W	23453	23337	23281	0.434
E + MS + W + E × MS	23437	23314	23248	0.435
E + R + MS + E × MS	23403	23279	23212	0.437
E + R + MS + W + E × MS	23430	23260	23175	0.439
E + R + MS + W + E × R	23411	23241	23157	0.439

Note. Five best-fitting models in Experiment 1, based on *BIC*. Model predictors coded as *E*, Encoding; *R*, Removal; *MS*, Mode Switch; *W*, Wiping; *RF*, Refreshing. *BIC*, Bayes Information Criterion; *AIC*, Akaike Information Criterion; *COD*, coefficient of determination. Best-fitting model in bold font. See text for details.

Table 3: Model Fits in Experiment 2

Model	<i>BIC</i>	<i>AIC</i>	<i>deviance</i>	<i>COD</i>
E + MS	6910	6847	6811	0.378
R + MS	6865	6802	6765	0.380
R + MS + R × MS	6880	6810	6765	0.380
R + MS + c	6871	6801	6756	0.382
R + MS + c + R × MS	6886	6810	6756	0.382
E + R + MS	6889	6794	6742	0.391
E + R + MS + E × MS	6903	6801	6741	0.391

Note. Five best-fitting models in Experiment 2, based on *BIC*, plus best-fitting model without removal predictor (first row), and best-fitting model from Experiment 1 (last row). Model predictors coded as *E*, Encoding; *R*, Removal; *MS*, Mode Switch; *c*, constant benefit for very-long CTI condition. *BIC*, Bayes Information Criterion; *AIC*, Akaike Information Criterion; *COD*, coefficient of determination. Best-fitting model in bold font. See text for details.

Table 4: Model Predictor Values for each Updating Condition in Experiment 3

Condition	E	R	MS	FS	D	RF
UUM-SN	2	2	2	2	0	1
UMU-SN	2	2	3	2	0	1
MUU-SN	2	2	1	2	0	1
UUM-LN	2	1	1	2	0	1
UMU-LN	2	1	2	2	0	1
MUU-LN	2	1	0	1	0	0
UUM-SL	1	1	2	2	1	1
UMU-SL	1	1	3	2	1	1
MUU-SL	1	1	1	2	1	1
UUM-LL	2	1	1	2	0	1
UMU-LL	2	1	2	2	0	1
MUU-LL	2	1	0	1	0	0
UUM-SR	1	1	2	2	1	1
UMU-SR	1	1	3	2	1	1
MUU-SR	1	1	1	2	1	1
UUM-LR	1	0	1	2	1	1
UMU-LR	1	0	2	2	1	1
MUU-LR	1	0	0	1	1	0

Note. Updating conditions specified by U and M, updating and maintenance requirements for each of the three frames, S and L, short and long CTI conditions, and N, L, and R, coding no-repetition, repetition in the left ‘U’ frame, and repetition in the right ‘U’ frame; Model predictors coded as *E*, Encoding; *R*, Removal; *MS*, Mode Switch; *FS*, Focus Switch; *D*, Detection of repetition; *RF*, Refreshing. See text for details.

Table 5: Model Fits in Experiment 3

Model	<i>BIC</i>	<i>AIC</i>	<i>deviance</i>	<i>COD</i>
MS + FS	37950	37869	37849	0.423
R + MS + R × MS	37900	37811	37789	0.427
R + MS + D + MS × D	37916	37786	37754	0.430
R + MS + D + R × D	37906	37776	37744	0.430
R + MS + D	37890	37769	37739	0.430
R + MS + D + R × MS	37896	37767	37735	0.430

Note. Five best-fitting models in Experiment 3, based on *BIC*, plus best-fitting model without removal predictor (first row). Model predictors coded as *R*, Removal; *MS*, Mode Switch; *FS*, Frame Switch; *D*, Detection of repetition. *BIC*, Bayes Information Criterion; *AIC*, Akaike Information Criterion; *COD*, coefficient of determination. Best-fitting model in bold font. See text for details.

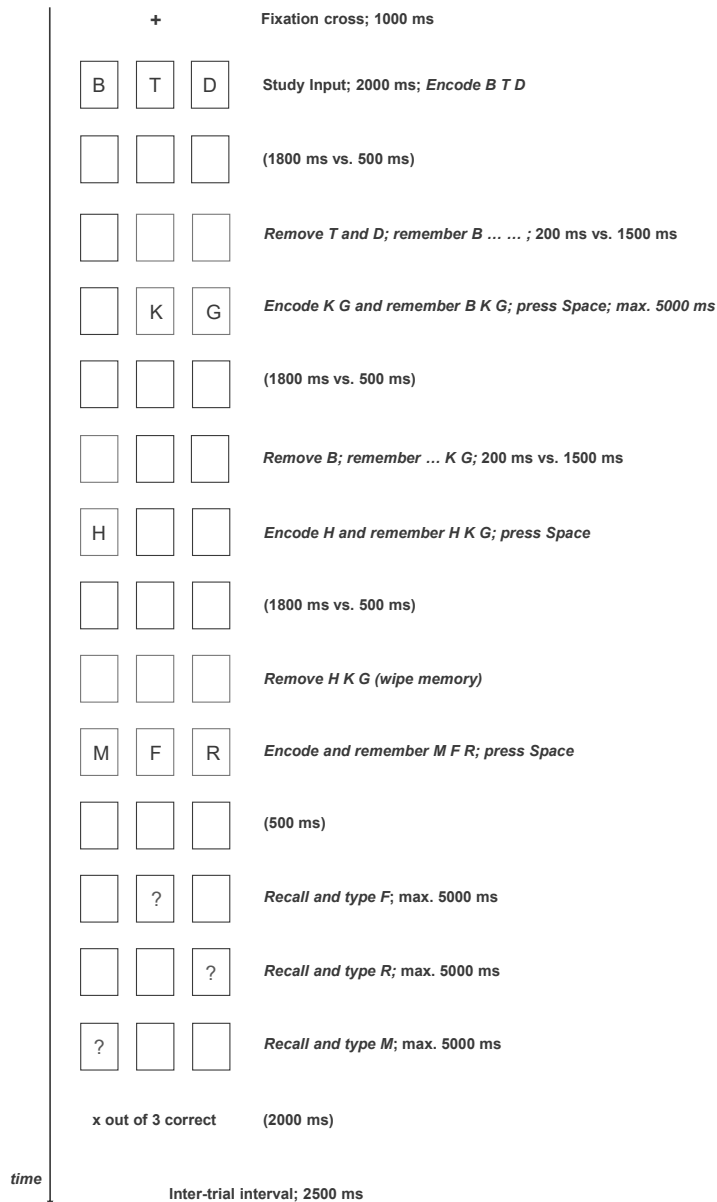


Figure 1: A representative but short trial sequence from Experiment 1, featuring 3 updating steps. Across trials, the number of updating steps ranged from 1 to 21, with a 10% termination probability after each step; this yielded a mean number of approx. 9 updating steps per trial. The duration of the empty-frames interval after each updating step was determined by the length of the subsequent cue-target interval (CTI): If the CTI was short (200 ms), the empty-frames interval was long (1800 ms), if the CTI was long (1500 ms), the empty-frames interval was short (500 ms), and hence the retention interval between updating steps was constant at 2000 ms.

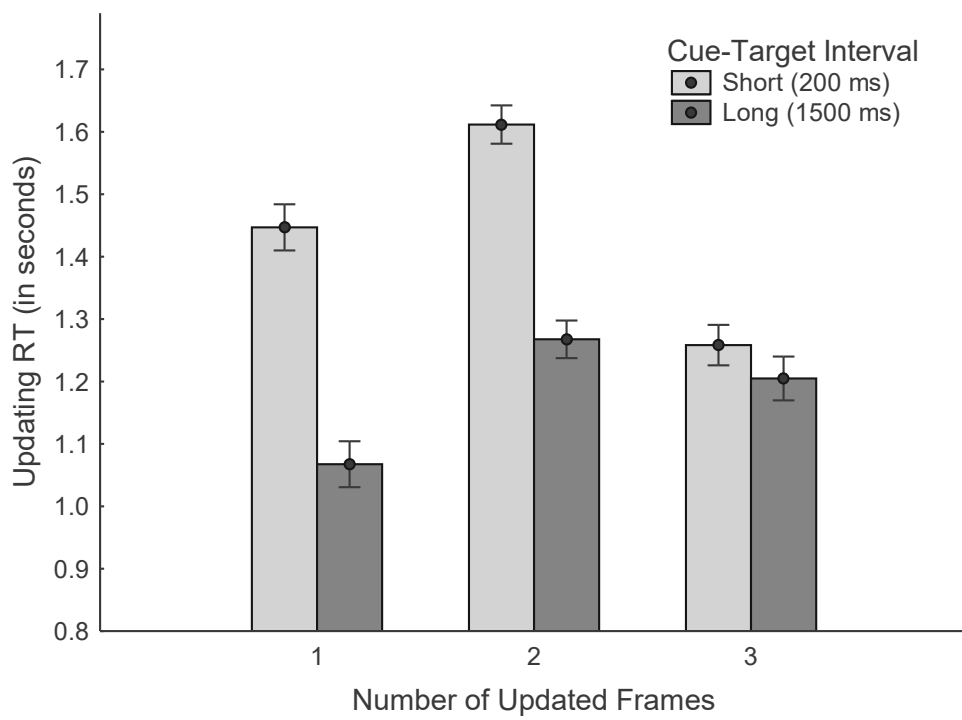


Figure 2: Updating response times from Experiment 1. Vertical bars denote within-subject standard errors of the mean.

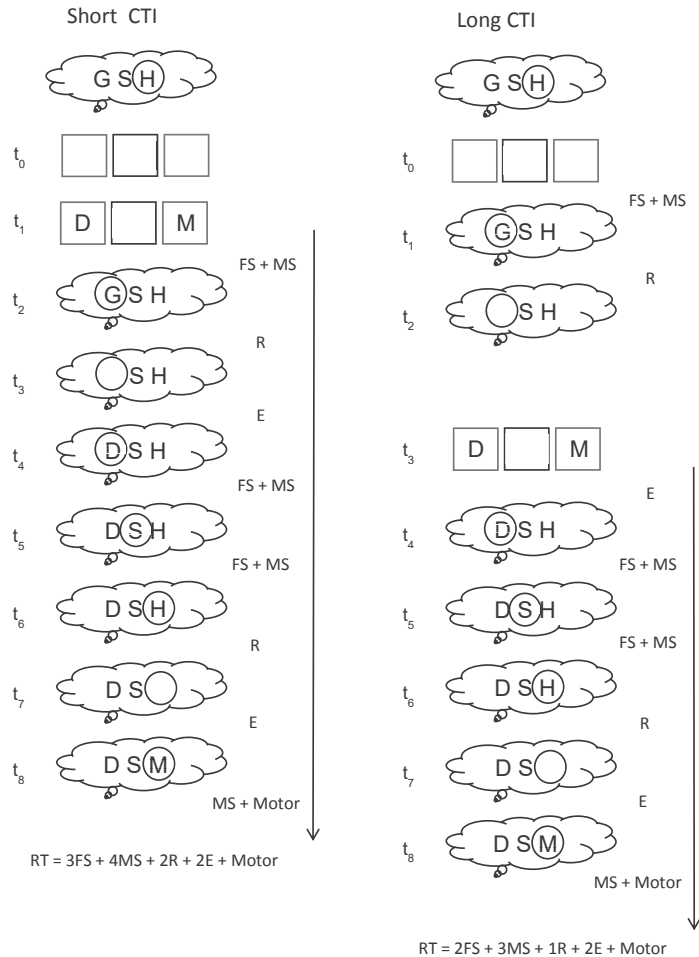


Figure 3: Snapshot of events during an updating step according to the scanning removal model, for short CTIs (left) and for long CTIs (right). Time runs from top to bottom, from t_0 to t_8 ; events on the screen are presented in frames, and events assumed to happen in working memory are presented in thought bubbles. The updating step starts with the state of working memory after the last step, with the focus of attention (bold circle) on the last item, and the mode of operation set to maintenance (continuous line of the bold circle). The updating step commences with a focus switch (FS) to the first item, together with a mode switch (MS) to updating mode (broken-line circle). The first item is removed (R). With a short CTI, the new item is by then available for encoding, and is immediately encoded (E). This is followed by a focus switch to the second item, together with a mode switch to maintenance mode. Nothing is to be done on the second item (except perhaps refreshing it, not illustrated), so the focus shifts to the third item, together with a switch to updating mode. The third item is removed, and the new letter encoded into the third frame. At this point, working memory switches back into its default maintenance mode, and the motor response is made. In the case of a long CTI, all processing steps starting with the first encoding operation wait until display of the new letters. The arrow on the right of each column shows the duration measured as updating RT in each condition.

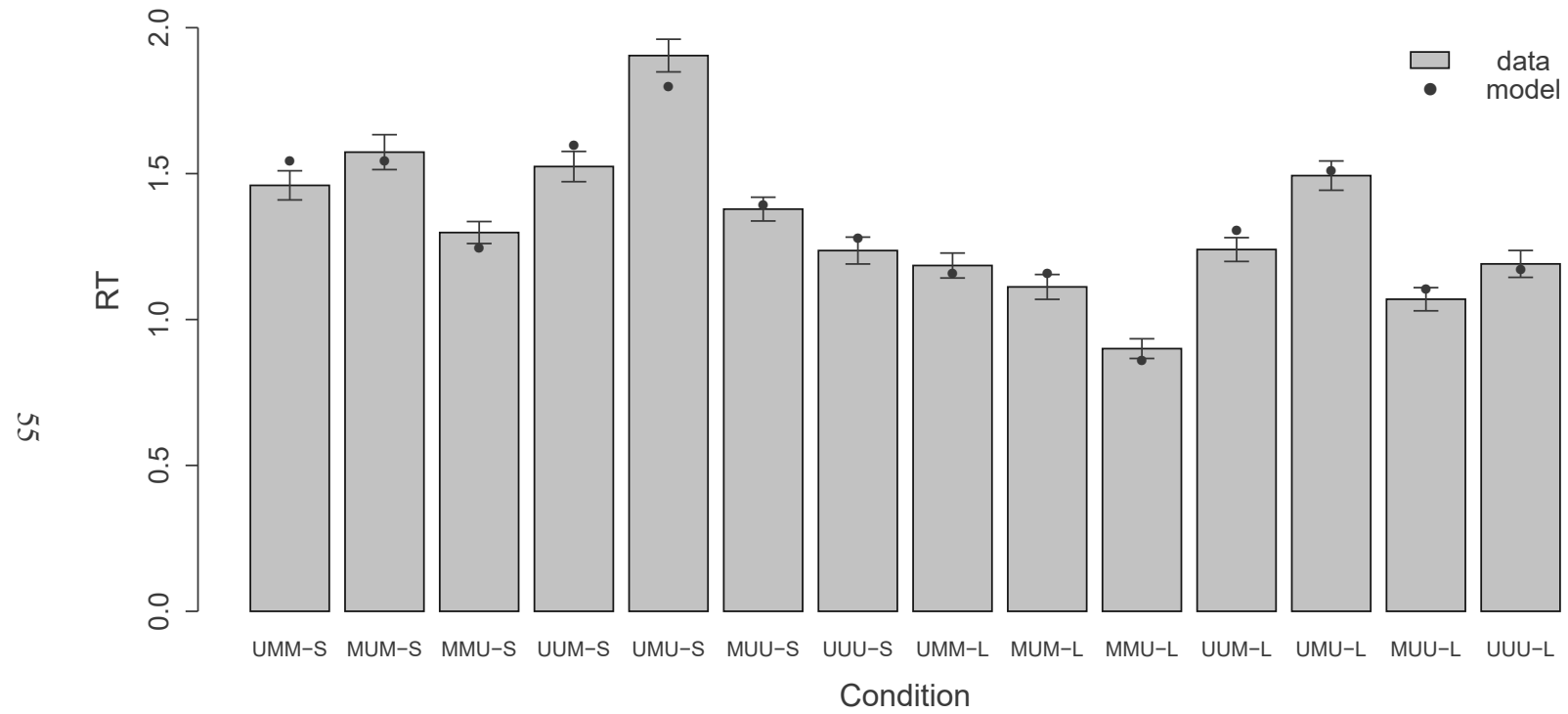


Figure 4: Model fit in Experiment 1. Bars represent observed mean RTs of specific updating conditions, which take position of updated frames into account; condition labels refer to the three frames, the content of which is either maintained (M) or updated (U); S and L refer to the duration of the CTI. Error bars show standard errors of the mean. Dots show predictions of the best-fitting regression model.

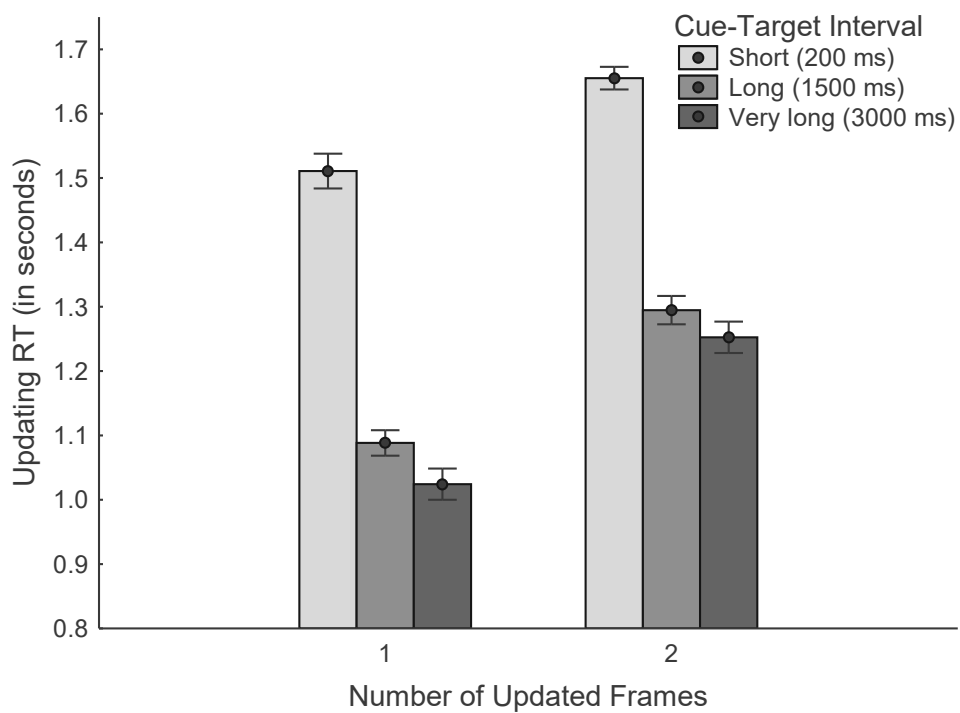


Figure 5: Updating response times from Experiment 2. Vertical bars denote within-subject standard errors of the mean.

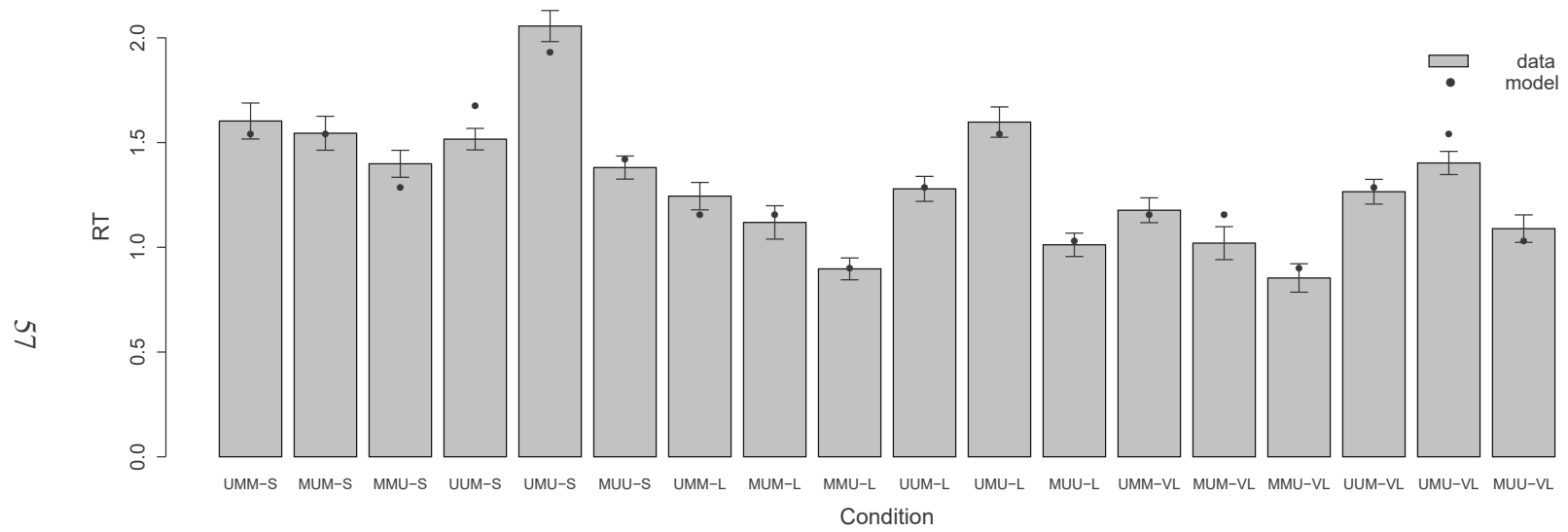


Figure 6: Model fit in Experiment 2. Bars represent observed means of specific updating conditions, which take position of updated frames into account; condition labels refer to the three frames, the content of which is either maintained (M) or updated (U); S and L refer to the duration of the CTI. Error bars show standard errors of the mean. Dots show predictions of the best-fitting regression model.

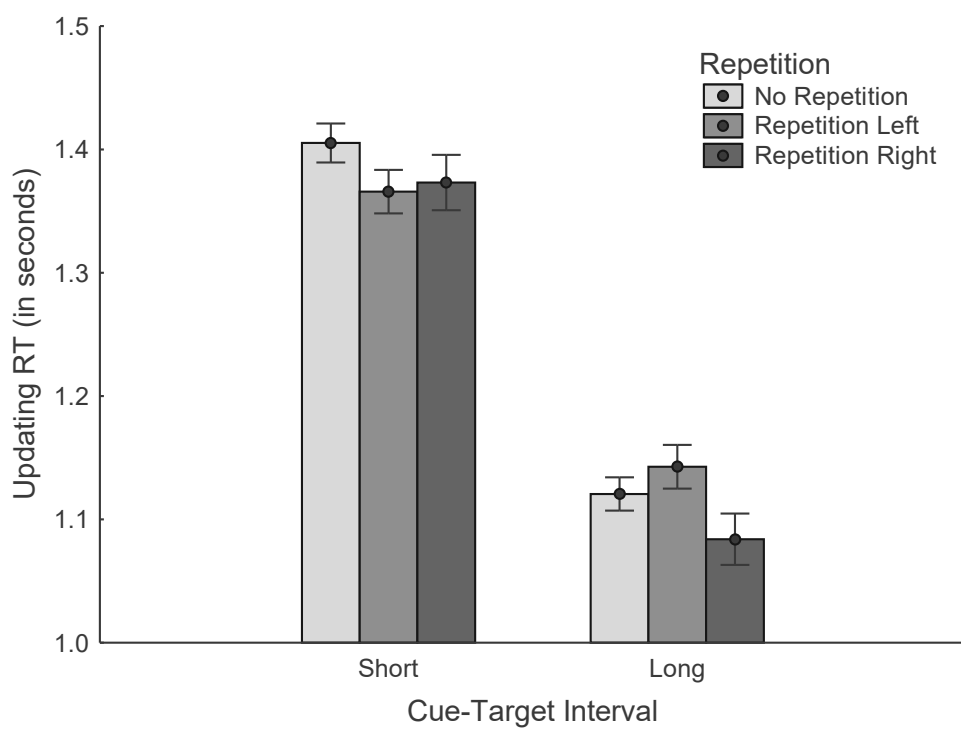


Figure 7: Updating response times from Experiment 3. Vertical bars denote within-subject standard errors of the mean.

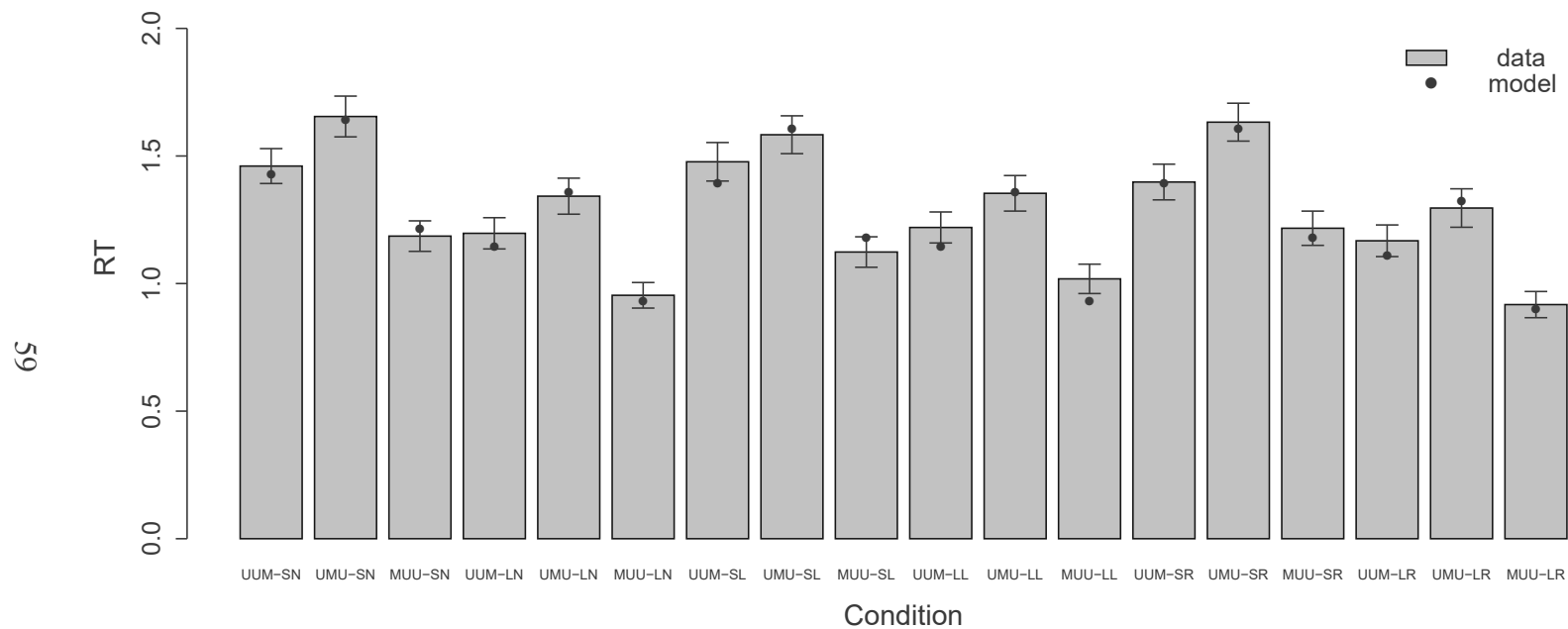


Figure 8: Model fit in Experiment 3. Bars represent observed means of specific updating conditions, which take position of updated frames into account; condition labels refer to the three frames, the content of which is either maintained (M) or updated (U); S and L refer to the duration of the CTI, N, L, and R code no-repetition, repetition in left 'U' frame, and repetition in right 'U' frame. Error bars show standard errors of the mean. Dots show predictions of the best-fitting regression model.