Automatizing Alphabet Arithmetic: I. Is Extended Practice Necessary to Produce Automaticity?

Gordon D. Logan
University of Illinois, Urbana-Champaign

Stuart T. Klapp
California State University, Hayward

Four experiments tested the necessity of extended practice in producing automaticity in an alphabet–arithmetic task in which subjects verified equations of the form A + 2 = C, asking whether C was two letters down the alphabet from A. Experiment 1 trained subjects on 40 alphabet–arithmetic facts for 12 sessions, demonstrating that extended practice was sufficient to produce automaticity. Experiment 2 produced the same degree of automaticity in a single session by having subjects rote memorize 6 facts, suggesting that extended practice is not necessary. Experiments 3 and 4 explored procedural differences between Experiments 1 and 2 to determine what was responsible for the large difference in the time required to develop automaticity. Experiment 3 compared learning rates with different numbers of facts (6, 12, and 18), and found learning rate to depend on the number of presentations of individual items, not on the number of items to be learned. Experiment 4 compared learning by performing the task (as in Experiment 1) with learning by remembering the facts (as in Experiment 2) and found no important differences between them. The results of all 4 experiments cannot be predicted by approaches that define automaticity in terms of resources or by listing properties, although they are readily predictable from theories that assume memory retrieval is the process that underlies automaticity.

Is extended practice necessary to produce automaticity? Can automaticity to be produced without it? Extended practice is certainly sufficient. Many of the properties of automaticity can be produced by extended practice (e.g., Logan, 1978; Shiffrin & Schneider, 1977) or by using materials such as familiar words that have a history of extended practice outside the laboratory (e.g., Neely, 1977). However, sufficiency does not imply necessity. Existing data provide no answer.

Common theoretical approaches to automaticity provide no answer either. Approaches that define atuomaticity in terms of manifest properties, such as speed, effortlessness, and autonomy (property-list approaches), are stipulative or descriptive but not predictive (e.g., Hasher & Zacks, 1979; LaBerge & Samuels, 1974; Shiffrin & Schneider, 1977). They provide no underlying mechanism from which predictions about the properties of automaticity or necessity of extended practice can be deduced.

Approaches that define automaticity in terms of resource theory specify an underlying mechanism, arguing that automatic processing is (relatively) resource free (e.g., Logan, 1978; Posner & Snyder, 1975). The resource mechanism allows properties of automaticity to be deduced: Automatic processing is fast because it is not limited by the availability of resources; automatic processing is effortless because effort is proportional to the amount of resources required; and automatic processing is obligatory because control is exerted primarily by allocating resources, and a process that does not require resources cannot be controlled by allocating resources. However, resource theories do not specify a learning mechanism and so cannot make predictions about the necessity of extended practice (or anything else) in producing automaticity. The answers they provide are to questions other than the one that concerns us here.

Answers are available in several modern theories that argue that the process underlying automaticity is memory retrieval: According to these theories, performance is automatic when it is based on direct-access, single-step retrieval of solutions from memory rather than some algorithmic computation (Logan, 1988b; Newell & Rosenbloom, 1981; Rosenbloom & Newell, 1986; Schneider, 1985; Schneider & Detweiler, 1987). Automatic processing is fast because memory retrieval must race with the algorithm, and the algorithm "screens out" slow and ineffective retrievals. Automatic processing is effortless, relative to the algorithm, because memory retrieval wins the race only when it is reliable enough for solutions to "pop into mind" easily. Automaticity processing is obligatory because memory retrieval is obligatory; memory retrieval wins the race only when associative strength is high and the conditions

for retrieval are near optimal (for discussion of these points, see Logan, 1988a, 1988b; Schneider, 1985).

Automaticity-as-memory theories argue that having traces readily available in memory is a necessary condition for producing automaticity regardless of the amount of practice. Extended practice may be sufficient to produce automaticity insofar as it guarantees that traces will be available, but is not necessary. A sufficient number of traces could be made available by other methods (e.g., deliberate memorization) in a relatively short period of time. What matters is having traces available in memory; how they got there is less important. Extended practice may strengthen automaticity by adding even more traces to memory, and performance may continue to improve indefinitely, but in theory, extended practice is not necessary to produce automaticity (for further discussion of this theme, see Klapp, Boches, Trabert, & Logan, 1991; Logan, 1985).

The purpose of this article is to investigate the necessity of extended practice in producing automaticity. We report four experiments on automatizing performance on a novel task we call alphabet arithmetic. The first experiment was a conventional automaticity experiment, demonstrating that extended practice is sufficient to produce important characteristics of automaticity in this task (also see Klapp et al., 1991). The second experiment provided a test of the necessity of extended practice. Subjects learned alphabet–arithmetic facts by rote memorization and were tested for automaticity in a single session. Evidence of automaticity in a single session would be evidence against the necessity of extended practice. Experiments 3 and 4 explored procedural differences between Experiments 1 and 2 to determine what was responsible for the vast differences in the time required to produce automaticity. Experiment 3 examined the effects of the number of items to be learned, and Experiment 4 examined the method of learning (rote memorization vs. learning by performing the task).

Automaticity and Arithmetic

Children's acquisition of skill at mental addition is a paradigm case of automaticity. Typically, children learn to add with a general counting algorithm, based on knowledge of the sequence of numbers, in which they increment a counter successively by ones for each unit of each addend (Groen & Parkman, 1972). With some experience, they streamline the algorithm, beginning their count with the larger addend and incrementing the counter once for each unit of the smaller addend (the min strategy; see Groen & Resnick, 1977). The counting algorithm is general in that it allows them to add any two numbers, provided they apply it correctly. With further experience, however, they memorize the sums of all possible pairs of single digits, and retrieve the sums directly from memory instead of counting (Ashcraft, 1987; Siegler, 1987). By the time they reach adulthood, addition displays many of the properties of automaticity: It is fast, effortless, and obligatory (LeFevre, Bisanz, & Mrkonjic, 1988; Zbrodoff & Logan, 1986). The primary evidence that skilled addition depends on memory retrieval is phenomenological (most adults think they add by remembering sums), but there are a number of associative interference and facilitation effects that

appear in adults' and older children's data but not in younger children's data, evidencing a transition from counting to remembering (for a review, see Ashcraft, 1982).

There are a number of problems involved in using addition to study the acquisition of automaticity: First, the transition from counting to remembering occurs in young children—in some cases as early as the second grade—whose reaction time performance is notoriously slow and variable. Second, the transition is most apparent in comparing children at different age levels, and that comparison is confounded with large changes in base reaction time. For example, first graders may take twice as long to perform a task as fifth graders (Kail, 1986, 1988). Theoretically interesting effects are often small compared with such large differences in base reaction time. Third, the amount of practice with addition is largely unknown and difficult to control for practical and ethical reasons. It may vary substantially between individuals in the same grade.

Alphabet Arithmetic

To circumvent these problems, we developed an alphabet–arithmetic task in which adults learned to add digits to letters of the alphabet to produce other letters of the alphabet. Adult subjects were required to verify equations of the form A + 2 = C, indicating whether C was two letters down the alphabet from A. Like numerical addition, this task is initially performed by a counting algorithm in which subjects count through the alphabet from the initial letter (e.g., A) for a number of steps determined by the digit addend (e.g., 2) to recover the true sum of the letter and digit (e.g., C), which they compare with the presented sum and report a match or a mismatch. Like counting with numbers, this counting algorithm requires knowledge of the alphabet sequence and the ability to count through the sequence. In principle, any letter and digit can be added by this algorithm, just as any two numbers can be added by the number-counting algorithm and in principle, the counting algorithm can be replaced by memory for alphabet–arithmetic facts, just as number counting is replaced by memory for numerical sums.

Alphabet arithmetic can be studied with adults whose reaction times are fast and relatively invariant (compared with children's), whose base reaction times are not likely to change much over the course of the experiments, and whose experience with the task can be controlled precisely. The counting algorithm can be identified readily in reaction time data: It predicts a linear increase in reaction time as a function of the magnitude of the digit addend, which determines the number of counting steps through the alphabet. Our data show that the slope of the function is typically 400 to 500 ms per count. Memory can be identified almost as clearly in reaction time data: The memorability of problems should not depend on the magnitude of the digit addend, so performance based on (automatic) memory retrieval should produce a slope of zero in the linear function relating reaction time to the magnitude of the digit addend. Moreover, memory-based performance should be fast compared with algorithm-based performance, because in theory subjects should not switch to memory until memory is faster and more reliable than the algorithm (see

Logan, 1988b; Schneider, 1985).[1] The process of automatization should be revealed as a reduction in slope and an increase in speed over practice. Also the counting algorithm and memory retrieval are phenomenologically distinct: Subjective reports can supplement the analyses of reaction time to document the development of automaticity (see Experiment 1; also see Compton & Logan, in press).

Experiment 1

In the first experiment, subjects performed the alphabet-arithmetic task for 16 sessions. They began with 12 sessions of training on 40 facts, learning to add the digits 2, 3, 4, and 5 to the first or second 10 letters of the alphabet. The 13th session involved transfer to the untrained half of the alphabet to test the specificity of what they learned in the first 12. Automaticity-as-memory theories predict item-specific learning and, thus, poor transfer to untrained items. Session 14 returned them to baseline, using only the facts learned in the first 12 sessions.

Session 15 involved the whole alphabet (all 20 letter addends), randomly mixing trained and untrained items. Its purpose was to test an intentionality criterion of automaticity. According to Logan's (1988b) instance theory, memory retrieval is obligatory, occurring whether or not the subject intends it. Subjects may intend to rely on memory retrieval on Sessions 12 and 14, when only old items are presented, but they must intend to count on Session 15 because memory retrieval will not support responses to new items. If retrieval is obligatory, old items should be processed automatically even when subjects are prepared to count. There should be no loss of automaticity for old items on Session 15 (relative to Sessions 12 and 14).

Session 16 was conducted 1 month after Session 15 as a test of the retention of automaticity. Common lore and a few experiments (Feustal, Shiffrin, & Salasoo, 1983; Healy, Fendrich, & Proctor, 1990; Kolers, 1976) suggest that automatic skills are retained for long periods of time. The intention here was to see whether the automaticity acquired in the training period would survive a month of inactivity.

In addition to performing the alphabet–arithmetic task, subjects recited the alphabet aloud as quickly as possible each day, and their recitation rate was calculated. One purpose was to provide a control measure of the speed at which subjects could sequentially access the alphabet to compare with their counting rates in alphabet arithmetic. Previous research has shown that subjects take 100 to 125 ms per letter to recite the alphabet, both overtly and subvocally (Landauer, 1962). Pilot data suggested that early in practice, alphabet–arithmetic counting rates were around 400 ms per count. If training simply increased the rate of sequential access to the alphabet, then after practice the alphabet–arithmetic counting rate should approach but not exceed the alphabet recitation rate. However, if training permits retrieval of sums from memory (and retrieval time is the same for all sums regardless of digit addend), then the alphabet–arithmetic counting rate should approach 0 ms per count, exceeding the recitation rate by a wide margin.

A second purpose of the overt recitation controls was to see whether training on alphabet arithmetic affected subjects' ability to access the alphabet sequentially. One interpretation of automatizing alphabet arithmetic is that the links between successive letters of the alphabet become strengthened by training, so subjects can access the next letter in the sequence more rapidly. If that were the case, we may find the overt recitation rate speeding up as subjects gain skill on the alphabet–arithmetic task.

At the end of each session (except for Session 15) we asked subjects whether they performed the task by counting or remembering, and we asked them to estimate the proportion of trials performed by counting and remembering. These subjective reports were used to converge on conclusions drawn from the reduction in estimated counting rates, as evidence of the switch from counting to remembering with practice predicted by automaticity-as-memory theories.

Method

Subjects. The subjects were 8 graduate and undergraduate students from Purdue University who were paid $3.50 per session for participating.

Apparatus and stimuli. The stimuli were presented on a Digital Equipment Corporation VT240 monochrome monitor controlled by a PDP 11/23 laboratory computer. Responses were recorded on the keyboard associated with the monitor. The stimuli were capital letters, digits, the plus symbol (+), and the equal sign (=). During training (Sessions 1–12, 14, and 16), the letter addends were A through J for half of the subjects and K through T for the other half. In transfer sessions (Sessions 13 and 15), the other half of the alphabet was used. For all subjects in all sessions, the digit addends were 2, 3, 4, and 5. The "answers" were either the correct sum of the letter and digit addends or the sum plus one letter (e.g., A + 2 = D) or the sum minus one letter (e.g., A + 2 = B). The equations were presented horizontally in the center of the monitor screen. Each symbol (letter, digit, +, =) was separated by a space, so the entire equation occupied eight character spaces on the screen.

Procedure. Each session involved 480 trials. All sessions except Session 15 involved six replications of 80 problems, produced by the factorial combination of 10 letter addends, 4 digit addends, and true versus false. Session 15 involved three replications of 160 problems formed by the factorial combination of 20 letter addends (10 trained and 10 untrained), 4 digit addends, and true versus false. Each trial began with a fixation point exposed for 500 ms in the center of the screen. It was extinguished and replaced immediately by an equation, which remained on the screen until the subject responded. When the subject responded, the equation was extinguished, and the screen remained blank for a 1.5-s intertrial interval. Subjects were allowed brief breaks every 120 trials.

---

[1] Note that memory-based performance can be faster than algorithm-based performance even if memory-retrieval time is not faster on average than algorithm-completion time. In Logan's (1988b) instance theory, several memory traces race against the algorithm and the fastest "runner" wins. Memory-based performance depends on the distribution of minima sampled from the original retrieval-time distribution, and the distribution of minima will necessarily be faster than the parent distribution from which minima are sampled. The minimum retrieval times can be much faster than the algorithm even if the mean retrieval time is slower.

Subjects recited the alphabet out loud six times each session: three times before performing the alphabet-arithmetic task and three times afterward. They were told to proceed through the alphabet as quickly as they could but to say each letter distinctly. At the end of the session, after their final recital of the alphabet, subjects estimated the percentage of trials on which they solved the alphabet-arithmetic task by counting and the percentage they solved by remembering the answer. They were not told that their percentages should add up to 100.

Subjects were given complete instructions on the first session and reviews as needed on subsequent sessions. Initially, the alphabet-arithmetic task was described as an analog of numerical addition, and the procedure of counting through the alphabet was described to them using an example from the untrained half of the alphabet. They were told they would see equations and their task was to indicate whether or not the equation was true by pressing one key or the other. They were told to respond as quickly as possible but to keep their error rate below 10%. Then they began the task with no practice or warm-up.

Half of the subjects were trained on the first half of the alphabet (the letters A through J), and half were trained on the second half (the letters K through T). Half had false problems in which the answers were true plus one letter (e.g., A + 2 = D), and half had false problems in which the answers were true minus one letter (e.g., A + 2 = B). Half pressed the right key to indicate the equation was true and the left key to indicate it was false, and half did the opposite. These counterbalanced conditions were combined factorially, and 1 subject was assigned to each combination of conditions.

Subjects were trained on one half of the alphabet for 12 sessions, transferred to the other half on Session 13, returned to the trained half on Session 14, and transferred to a random mixture of trained and untrained problems on Session 15. Session 16 was scheduled 30 days after Session 15, and involved the half of the alphabet experienced during training. The procedure was the same each session. We forgot to ask subjects to distinguish between trained and untrained stimuli in their subjective reports on Session 15, so we discarded their estimates.

Data analysis. The data were analyzed in several analyses of variance (ANOVAs). Reaction time data for Sessions 1 and 12 were compared in a 2 × 4 × 2 × 2 (Sessions × Addend × True vs. False × Groups[false answers true plus 1 or true minus 1]) ANOVA. Slopes of the linear function relating reaction time to the magnitude of the digit addend were analyzed in a 12 × 2 × 2 (Sessions × True vs. False × Groups [true plus 1 vs. true minus 1]) ANOVA. Reports of counting and remembering were analyzed in separate 12 × 2 (Session × Groups [true plus 1 vs. true minus 1]) ANOVAs. Transfer to new items on Session 13 was analyzed in a 3 × 2 × 2 (Session 12 vs. 13 vs. 14 × True vs. False × Groups [true plus 1 vs. true minus 1]) ANOVA. The effects of mixing old and new items on Session 15 were analyzed in a 2 × 2 × 2 (Old vs. New × True vs. False × Groups [true plus 1 vs. true minus 1]) ANOVA on the slopes and a 4 × 2 × 2 × 2 (Addend × Old vs. New × True vs. False × Groups [true plus 1 vs. true minus 1]) ANOVA on reaction times. Retention of automaticity on Session 16 was tested in a 2 × 2 × 2 (Session 14 vs. 16 × True vs. False × Groups [true plus 1 vs. true minus 1]) ANOVA on the slopes and in separate 2 × 2 (Session 14 vs. 16 × Groups [true plus 1 vs. true minus 1]) ANOVAs on reports of counting and remembering.

Results and Discussion

Training (Sessions 1–12). Accuracy was high, averaging 94.5% over the 12 training sessions. Accuracy did not vary much with digit addend (96%, 95%, 93%, and 94% for Addends 2–5, respectively), true versus false equations (95% vs. 94%, respectively), or practice (95% on average in Session

1 vs. 94% on average in Session 12). No ANOVAs were performed on the accuracy data because there were no important trends apparent in the means and because accuracy was so near ceiling.

Mean reaction times for true and false equations are displayed as a function of the digit addend for the 1st and 12th session in Figure 1. There is a sharp contrast between sessions both in overall reaction time, $F(1, 6) = 118.90$, $p < .01$, $MS_e = 667,426.50$, for main effect of sessions, and in the effect of digit addend, $F(3, 18) = 45.58$, $p < .01$, $MS_e = 60,032.26$, for Sessions × Addend interaction. In the first session, reaction time increased strongly with the magnitude of the digit addend. The function was clearly linear, with a slope of 486 ms per count; the linear trend was highly significant, $F(1, 18) = 313.59$, $p < .01$, $MS_e = 60,032.26$. By the 12th session, the function flattened considerably. The slope reduced to 45 ms per count and the linear trend was not significant, $F(1, 18) = 2.62$, $MS_e = 60,032.26$. These data suggest relative extremes of automatization: Session 1 performance was dominated by the counting algorithm, whereas Session 12 performance was dominated by memory retrieval. Subjective reports were consistent with this analysis. Subjects reported counting 93% of the time in Session 1 and 28% in Session 12; they reported remembering 7% of the time in Session 1 and 64% in Session 12.

The transition between Session 1 and Session 12 is shown in Figure 2, which plots the slopes of the linear function relating reaction time to the magnitude of the digit addend each session. On average, at least, the transition was gradual, $F(11, 66) = 27.84$, $p < .01$, $MS_e = 13,223.14$. The slope decreased quickly in the first few sessions and appeared to reach asymptote by the 12th session. The reduction in slope could be interpreted as an improvement in the speed of the counting algorithm instead of a transition from counting to remembering. However, three sets of data argue against improved counting. First, the slopes on Session 12 averaged 45 ms per count, which is too fast to be mediated by sequential access to the alphabet, as measured by the rate of reciting the alphabet. Mean recital rates were calculated by dividing the
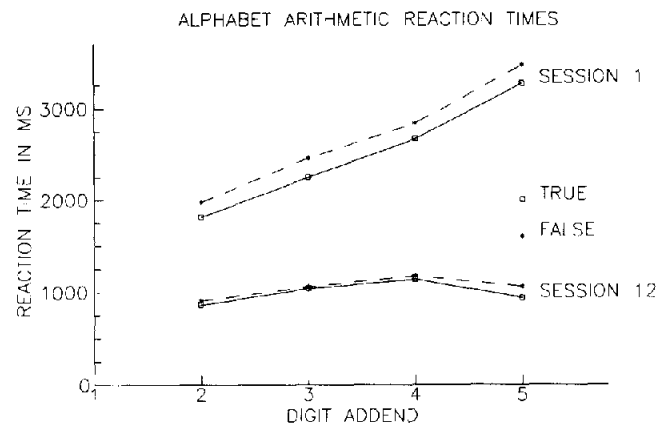

Figure 1. Mean reaction times for Sessions 1 (top lines) and 12 (bottom lines) in Experiment 1 as a function of digit addend (true vs. false equation is the parameter; true = solid lines, false = broken lines).
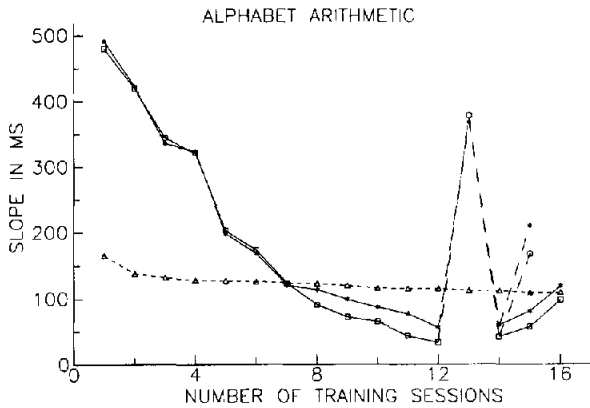
Figure 2. Mean slopes of the linear function relating reaction time to the magnitude of the digit addend in Experiment 1 as a function of training sessions. (True vs. false equations is the parameter; true = boxes; false = stars. Solid lines connect points that represent training on the initial 10 problems; coarsely broken lines connect points that represent performance on the 10 new problems. Finely broken lines connecting triangles represent the mean rate per letter of reciting the alphabet overtly.)

mean of the six recital times each day by 26. Recital rate averaged 115 ms per letter on Session 12, $t(7) = 2.57, p < .05$, $MS_e = 27.13$.

Second, subjects reported having experienced a transition from counting to remembering. The average percentage of trials counted and remembered are plotted for each session in Figure 3. Reports of counting dropped steadily from Session 1 to 12, $F(11, 66) = 28.35, p < .01, MS_e = 130.29$, while reports of remembering increased steadily, $F(11, 66) = 20.60$, $p < .01, MS_e = 140.40$.

Third, the means and standard deviations of reaction times decreased as power functions of the number of presentations, and both power functions had the same exponent, as predicted by the instance theory of automatization (also see Cohen, Dunbar, & McClelland, 1990). The fits are reported in detail in Logan (1988b). The instance theory describes performance as a race between the algorithm and each trace of each prior
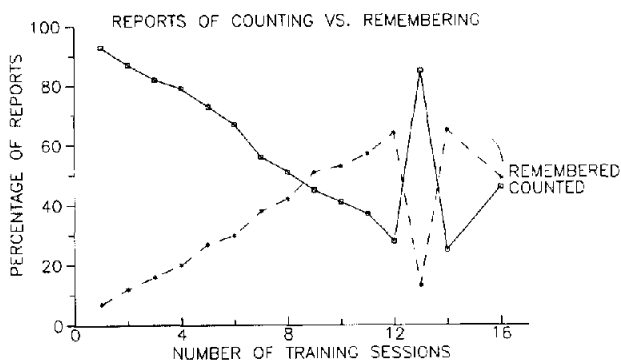

Figure 3. Mean reports of percentage of trials counted (solid lines) and remembered (broken lines) in Experiment 1 as a function of session.

exposure to an item in memory; the first one to finish determines reaction time. On the nth presentation, there are n runners in the race (i. e., one algorithm and $n - 1$ memory traces). Assuming the same distribution of retrieval times for each trace and a roughly similar distribution of finishing times for the algorithm, the problem amounts to selecting the minimum of n samples from the same distribution and analyzing how the minimum behaves as n increases. This is a well-studied problem in the statistics of extreme values, and it is possible to prove that for distributions like reaction time distributions, the entire distribution of minimum values decreases as a power function of n. This implies that the mean and standard deviation will decrease as power functions of n with the same exponent. The prediction is developed formally in Logan (1988b). For now, the important point is that the prediction follows from the theory's assumptions about memory retrieval and the transition from the algorithm to memory retrieval with practice.

The data from Addends of 2, 3, and 4 were well fit by power functions constrained to have the same exponent, consistent with the instance theory. The data from Addend 5 were less well fit, primarily because they showed a discontinuity after about 24 exposures (after Session 4; see Logan, 1988b, Figures 9 and 10). The discontinuity can be seen in the slopes plotted in Figure 2: The slope dropped 121 ms from Session 4 to Session 5, the largest session-to-session drop in the training period. No theory of automatization, including the instance theory, can account for this discontinuity because they all predict power-function reductions in mean reaction time, and power functions are continuous.

An explanation for the discontinuity came from subjective reports: Subjects reported developing mnemonics to commit the Addend 5 problems to memory to avoid counting. Thus, Addend 5 problems may have been encoded with more effective mnemonic strategies than Addend 1 to 4 problems. There may have been a shift from relatively "automatic" encoding of the counting episode to deliberate mnemonic encoding strategies around Sessions 4 and 5. It is not clear why this would have happened between Sessions 4 and 5. Subjects served in four sessions a week, so Sessions 4 and 5 were typically separated by a weekend, which may have provided time for reflection on how to deal with the task. Regardless, a shift in mnemonic strategies during automatization is broadly consistent with the idea that memory underlies automaticity. It suggests intriguing possibilities for future experiments.

Transfer to new items (Sessions 13-15). On Session 13, all subjects switched to items from the untrained half of the alphabet. The slope, plotted in Figure 2, increased from 45 ms on Session 12 to 373 ms on Session 13. Reports of counting rose from 28% on Session 12 to 85% on Session 13; reports of remembering dropped from 64% to 13%. Apparently, subjects reverted to counting with the new items. (Accuracy remained relatively constant, increasing only slightly from 94% on Session 12 to 95% on Session 13.)

The differences between new items on Session 13 and the old items on Sessions 12 and 14 were documented by ANOVAs: A main effect of sessions was found in an ANOVA on the slopes (Sessions 12, 13, and 14 × True vs. False), $F(2, 12)$

$= 28.54, p < .01, MS_e = 19,808.01$, in an ANOVA on reports of counting (Sessions 12, 13, and 14 × True Plus 1 vs. True Minus 1 Groups), $F(2, 12) = 58.53, p < .01, MS_e = 153.26$, and an ANOVA on reports of remembering (Sessions 12, 13, and 14 × True Plus 1 vs. True Minus 1 Groups), $F(2, 12) = 35.57, p < .01, MS_e = 199.83$.

The difficulty with new items is not consistent with the idea that subjects learned to count faster in the first 12 sessions. If they had, they could have counted new items as easily as old; transfer should be near perfect. The data are more consistent with automaticity-as-memory theories, which predict poor transfer to new items (subjects learn only the items they experience; only those or closely similar items will evoke responses from memory).

Even though transfer was poor, some transfer was observed. The slope on Session 13 (373 ms) was midway between the slope on Session 2 (421 ms) and the slope on Session 3 (340 ms). It did not approach the initial value in Session 1 (486 ms). Subjective reports of counting and remembering on Session 13 also fell between the values observed on Sessions 2 and 3 and were less extreme than the values observed on Session 1. Subjects may have learned to count more rapidly in the first few sessions, or they may have learned mnemonic strategies during training that they applied immediately to the new problems in Session 13.

On Session 14, all subjects returned to baseline, experiencing six more presentations of the items on which they were originally trained. The slope dropped to 51 ms, not different from the 45-ms value observed on Session 12 (see Figure 2). Reports of counting dropped to 25%, while reports of remembering rose to 65%, very close to Session 12 values (see Figure 3). Mean accuracy was 95%.

The purpose of Session 15 was to determine whether memory retrieval met the intentionality criterion of automaticity. According to Logan's (1988b) instance theory, retrieval is obligatory, occurring whether the subject intends to retrieve or intends to count. New and old items were mixed randomly on Session 15 so that subjects would begin each trial with the intention to count. If retrieval is obligatory, old items should be processed just as automatically on Session 15 as on Session 14 when subjects likely intended to remember. Slopes on Session 15 were calculated separately for old and new items, and the slope for old items was 69 ms per count, not significantly larger than the 51-ms value observed on Session 14. These data are consistent with the idea of obligatory retrieval. Apparently, automaticity was sustained.

The slope for new items on Session 15 was 189 ms per count, between the values observed on Sessions 5 and 6. The slope for new items was not significantly larger than the slope for old items in an analysis of slopes, $F(1, 6) = 1.89, MS_e = 60,320.93$, but it is hard to have confidence in the null hypothesis given the large difference. In another analysis, mean reaction times were significantly longer for new than for old items, $F(1, 6) = 64.77, p < .01, MS_e = 128,311.06$. Mean accuracy was 96% for new items and 96% for old items.

Retention (Session 16). Session 16 was scheduled 1 month after Session 15. All subjects were tested on the half of the alphabet on which they were initially trained. The slopes rose from 51 ms per count on Session 14 (the last full session with

the initially trained items) to 109 ms per count on Session 16 (see Figure 2). The difference approached statistical significance, $F(1, 6) = 4.68, p < .10, MS_e = 5,631.89$. Reports of counting increased from 25% on Session 14 to 46% on Session 16, $F(1, 6) = 8.74, p < .05, MS_e = 192.29$, and reports of remembering decreased from 65% to 49%, $F(1, 6) = 3.89, p < .10, MS_e = 267.40$ (see Figure 3). Mean accuracy was 95%.

The slope on Session 16 was between the values observed on Sessions 7 and 8; reports of counting were between the values observed on Sessions 8 and 9; and reports of remembering were close to the value on Session 9. This suggests a loss of four or five sessions of training over the retention interval or, conversely, a savings of eight or nine sessions. These estimates are necessarily variable, however, and cannot support firm conclusions. We may have observed some loss of automaticity; we did observe substantial retention (cf. Feustal et al.,1983; Healy et al., 1990; Kolers, 1976).

Conclusions. Several aspects of the data provide evidence of the development of automaticity in the alphabet–arithmetic task, suggesting that it reflects a transition from an initial counting algorithm to memory for past problems. First, the slope decreased from 486 ms per count on the first session to 45 ms per count on Session 12. Counting produces a substantial slope; remembering produces a slope of zero, provided all items are equally memorable. Second, subjective reports indicated a transition; reports of counting dropped and reports of remembering increased from Sessions 1 to 12. Third, transfer to new items was poor, as would be expected if subjects had memorized only the items they experienced during training. Fourth, in another report, the means and standard deviations from the first 12 sessions of the present experiment were well fit by power functions constrained to have the same exponent, as predicted by the instance theory of automatization (Logan, 1988b), reflecting a transition from an initial algorithm to memory-based responding. Finally, when old and new items were mixed randomly in Session 15, subjects maintained their speed and flat slopes for the old items, while they responded more slowly and with steeper slopes to new items, suggesting that the retrieval of old facts was obligatory, as the instance theory predicts (Logan, 1988b).

The data were characteristic of automatization in that automaticity took a long time to develop, as it does in other paradigms with other stimuli (e.g., Shiffrin & Schneider, 1977). Extensive practice may be typical of situations in which automaticity is observed, and it may be sufficient to produce automaticity in many cases, but it is not necessary, according to automaticity-as-memory theories. What is necessary is having traces available in memory, and that may occur after one trial or 10,000 trials.[2] It may not matter much how the

---

[2] There are two senses in which performance could be automatic after one trial, depending on whether automaticity is assessed trial by trial or session by session: Trial-based automaticity follows from Logan's (1988b) instance theory in which traces race against the algorithm. Random variability could allow a single trace to beat the algorithm on a single trial. Performance on that trial would be considered automatic, even though performance on subsequent trials may not be (i.e., the algorithm may be faster than memory retrieval

traces got there. They may have been acquired through experience on the task or by rote memorization. (This insight motivated the next experiment.) Extensive practice may guarantee enough exposure that traces are available in memory, but it is not logically necessary.

Experiment 2

Experiment 2 asked whether extended practice is necessary to develop automaticity, and whether practice on the task itself is necessary. Subjects were trained on alphabet–arithmetic facts, but not by performing the verification task. They learned the facts by rote memorization and then transferred to the verification task. The slope of the function relating verification time to the magnitude of the digit addend was the primary index of automatization. Half of the subjects verified the facts they had just learned; half verified untrained facts. If practice on the (verification) task itself is necessary for automatization, there should be no difference between groups. Both groups should produce large slopes on the order of those observed in Session 1 of Experiment 1. However, if practice on the task is not necessary, if having the facts in memory by other means is sufficient, then the group with trained facts should show flat slopes characteristic of automaticity, similar to those observed in Sessions 12 and 14 of Experiment 1, whereas the group with untrained facts should show steep slopes.

The hypothesis that extensive practice is necessary was tested by limiting training to one session. The whole experiment—training and transfer—was finished in less than an hour. The training itself took less than 15 min. That is two orders of magnitude less than training in many studies of automaticity (e.g., Shiffrin & Schneider, 1977). If several hours of practice are necessary for automatization, then both groups should show steep slopes like those in Session 1 of Experiment 1. However, if only practice sufficient to commit the items to memory is necessary, then automaticity should occur. The group with trained facts should show a shallow slope; the group with untrained facts should not.

Method

Subjects. The subjects were 16 Introductory Psychology students from California State University, Hayward, who participated as one option of a course requirement. All claimed to be native speakers of English.

Apparatus and stimuli. The stimuli were presented on the monitor of an Apple II computer. Responses were recorded from the F and J keys on the computer keyboard. The stimuli were the letters *B*,

---

on subsequent trials). Session-based automaticity is a more conventional definition in which performance must be based on memory retrieval for the vast majority of trials to be considered automatic. Nevertheless, performance could be automatic in the session-based sense after a single trial if subjects rehearsed the solution between trials (or committed it to memory in some other manner) so that enough traces were available on subsequent trials for memory to win the majority of the races. Session-based automaticity may be implausible after a single trial, but it is not in principle impossible.

N, H, and *T,* the digits *2, 3,* and *4,* the plus symbol (+) and the equal sign (=) displayed horizontally in the center of the screen as in Experiment 1. The letters were divided into two sets (B and N vs. H and T) and paired with each digit addend to produce two sets of six facts. In the verification task, each fact was paired twice with its true answer and once with an answer that was true plus one (e.g., B + 2 = E) and once with an answer true minus one (e.g., B + 2 = C), producing two sets of 24 items.

In the verification task, each trial began with the display "GET READY" for 1.3 s, followed by a blank display for 1.3 s, and then the problem which remained on until the subject responded. A feedback display appeared immediately after the response, containing "CORRECT" or "ERROR" as appropriate. If reaction time was less than 3 s, the display "TIME WAS GOOD" appeared below the accuracy feedback. For reaction times between 3 and 4.6 s, the display was "TIME WAS OK, BUT SHOULD BE FASTER." Slower responses received "TIME WAS TOO SLOW." The feedback display remained on for 2.5 s and was followed by a 3.5-s interval between trials in which the display was blank.

Each letter and digit subtended about 0.6 degrees of visual angle on the CRT.

Procedure. Subjects were trained one of the two sets of six facts. Half of the subjects were trained on one set and half on the other. Subjects were shown a list of the six facts in their set, and were told that the facts had been generated by the counting algorithm used in Experiment 1: counting up the alphabet a number of steps determined by the digit addend. They were asked to study the list until they had mastered all six facts. Then they were tested with a set of six "flash cards" presenting the letter and digit addends of each fact, to which the subject was to respond by uttering the correct answer (e.g., if given *B + 2 = ?,* the subject would say *D*). The six cards were shuffled to randomize their order, shown one at a time to the subject, then shuffled and shown again as many times as necessary for the subject to pass through the deck three times in succession without error.

Subjects were then pretrained for responding on the verification task. They were tested on 20 trials in which the word *yes* or *no* appeared in the center of the screen. Half of the subjects in each group were instructed to press the F key if *yes* appeared and the J key if *no* appeared, and half were told the opposite. After pretraining, they were tested with the flash cards until a criterion of three successive error-free passes through the six facts had been achieved for a second time. After a brief rest, subjects received one more pass through the six flash cards just before the transfer task. The maximum time spent studying the facts was about 15 min.

At transfer, subjects performed the verification task on the computer. Half of the subjects were tested on the facts on which they were trained and half were tested on untrained facts. Each subject performed 48 verification trials.

Results and Discussion

Each subject experienced eight trials at each combination of digit addend and true versus false equation. In an initial analysis, median reaction times were calculated in each cell and submitted to a 2 × 3 × 2 (Groups [trained facts vs. untrained facts] × 3 Addend × True vs. False Equation) ANOVA. The difference between true and false equations was not significant, $F(1, 14) = 1.67$, $MS_e = 779,971.01$, and there were no significant interactions involving true and false equations (all $Fs < 1.0$). Consequently, the data were collapsed across true and false equations by calculating mean reaction times at each level of digit addend for each subject. The means of these means appear in Figure 4.

The slope of the linear function relating reaction time to the digit addend was 447 ms per count for the group tested on untrained facts, slightly larger than the slope observed on Session 2 of Experiment 1. By contrast, the slope for the group tested on trained facts was slightly negative (-52 ms), suggesting automaticity. The contrast was confirmed in a 2×3 (Group [trained facts vs. untrained facts] $\times$ Addend) ANOVA on the mean reaction times: The interaction between training group and addend was significant, $F(2, 28) = 8.74$, $p < .01$, $MS_e = 121,693.31$. A test for linear trend was highly significant in the group tested on untrained facts, $F(1, 28) = 26.27$, $p < .01$, but not in the group tested on trained facts, $F(1, 28) < 1$, both MS_es $= 121,693.31$. An ANOVA on the median reaction times produced nearly identical F ratios, supporting exactly the same conclusions as the ANOVA on the means.

Accuracy was lower for untrained facts than for trained facts and decreased with the digit addend, particularly for untrained facts. The means across subjects and true versus false problems were 92%, 85%, and 82% for untrained facts with addends of 2, 3, and 4, respectively, and 91% 89%, and 90% for trained facts with addends of 2, 3, and 4, respectively. Essentially, the accuracy data corroborate the reaction times.

These results suggest that automaticity can be attained with less than 15 min of training, and that automaticity can be obtained without practice on the criterion task as long as the relevant facts are in memory. Both results are predictable from automaticity-as-memory theories. They claim that having traces available in memory is necessary for automaticity to occur; they do not claim that any particular route to memory is necessary or any particular "consolidation period" is necessary.

The present results are remarkable in contrast with Experiment 1. In less than an hour, these subjects were able to attain a level of automaticity that took Experiment 1 subjects 12 sessions (5,760 trials) to attain. The experiments differed in two important respects: how many facts were learned and how the facts were learned. There were 40 facts to learn in Experiment 1 (counting only true equations) and 6 in Experiment 2. It may take more time to learn 40 facts than to learn 6, hence the difference in the apparent rate of automatization. Experiment 3 addressed this possibility.
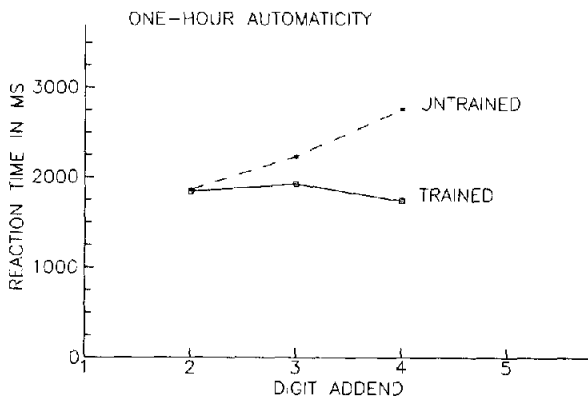

Figure 4. Mean reaction times for trained (solid line) and untrained (broken line) facts in Experiment 2 as a function of digit addend.

The method of learning was different in Experiment 1 and Experiment 2. In Experiment 1, subjects learned the facts while performing the task (learning by doing), whereas in Experiment 2 subjects learned the facts by rote memorization. Rote memorization may be a more effective way to learn the facts than learning by doing. Experiment 4 addressed that possibility.

Experiment 3

Experiment 3 had two main purposes. The first was to determine whether the rate of automatization depends on the amount to be learned. Subjects in Experiment 2 had only six items to learn, whereas subjects in Experiment 1 had 40. In Experiment 3, different groups of subjects learned 6, 12, and 18 alphabet–arithmetic facts by performing the verification task. Subjects practiced for several sessions until each fact had been presented in its true and its false version 96 times (for 6 and 12 facts) or 80 times (for 18 facts). In Experiment 1, performance approached asymptote after 72 presentations of each item, so the practice schedules in this experiment should be sufficient to produce automaticity. The slopes of the functions relating reaction time to the magnitude of the digit addend were analyzed to assess learning rates.

The second purpose of Experiment 3 was to assess the effect of similarity on transfer to facts not experienced in training. Subjects trained on 6 facts and 12 facts were transferred to a set of 18 facts. Six were old facts, experienced in training (old digits, old letters), 6 were new facts about the six old letters (new digits, old letters), and 6 were new facts about six new letters (new digits, new letters). The contrast between old–old and new–new items replicates the transfer conditions in Experiment 1, where poor transfer was found to new items from the other half of the alphabet. The contrast between new–old and new–new items tests the effect of similarity. New–old items share letter addends with old–old items, and so should be similar to old–old items; new–new items share no letter addends with old–old items, and so should be dissimilar. If similarity facilitates learning (e.g., by providing an existing associative structure on which to build new associations), then new–old slopes should be shallower than new–new slopes. New–old slopes may even approach old–old slopes. However, similarity could also inhibit learning (e.g., by providing more opportunities for interference), so new–old slopes may be even steeper than new–new slopes (cf. Campbell, 1987). Failure to find a similarity effect could mean that new–old items were not sufficiently similar to evoke facilitation or interference or that facilitation balanced interference.

The transfer conditions also provided replications of the transfer conditions in Experiment 1. The contrast between old–old and new–new items was the same as the contrast in Experiment 1. The fact that old–old and new–new items were randomly mixed replicates the conditions of Session 15 in Experiment 1 in which old and new items were randomly mixed. This provided us with another opportunity to test the intentionality criterion of automaticity to determine whether old problems can be solved by retrieval even when new problems require counting.

Method

Subjects. The subjects were 27 undergraduate and graduate students at the University of Illinois who served in five 1-hr sessions for $20.

Apparatus and stimuli. The stimuli were displayed on Amdek Model 300 monochrome monitors controlled by IBM PC-XT computers. Letters and digits were 5 × 5 mm on the monitor screen. Equations were arrayed horizontally as in the previous experiments. Each trial began with a fixation display formed of two rows of nine dashes (-), one above and one below the line on which the equation was to appear, centered on the screen. After 500 ms, it was extinguished and replaced by an equation, which remained on until the subject responded. After the response, the screen went blank for a 2.5-s intertrial interval (ITI). Subjects responded by pressing the "slash" keys (/ and \), which were the rightmost and leftmost character keys on the bottom row of the XT keyboard.

Nine stimulus sets were formed by combining three numerical addends (2, 3, and 4) with three sets of six consecutive letters (A–F, G–L, and M–R). In each set, each digit addend appeared once with two letters (e.g., A + 2, B + 3, C+ 4, D + 2, E + 3, F + 4). Over all sets, each digit appeared with each letter equally often. False equations were formed by presenting answers that were true plus one letter (e. g., B + 3 = F).

Procedure. Each session involved 576 trials with brief rests every 72 trials. The number of practice sessions varied with the number of facts to be learned. Six-fact subjects practiced for two sessions, experiencing 96 exposures of the true and the false versions of each of the six facts. A different set of six facts was assigned to each of the 9 subjects in the group. Twelve-fact subjects practiced for four sessions to experience 96 exposures of the true and false versions of their 12 facts. The 12 facts were about different letters. Each subject was assigned two different sets of 6 facts about two different sets of 6 letters (e.g., one set from A–F and one from G–L). The nine sets of 6 facts were paired arbitrarily to produce nine pairs of sets with this characteristic, and a different pair was assigned to each of the 9 subjects in the group. Eighteen-fact subjects practiced for five sessions and experienced 80 exposures to true and false versions of their 18 facts. The 18 facts were about different letters. The nine sets of 6 facts were collapsed into three sets of 18, and each set was assigned to 3 of the 9 subjects in the group.

At transfer, 6-and 12-fact subjects were tested on 18 facts: 6 old facts experienced in training (old–old); 6 new facts about old letters (new–old); and 6 new facts about new letters (new–new). Six-fact subjects were tested for three sessions to complete their series of five; 12-fact subjects were tested for one session. Only the first session of transfer was analyzed.

Subjects were instructed as in Experiment 1.

Results and Discussion

Training. Learning rate can be assessed in two ways: in terms of sessions of practice or trials per item. Typically, learning is assessed as a function of sessions of practice, disregarding how often the items appeared in each session. According to automaticity-as-memory theories, this is inappropriate. The crucial variable is the number of trials per item, which reflects the opportunity to have memorized the items. When the conditions being compared involve the same number of items, the two methods of assessing learning rate are equivalent. However, when different numbers of items are learned, as in the present experiments and in the contrast

between Experiments 1 and 2, trials per item is the more appropriate analysis. To illustrate the importance of this point, we analyzed learning rates both ways. The slope of the linear function relating reaction time to the magnitude of the digit addend was calculated for true and false equations for each subject. The means across subjects in each group are plotted as a function of practice sessions in Figure 5 and as a function of trials per item in Figure 6. Each point in each figure represents the mean of eight presentations of each item. The top panel in each figure contains the slopes for true equations; the bottom panel contains the slopes for false equations.

The plot against sessions in Figure 5 shows orderly differences as a result of the amount to be learned. Six facts approach asymptote before 12 facts; 12 facts approach asymptote before 18. The differences disappear and lose their orderliness in the plot against trials per item in Figure 6. (Differences between conditions in the initial eight presentations presumably reflect random variation.) These impressions were
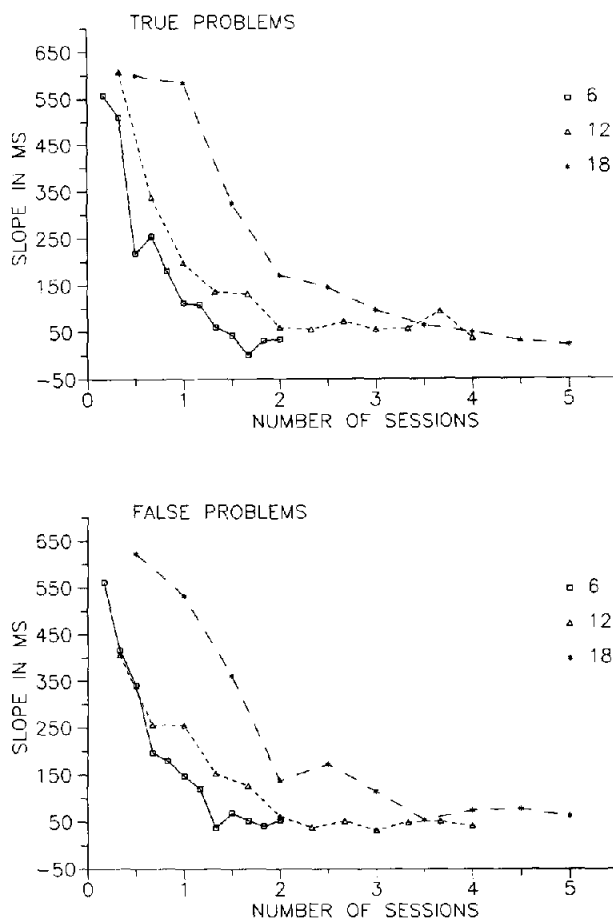

Figure 5. Mean slopes of the linear function relating reaction time to the magnitude of the digit addend in Experiment 3 for subjects trained on 6 (boxes and solid lines), 12 (triangles and short dashed lines), and 18 (stars and long dashed lines) facts as a function of training sessions. (Top panel contains data from true equations; bottom panel contains data from false equations.)
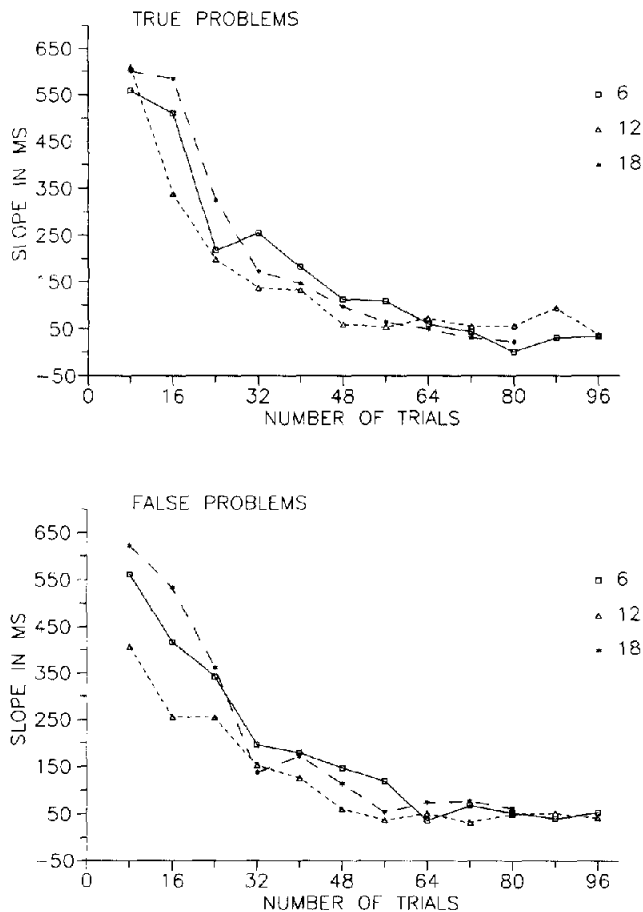
Figure 6. Mean slopes of the linear function relating reaction time to the magnitude of the digit addend in Experiment 3 for subjects trained on 6 (boxes and solid lines), 12 (triangles and short dashed lines), and 18 (stars and long dashed lines) facts as a function of trials per item. (Top panel contains data from true equations; bottom panel contains data from false equations.)

confirmed by a $3 \times 10 \times 2$ (Groups [6, 12, and 18 facts to be learned] $\times$ Practice Blocks [defined as trials per item] $\times$ True vs. false) ANOVA on the slopes. Neither the main effect of groups, $F(2, 24) < 1$, $MS_e = 637,902.39$, nor the interaction between groups and presentations, $F(18, 216) < 1$, $MS_e = 49,659.84$, was significant.

Accuracy was high, averaging 96%. It was not affected much by digit addend (97%, 96%, and 95% for 2, 3, and 4, respectively) or by amount to be learned (97%, 96%, and 95% for 6, 12, and 18 facts, respectively). No interesting trends were apparent, so accuracy was not analyzed further.

The pattern of the reaction time data is consistent with automaticity-as-memory theories, which assert that learning rate is determined primarily by trials per item, not the total amount of practice on the task. The results suggest that the different rates of automatization observed in Experiments 1 and 2 may have been due to the different number of items to learned (40 vs. 6).

Transfer. The slopes from the first transfer session are shown for the 6-fact group and the 12-fact group in Table 1. For both groups, the old–old slopes were shallow and close to the values observed at the end of training, which is consistent with the intentionality criterion of automaticity. Apparently, subjects dealt with old–old problems by retrieving facts from memory even though they had to count to solve new problems.

New–new slopes were substantially steeper than old–old slopes, replicating Experiment 1 in showing a lack of transfer. Lack of transfer is predicted by automaticity-as-memory theories, which assume that subjects learn specific responses to specific items. New–old slopes were very close to new–new slopes, suggesting no facilitation or interference as a result of similarity. These conclusions were confirmed by a $2 \times 3 \times 2$ (Groups [6 vs. 12 fact] $\times$ 3 Condition [old–old vs. old–new vs. new–new] $\times$ True vs. False Equations) ANOVA on the slopes. The main effect of condition was significant, $F(2, 32) = 17.61$, $p < .01$, $MS_e = 95,381.47$, but a contrast comparing old–new and new–new slopes was not, $F(1, 32) < 1$.

The new–old and new–new slopes for the 12-fact subjects were 161 ms smaller than the new–old and new–new slopes for the 6-fact subjects, $F(1, 32) = 9.74$, $p < .01$, $MS_e = 95,381.47$, suggesting some general transfer resulted from the two extra sessions on the task. Because there was no difference between new–old and new–new slopes, the reduction probably reflects improvement in the counting algorithm. Similar improvement was suggested in Experiment 1, but it could not be tested as directly.

Overall accuracy averaged 92.5%. It was higher for old–old problems (97%) than for old–new (91%) and new–new (89%) problems, reflecting trends in the reaction-time data. It was slightly higher for the 12-fact group (93.5%) than for the 6-fact group (91.6%), perhaps reflecting a general practice effect on accuracy regulation (cf. Rabbitt, 1981).

Experiment 4

In Experiment 4, learning by rote memory and learning by performing task were compared directly. All subjects saw alphabet–arithmetic equations followed by their truth value (the word *true* or *false* written under the position the equation

Table 1

Slopes (in ms per Addend) From First Transfer Session of Experiment 3

Truth	Old–old	Old–new	New–new
6 problems			
True	64	377	463
False	91	435	408
M	78	406	436
12 problems			
True	113	324	304
False	36	192	221
M	75	258	263

occupied) after the subject responded. Half of the subjects—the learn-by-doing subjects—were told to verify the equations, pressing different keys to indicate whether the equation was true or false. For learn-by-doing subjects, the truth value provided feedback about the correctness of their response. The other half of the subjects—the learn-by-remembering subjects—were told to memorize the equations, learning which were true and which were false. They examined the equations and pressed the space bar to reveal the truth value. For learn-by-remembering subjects, the truth value was the "response" to be associated with the equation "stimulus" in a classic paired-associate anticipation paradigm. The number of exposures and the nature of the exposures were the same for learn-by-doing and learn-by-remembering subjects. They were then transfered to a common verification task to assess what they had learned and to compare the learning methods.

Experiment 4 manipulated the number of facts to be learned, replicating parts of the first session of Experiment 3. Half of the subjects learned 6 facts and half learned 18 facts. As in Experiment 3, automaticity as memory predicts more learning (flatter slopes) with 6 facts than with 18 because there are more trials per item with 6 facts. The number of facts to be learned was crossed with learning method to test the interaction. Automaticity-as-memory theories predict no interaction. Memory should depend on the number of presentations regardless of how those presentations were made.

Method

Subjects. The subjects were 72 undergraduate and graduate students at the University of Illinois who served in a single 1-hr session for $3.50. They were divided into four groups of 18 subjects.

Apparatus and stimuli. The apparatus and stimuli were the same as in Experiment 3, except that the word *true* or *false* appeared one line below the position occupied by the equation and remained exposed for 500 ms after a response was made.

Procedure. The experiment involved 432 training trials and 144 test trials for a total of 576 trials. Brief rests were allowed every 72 trials. There were four groups of subjects formed from the factorial combination of learning method (learn by doing vs. learn by remembering) and number of facts to be learned (6 vs. 18). The nine sets of 6 facts used in Experiment 3 were used again. Two subjects in each 6-fact group were assigned to each of the nine sets. For the 18-fact groups, sets of 6 facts were combined to produce three different sets of facts about the 18 different letters. Six subjects in each 18-fact group were assigned to each of these sets. Six-fact subjects had 36 exposures to the true and false versions of each fact during training; 18-fact subjects had 12 exposures.

Learning method determined the training task. Learn-by-doing subjects performed the verification task, just as subjects did in Experiments 1 and 3. They were instructed in the same way as subjects in previous verification-task experiments. In addition, they were told that truth values would appear briefly after each equation, which they could use as feedback if they wished. Learn-by-remembering subjects were told to learn each fact and remember whether it was true or false without computing the answer, and to press the space bar when they remembered or when they knew they could not remember. They were told to try to associate the equation with the truth value that appeared after they pressed the space bar.

After 432 training trials, all subjects were switched to a transfer task in which they verified the equations they were trained on. Learn-by-doing subjects simply continued to verify; learn-by-remembering

subjects were given the same instructions about the verification task that learn-by-doing subjects received initially before beginning the task.

Results and Discussion

Training. Learn-by-remembering subjects differed from learn-by-doing subjects in several respects. Their reaction times were generally faster, they were less affected by the truth of the equation, and they were less affected by the magnitude of the digit addend. These differences were tested separately for the 6- and 18-fact groups. In the 6-fact groups, learn-by-remembering reaction times were 986 ms faster than learn-by-doing reaction times in the first training block (i.e., the first four presentations of each item), and the difference diminished to 120 ms by the ninth block. The difference between true and false equations was 54 ms in the learn-by-remembering group and 186 ms in the learn-by-doing group. The slopes of linear functions relating reaction time to the magnitude of the digit addend are plotted as a function of practice block (top panel of Figure 7). Slopes were shallower for learn-by-remembering subjects and were less affected by practice block.

In a 2 × 9 × 3 × 2 (Groups [learn by doing vs. learn by remembering] × Training Block × Digit Addend × True vs. False) ANOVA, the main effect of groups did not reach significance, $F(1, 34) = 2.85, p < .11, MS_e = 15,585,952.59$, but there were significant interactions between groups and training blocks, $F(8, 272) = 2.56, p < .05, MS_e = 1,702,185.90$, and between groups and truth, $F(1, 34) = 10.36, p < .01, MS_e = 204,259.62$. The interaction between groups and addend was marginally significant, $F(2, 68) = 3.00, p < .06, MS_e = 668,179.47$, and the interaction between groups, addend, and block was not significant, $F(16, 544) = 1.07, MS_e = 174,658.88$. However, in a 2 × 9 × 2 (Groups [learn by doing vs. learn by remembering] × Training Block × True vs. False) ANOVA on the slopes, which tested the crucial linear trends, the main effect of groups was significant, $F(1, 34) = 5.97, p < .05, MS_e = 334,471.59$, and the interaction between training blocks and groups was marginally significant, $F(8, 272) = 1.91, p < .06, MS_e = 85,532.74$.

Also significant in the ANOVA on reaction times were the main effects of blocks, $F(8, 272) = 54.72, p < .01, MS_e = 1,702,185.90$, truth, $F(1, 34) = 34.12, p < .01, MS_e = 204,259.62$, and addend, $F(2, 68) = 11.20, p < .01, MS_e = 668,179.47$, and the interaction between block and addend, $F(16, 544) = 3.24, p < .01, MS_e = 174,658.88$.

In the 18-fact groups, learn-by-remembering subjects were 328 ms faster than learn-by-doing subjects. The difference between true and false equations was 150 ms in learn-by-remembering subjects and 286 ms in learn-by-doing subjects. The slopes of linear functions relating reaction time to the magnitude of the digit addend are plotted in the bottom panel of Figure 7. They were generally shallower for learn-by-remembering subjects but changed in the same manner over (four-presentation per item) blocks.

In a 2 × 3 × 3 × 2 (Groups [learn by doing vs. learn by remembering] × Training Blocks × Digit Addend × True vs.
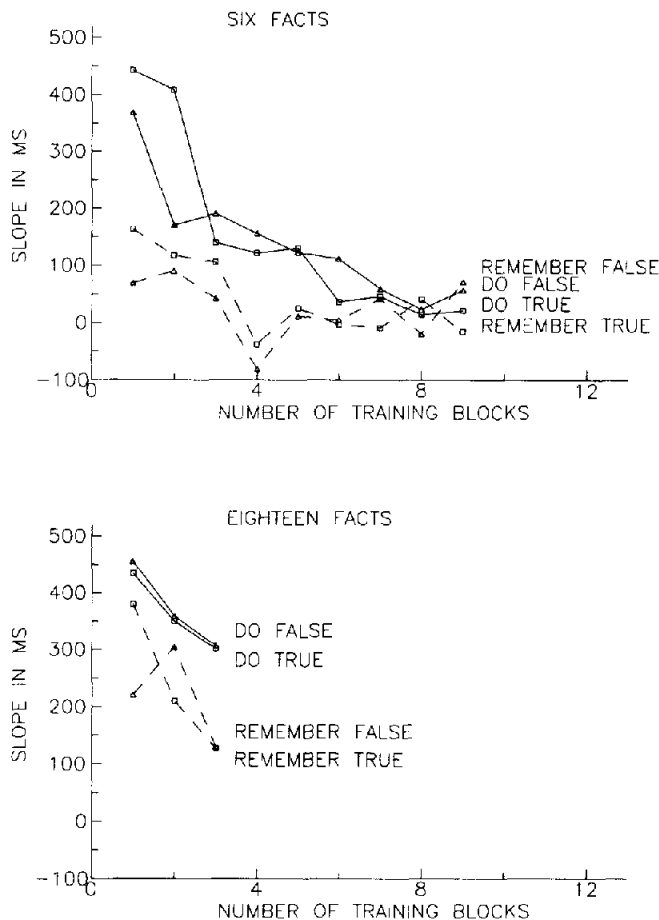
Figure 7. Mean slopes of the linear function relating reaction time to the magnitude of the digit addend in the training blocks of Experiment 4 for subjects trained on 6 (top panel) and 18 (bottom panel) facts under learn-by-doing (solid lines) and learn-by-remembering (broken lines) instructions. (True vs. false equations is the parameter: true = boxes; false = triangles.)

in the initial blocks). It is possible that learn-by-remembering subjects counted occasionally, because their slopes were greater than zero, but they did not count as often as learn-by-doing subjects.

Transfer. In the transfer block, all subjects performed the same verification task. Mean reaction times for true and false equations are plotted for each group as a function of the digit addend in Figure 8. The 6-fact groups are in the top panel and the 18-fact groups are in the bottom panel. As in Experiment 3, the number of facts had a strong effect on the amount of learning manifest in a single session: The functions for the 18-fact groups were higher and steeper than functions for the 6-fact groups. Thirty-six exposures were more effective than 12. The most important result was that learning method had very little effect on performance. Overall reaction times were slightly longer for learning-by-remembering subjects than for learning-by-doing subjects, but the pattern was the same. In

False) ANOVA, the main effect of groups did not reach significance, $F (1, 34) = 2.72, p < .11, MS_e = 6,416,588.03$, nor did the interaction between groups and blocks, $F(2, 68) < 1$, but the interaction between groups and truth was significant, $F(1,34) = 5.11, p < .05, MS_e = 146,305.57$, and the interaction between groups and addend was marginally significant, $F(2, 68) = 2.48, p < .10, MS_e = 415,440.11$. In addition, there were significant main effects of block, $F(2, 68) = 116.75, p < .01, MS_e = 546,600.84$, truth, $F(1, 34) = 52.50, p < .01, MS_e = 146,305.57$, and addend, $F(2, 68) = 48.50, p < .01, MS_e = 415,440.11$, and a significant interaction between block and addend, $F(4, 136) = 3.99, p < .01, MS_e = 114,187.86$.

These data suggest that learn-by-doing subjects performed differently than learn-by-remembering subjects during training. Learn-by-remembering subjects may have been faster, less affected by the truth of the equation, and less affected by the digit addend because they did not use the counting algorithm that learn-by-doing subjects were forced to use (at least
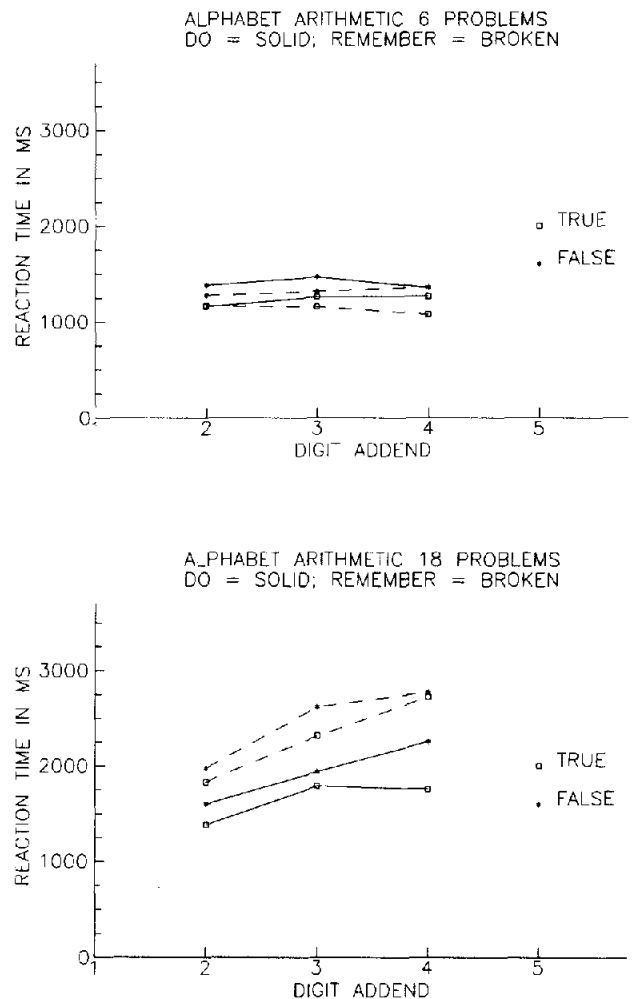

Figure 8. Mean reaction times in Experiment 4 as a function of digit addend for subjects trained on 6 (top panel) and 18 (bottom panel) facts under learn-by-doing (solid lines) and learn-by-remembering (broken lines) instructions. (True vs. false equations is the parameter: true = boxes; false = stars.)

both learning methods, 18-fact groups were slower and steeper than 6-fact groups.

These conclusions were confirmed by a $2 \times 2 \times 3 \times 2$ (Groups [learn by doing vs. learn by remembering] $\times$ 6 vs. 18 facts to be learned $\times$ Addend $\times$ True vs. False Equations) ANOVA on mean reaction times. Learning method had very little effect on performance. Reaction time was significantly longer for the 18-fact groups than for the 6-fact groups, $F(1,68) = 41.21$, $p < .01$, $MS_e = 934,841.07$. The main effect of digit addend was significant, $F(2, 136) = 7.12$, $p < .01$, $MS_e = 89,492.36$, as was the interaction between digit addend and number of facts to be learned, $F(2, 136) = 6.37$, $p < .01$, $MS_e = 89,492.36$. Reaction times were significantly slower for learn-by-remembering than for learn-by-doing subjects, $F(1, 68) = 5.26$, $p < .05$, $MS_e = 934,841.07$, but more important, there were no significant interactions involving learning method.

Accuracy averaged 93.4% overall. The 6-fact groups were slightly more accurate than the 18-fact groups (94.5% vs. 92.4%). The learn-by-doing groups were about as accurate as the learn-by-remembering groups (94.3% vs. 93.6%).

The similar effects of learning method suggest that how the facts got into memory was not very important; what mattered was the availability of the facts in memory. Number of facts to be learned or, equivalently, the number of presentations of each fact affected availability in the same way for both learning methods. The results suggest that the different rates of automatization observed in Experiments 1 and 2 were not due to differences in learning method. Although learning by remembering produced automaticity in one session in Experiment 2 and learning by doing produced automaticity in 12 sessions in Experiment 1, the difference was most likely due to differences in the number of facts to be learned (6 vs. 40).

General Discussion

Experiment 1 demonstrated the automatization of alphabet arithmetic in several ways: Reaction time sped up as a power function of practice (as reported in Logan, 1988b), the slope of the function relating reaction time to the magnitude of the digit addend decreased by an order of magnitude, and subjects reported large reductions in the difficulty of the task (remembering was much easier than counting). Several pieces of evidence suggested that a transition from counting to remembering underlaid the automatization: Standard deviations were fit by power functions with the same exponent as the reaction times (as reported in Logan, 1988b), confirming a prediction of the instance theory (also see Cohen et al., 1990). The asymptotic slopes were twice as fast as overt recitation rates, suggesting that performance no longer depended on sequential access to the alphabet. Subjects reported counting less often and remembering more often as practice progressed. Furthermore, transfer to new items was poor, suggesting that subjects memorized only the items they experienced during training.

Characteristically, automaticity took a long time to develop in Experiment 1: It took 12 sessions of practice on the task. The remaining experiments showed that neither extended practice nor practice on the criterion task were necessary for

automaticity to develop, testing predictions derived from automaticity-as-memory theories. Experiment 2 produced automaticity (a flat slope) in a 1-hr session by having subjects rote memorize the facts. Experiments 3 and 4 tested two procedural differences between Experiments 1 and 2. Experiment 3 compared learning rates with different numbers of facts (Experiment 1 required 40 facts; Experiment 2 required 6). It showed the rate of automatization was the same for 6, 12, and 18 facts when assessed per presentation of each item. Because 6 facts can be presented more often than 18 facts in a single session, there was a greater change in performance over sessions the smaller the number of facts to be learned, which accounts for the difference between Experiments 1 and 2. Experiment 4 compared learning by doing (the procedure in Experiment 1) and learning by remembering (the procedure in Experiment 2) directly, and showed no important differences in subsequent performance.

Property Lists, Resources, and Automaticity as Memory

The experimental results were predicted by theories that assume that memory is the process underlying automaticity. According to those theories, performance is automatic when it is based on single-step, direct-access retrieval of a solution from memory; in principle, that can occur after a single exposure. The theories also imply that having traces available in memory is necessary for automatic performance. It is not necessary that the traces were put into memory in the course of performing the task. Other routes to memory, such as rote rehearsal, may be sufficient.

The results do not rest easily with property-list approaches. Extended practice is characteristic of automaticity and figures prominently on property lists (e.g., Hasher & Zacks, 1979; Schneider, Dumais, & Shiffrin, 1984), and practice on the criterion task is implicitly important. Property-list approaches provide no learning mechanism and so cannot predict anything about the conditions that produce automaticity. They describe surface features, not underlying processes, and so they are limited in explanatory power.

Property-list approaches have not done well in other investigations of automaticity. Several studies have shown that the properties of automaticity do not cooccur as property-list theories imply they should. For example, Regan (1981), Paap and Ogden (1981), and Kahneman and Chajzyck (1983) showed that obligatory processes can be effortful, whereas property-list theories imply that obligatory processes should be effortless (see, e.g., Hasher & Zacks, 1979). There appear to be no properties that are individually necessary and jointly sufficient to define automaticity, which has led some investigators to challenge the utility of the concept (Cheng, 1985; Ryan, 1983) and others to propose intermediate states of automaticity that possess some properties but not others (Kahneman & Treisman, 1984; Zbrodoff & Logan, 1986).

The problem may lie with the method of defining automaticity and not with the concept of automaticity itself. In general, property-list approaches offer little conceptual coherence and suggest only limited avenues of research. Properties

often appear on the list by stipulation rather than by deduction. The theories do not explain why automaticity should have the properties it has or why the properties should go together. Thus, they offer no explanation of exceptions to co-occurrence, when they are observed. By contrast, theories that define automaticity in terms of underlying processes allow us to deduce the properties of automaticity and explain why certain properties should appear together and others should not. In principle, at least, they offer ways to explain exceptional cases. Murphy and Medin (1985) made this point more generally in discussing how categories should be defined: Theories of underlying processes provide conceptual coherence and define the boundaries of a concept better than lists of manifest properties. Their logic applies to scientific concepts, like automaticity, as well as the lay concepts with which they were primarily concerned.

Approaches to automaticity based on resource theory (e.g., Logan, 1978; Posner & Snyder, 1975) fare somewhat better than property-list approaches. They provide a mechanism from which the properties of automaticity can be deduced. However, they do not specify a learning mechanism and so cannot account for the present results except in a post hoc manner.

Resource-theory approaches do not fare well in other contexts either. The explanations they provide are clearest when only a single resource is assumed (i.e., central processing capacity; Kahneman, 1973). However, single-resource theories have been replaced by multiple-resource theories (e.g., Navon & Gopher, 1979; Wickens, 1984), which make prediction difficult (see Logan, 1985, 1988a). Moreover, there are strong arguments against the utility of resource theory; prominent theorists claim there are no resource limitations (e.g., Allport, 1980; Navon, 1984; Navon & Miller, 1987). These challenges to resource theory threaten to remove the core concept that gives coherence to resource-based theories of automaticity. Consequently, automaticity-as-memory theories are becoming more attractive.

Learning by Doing Versus Learning by Remembering

The equivalence of learning by doing and learning by remembering might have been anticipated from the arithmetic literature. Children may learn addition facts as a side effect of executing a counting algorithm, but they learn multiplication mostly by rote memorization. Teachers may explain multiplication as iterative addition to help students grasp the concept, but they do not expect students to use that algorithm to multiply. Flash cards and speeded tests are designed to promote memorization. Interestingly, the associative structures that underlie multiplication are similar to those that underlie addition (e.g., Ashcraft, 1987; Campbell, 1987; Siegler, 1987, 1988). Moreover, addition and multiplication interfere with one another (e.g., 3 + 4 = 12 and 3 × 4 = 7 are hard to reject because they are true for the opposite operation; see Winkelman & Schmidt, 1974; Zbrodoff & Logan, 1986), which suggests they are part of the same associative structure. Apparently, the different learning methods lead to the same memory representation.

We do not mean to imply that learning by doing and learning by remembering are exactly equivalent. Certainly, different mnemonic strategies could make learning by remembering more or less effective than learning by doing, and learning by doing could be more or less effective depending on task parameters such as the time between trials. The important point is that both methods are sensitive to the number of presentations of individual items, as we saw in Experiment 4. With enough practice, one method should be able to reproduce the degree of automaticity attained by the other.

It is tempting to relate the present results to the distinction between explicit and implicit memory. Explicit memory is typically observed in recognition and recall tasks, and it is thought to involve an awareness by the subject that memory is being used. Implicit memory is typically observed as an improvement in performance that results from prior experience with the stimulus, and it is thought to involve no necessary awareness that memory is being used (for reviews, see Jacoby & Brooks, 1984; Schacter, 1987). Learning by remembering seems more likely to involve explicit memory than implicit memory because subjects consciously memorize facts for later use. Learning by doing seems more likely to involve implicit memory because performance improves from having performed the task before. The equivalence of learning by remembering and learning by doing is interesting because it suggests an equivalence of explicit and implicit memory. Much of the literature addresses differences between them; we show commonalities.

There are at least three ways in which implicit and explicit memory could be used in our tasks. First, there may be only one memory system that is affected in the same way by the two learning methods. Second, there may be separate memory systems, one for each learning method, that happen to respond similarly to our experimental manipulations (varying the number of trials per item and the number of facts to be learned). Third, there may be separate memory systems, but each learning method lays down traces in both systems. In each case, learning by remembering would be equivalent to learning by doing, but for vastly different reasons. The present experiments provide no way to distinguish among the cases, so we can draw no firm conclusions however tempting it might be to do so.

Single-Step, Direct-Access Retrieval

Automaticity-as-memory theories assume that the process underlying automatic performance is single-step, direct-access retrieval. The data seem to provide clear evidence that a retrieval process is involved in automatized alphabet arithmetic, but they say nothing, about the number of steps involved or the nature of the access to retrieved representations. Possibly more than one step is involved in the retrieval process, and possibly the representations are not accessed directly. These are questions for future research.

It seems likely that the retrieval process involves direct access. Most of the modern, global theories of memory, such as Ratcliff's (1978), Murdock's (1982, 1983), Hintzman's (1986, 1988), Shiffrin's (Gillund & Shiffrin, 1984; Raaijmak-

ers & Shiffrin, 1981), and Humphreys, Bain, and Pike's (1989), assume direct access. At present, there seems to be no viable theoretical alternative to direct access. Does the retrieval process involve a single step? The issue is important because some acts of retrieval, such as recalling the name of one's first-grade teacher or one's last phone number seem to be the antithesis of automaticity: They are difficult, time consuming, and often unsuccessful.

It is important to distinguish between retrieval tasks, such as recognition and recall, and the retrieval process that underlies them. Retrieval tasks may be automatic or effortful depending on what the retrieval process provides. The instance theory of automaticity assumes that the retrieval process is obligatory, operating continuously, retrieving whatever is associated with the cues that are presented to it. Sometimes the cues are very effective and associates spring readily to mind. This easy, effortless retrieval is the essence of automaticity according to instance theory. Other times the cues will be ineffective and nothing will come out of the retrieval process, as when we try to recall former teachers' names or old phone numbers. New cues must be formulated and fed back into the retrieval process until something comes out or we give up. The process of formulating new cues may be the effortful step in difficult retrieval tasks; the retrieval process itself may be effortless.

The single-step nature of the retrieval underlying automaticity may be defined in terms of processing cycles in which cues are presented to the retrieval process and associates are retrieved (e.g., Gillund & Shiffrin, 1984; Siegler, 1988). When many traces are associated with a given cue, at least one of them is likely to come out in a single cycle. This is likely the case in well-practiced alphabet arithmetic. What happens if nothing comes out in response to the cues? It depends on the retrieval task. In a typical automaticity experiment, subjects are required to solve a problem. Most likely, they will execute a problem-solving algorithm at the same time as they attempt retrieval, and they may rely on the solution the algorithm produces if retrieval fails without attempting further retrieval. In recall or recognition experiments, subjects are required to retrieve something from memory. If retrieval fails, they must formulate new cues and put them back into the retrieval process, iteratively, until something comes out. After a number of unsuccessful cycles—after several steps—they may give up.

Given this conception, we argue that the automaticity observed in the present experiments is based on single-step, direct-access retrieval. Although our experiments cannot distinguish single-step from multiple-step retrieval, the difference should be clear conceptually. Perhaps future research will provide empirical distinctions between them.

Varieties of Automaticity

Automaticity is a broad concept that covers many different domains. For example, "pop-out" effects in visual search, priming effects in lexical decision tasks, Stroop effects, sequential effects in serial reaction time tasks, and perceptual–motor skills have all been attributed to automaticity. Can one theory of automaticity cover all of these domains? Does the retrieval process that operates in automatic alphabet arithmetic underlie these different examples of automaticity?

On the one hand, it seems unlikely that automatic alphabet arithmetic characterizes the variety of paradigms in which automaticity appears. Alphabet arithmetic involves fact retrieval from long-term memory (as evidenced by month-long retention seen in Experiment 1), possibly from declarative memory (as suggested by the equivalence of learning by doing and learning by remembering). It should be easy to model Stroop and priming effects in a similar manner. However, pop-out effects seem to depend on the perceptual discriminability of targets and distractors (Duncan & Humphreys, 1989; Treisman & Gormican, 1988), and sequential effects in serial reaction time are short lived, presumably depending on short-term memory (Soetens, Deboeck, & Hueting, 1984). It seems unlikely that the same memory system underlies these different phenomena.

On the other hand, it may be possible to model automaticity in each of these paradigms as retrieval from memory. The paradigms may differ in the memory systems they tap, but they all may rely on retrieval of one sort or another. Some kinds of perceptual discriminations depend on early experience (e.g., Weisel & Hubel, 1963), and one can view the effects of that experience as residing in a memory system. Thus, pop-out effects may be due to rapid retrieval in long-term perceptual memory. Similarly, sequential effects may depend on short-term memory. Short-term memory traces of past responses may race with a response selection process that applies the instructed mapping rules to govern performance (e.g., Fletcher & Rabbitt, 1978; Krueger & Shapiro, 1981). In addition, perceptual–motor skills may depend on retrieval from motor memory.

Automaticity as memory may be viewed as a general theory that can be instantiated in several different ways. There may be different varieties of automaticity associated with different memory systems. The substrate on which they operate may be different, but they all may be characterized as relying on single-step, direct-access memory retrieval. Whether different retrieval theories can account for the varieties of automaticity remains an open question; the answer awaits future research and serious theoretical development. Moreover, if different retrieval theories succeed in the different domains, it is not clear that there will be important commonalities among them at a more abstract, general level of analysis. Nevertheless, the possibility of a general, all-encompassing theory is enticing.

Automaticity and Skill

The conclusion that automatization depends on the number of presentations of individual items rather than the total amount of practice on the task has interesting implications. It suggests that automaticity can be attained very quickly if there is not much to be learned. Even if there is much to be learned, parts of it can be automatized quickly if they are trained in isolation. Subjects in Experiment 2 learned six facts to a criterion of automaticity in about 15 min. This conclusion has important implications for future studies of automaticity. It may no longer be necessary to train subjects for several sessions to produce automaticity; a single session may suffice

(e.g., Logan, 1990; also see Klapp et al., 1991). Automaticity experiments can be less costly and less cumbersome to conduct. Experiments can be completed quickly, and progress in understanding automaticity can proceed more rapidly.

The conclusion that automaticity depends on the number of presentations and not the number of practice sessions raises an interesting problem: How do we compare the automaticity of subjects trained for 15 min in Experiment 2 with that of subjects in Experiment 1 who were trained on 40 facts for 12 h before attaining automaticity? By the zero-slope criterion, performance was automatic in both experiments, yet subjects in Experiment 1 had automatized more alphabet-arithmetic facts than subjects in Experiment 2.

We suggest drawing a distinction between automaticity and skill, in which automaticity refers to the memorability of individual facts in a domain and skill refers to the proportion of the domain that is automatized. Thus, individual facts can vary in their degree of automaticity, and people who know more facts (who have more facts automatized) are more skilled. Subjects in Experiment 1 were more skilled alphabet arithmeticians than subjects in Experiment 2, although their performance was no more automatic.

This distinction should apply broadly to domains beyond alphabet arithmetic. For example, a student who knew multiples of 3 to a criterion of automaticity would be less skilled at multiplication than one who knew the products of all combinations of single digits to a criterion of automaticity. Another student could have skill without a high degree of automaticity, knowing most of the multiplication facts but not knowing them well. In principle, the distinction can apply to less well-defined domains than arithmetic, in which it is harder to assess complete mastery of the domain. One person could still know more of the domain than another, and thus be more skilled; one could know parts of the domain better than another, and thus be more automatic. Essentially, the contrast is between breadth and depth of knowledge.

This distinction raises an important practical question: It may be possible to produce automaticity in a single session, but what about skill? It probably depends on the domain. In domains like arithmetic, the facts to be learned are largely independent, and skill consists primarily in mastering each separate fact. In that case, single-session automaticity can portray the essence of skill. However, the facts may be more interactive in other domains. There may be complex hierarchical relations between them that can be appreciated only when a substantial proportion of the domain is mastered. In these cases, there may be no substitute for extended practice. Single facts may be committed to memory in a single session, but it may take years before the relationships between those facts and the rest of the domain are discovered and committed to memory.

References

Allport, D. A. (1980). Attention and performance. In G. Claxton (Ed.), Cognitive psychology (pp. 112–153). London: Routledge and Kegan Paul.

Ashcraft, M. H. (1982). The development of mental arithmetic: A chronometric approach. Developmental Review, 2, 213–236.

Ashcraft, M. H. (1987). Children's knowledge of simple arithmetic: A developmental model and simulation. In C. J. Brainerd, R. Kail, & J. Bisanz (Eds.), Formal models in developmental psychology (pp. 302–338). New York: Springer Verlag.

Campbell, J. I. D. (1987). Network interference and mental multiplication. Journal of Experimental Psychology: Learning, Memory, and Cognition, 13, 109–123.

Cheng, P. W. (1985). Restructuring versus automaticity: Alternative accounts of skill acquisition. Psychological Review, 92, 414–423.

Cohen, J. D., Dunbar, K, & McClelland, J. L. (1990). On the control of automatic processes: A parallel distributed processing account of the Stroop effect. Psychological Review, 97, 332–361.

Compton, B. J., & Logan, G. D. (in press). The transition from algorithm to retrieval in memory based theories of automaticity. Memory & Cognition.

Duncan, J., & Humphreys, G. W. (1989). Visual search and stimulus similarity. Psychological Review, 96, 433–458.

Feustal, T. C., Shiffrin, R. M., & Salasoo, A. (1983). Episodic and lexical contributions to the repetition effect in word identification. Journal of Experimental Psychology: General, 112, 309–346.

Fletcher, B., & Rabbitt, P. M. A. (1978). The changing pattern of perceptual analytic strategies and response selection with practice in a two-choice reaction time task. Quarterly Journal of Experimental Psychology, 30, 417–427.

Gillund, G., & Shiffrin, R. M. (1984). A retrieval model for both recognition and recall. Psychological Review, 91, 1–67.

Groen, G. J., & Parkman, J. (1972). A chronometric analysis of simple addition. Psychological Review, 79, 329–343.

Groen, G. J., & Resnick, L. B. (1977). Can preschool children invent addition algorithms? Journal of Educational Psychology, 69, 645–652.

Hasher, L., & Zacks, R. T. (1979). Automatic and effortful processes in memory. Journal of Experimental Psychology: General, 108, 356–388.

Healy, A. F., Fendrich, D. W., & Proctor, J. D. (1990). Acquisition and retention of a letter-detection skill. Journal of Experimental Psychology: Learning, Memory, and Cognition, 16, 270–281.

Hintzman, D. L. (1986). "Schema abstraction" in a multiple-trace model. Psychological Review, 93, 411–428.

Hintzman, D. L. (1988). Judgments of frequency and recognition memory in a multiple-trace memory model. Psychological Review, 95, 528–551.

Humphreys, M. S., Bain, J. D., & Pike, R. (1989). Different ways to cue a coherent memory system: A theory for episodic, semantic, and procedural tasks. Psychological Review, 96, 208–233.

Jacoby, L. L., & Brooks, L. R. (1984). Nonanalytic cognition: Memory, perception, and concept learning. In G. H. Bower (Ed.), The psychology of learning and motivation (pp. 1–47). New York: Academic Press.

Kahneman, D. (1973). Attention and effort. Englewood Cliffs, NJ: Prentice-Hall.

Kahneman, D., & Chajzyck, D. (1983). Tests of the automaticity of reading: Dilution of Stroop effects by color-irrelevant stimuli. Journal of Experimental Psychology: Human Perception and Performance, 9, 497–509.

Kahneman, D., & Treisman, A. M. (1984). Changing views of attention and automaticity. In R. Parasuraman & R. Davies (Eds.), Varieties of attention (pp. 29–61). San Diego, CA: Academic Press.

Kail, R. (1986). Sources of age differences in speed of processing. Child Development, 57, 969–987.

Kail, R. (1988). Developmental functions for speeds of cognitive processes. Journal of Experimental Child Psychology, 45, 339–364.

Klapp, S. T., Boches, C. A., Trabert, M. L., & Logan, G. D. (1991). Automatizing alphabet arithmetic: II. Are there practice effects after automaticity? Journal of Experimental Psychology: Learning,

Memory, and Cognition, 17, 194–207.

Kolers, P. A. (1976). Reading a year later. *Journal of Experimental Psychology: Human Learning and Memory, 2,* 554–565.

Krueger, L. E., & Shapiro, R. G. (1981). Intertrial effects of same-different judgments. *Quarterly Journal of Experimental Psychology, 33A,* 241–265.

LaBerge, D., & Samuels, S. J. (1974). Toward a theory of automatic information processing in reading. *Cognitive Psychology, 6,* 293–323.

Landauer, T. K. (1962). Rate of implicit speech. *Perceptual and Motor Skills, 15,* 646.

LeFevre, J., Bisanz, J., & Mrkonjic, L. (1988). Cognitive arithmetic: Evidence for obligatory activation of arithmetic facts. *Memory & Cognition, 16,* 45–53.

Logan, G. D. (1978). Attention in character classification: Evidence for the automaticity of component stages. *Journal of Experimental Psychology: General, 107,* 32–63.

Logan, G. D. (1985). Skill and automaticity: Relations, implications, and future directions. *Canadian Journal of Psychology, 39,* 367–386.

Logan, G. D. (1988a). Automaticity, resources, and memory: Theoretical controversies and practical implications. *Human Factors, 30,* 583–598.

Logan, G. D. (1988b). Toward an instance theory of automatization. *Psychological Review, 95,* 492–527.

Logan, G. D. (1990). Repetition priming and automaticity: Common underlying mechanisms? *Cognitive Psychology, 22,* 1–35.

Murdock, B. B., Jr. (1982). A theory for the storage and retrieval of item and associative information. *Psychological Review, 89,* 609–626.

Murdock, B. B., Jr. (1983). A distributed memory model for serial-order information. *Psychological Review, 90,* 316–338.

Murphy, G. L., & Medin, D. L. (1985). The role of theories in conceptual coherence. *Psychological Review, 92,* 289–316.

Navon, D. (1984). Resources—A theoretical soup stone? *Psychological Review, 91,* 216–234.

Navon, D., & Gopher, D. (1979). On the economy of the human processing system. *Psychological Review, 86,* 214–255.

Navon, D., & Miller, J. (1987). Role of outcome conflict in dual-task interference. *Journal of Experimental Psychology: Human Perception and Performance, 13,* 435–448.

Neely, J. H. (1977). Semantic priming and retrieval from lexical memory: Roles of inhibitionless spreading activation and limited-capacity attention. *Journal of Experimental Psychology: General, 106,* 226–254.

Newell, A., & Rosenbloom, P. S. (1981). Mechanisms of skill acquisition and the law of practice. In J. R. Anderson (Ed.), *Cognitive skills and their acquisition* (pp. 1–55). Hillsdale, NJ: Erlbaum.

Paap, K. R., & Ogden, W. C. (1981). Letter encoding is an obligatory but capacity-demanding operation. *Journal of Experimental Psychology: Human Perception and Performance, 7,* 518–527.

Posner, M. I., & Snyder, C. R. R. (1975). Attention and cognitive control. In R. L. Solso (Ed.), *Information processing and cognition: The Loyola symposium* (pp. 55–85). Hillsdale, NJ: Erlbaum.

Raaijmakers, J. G., & Shiffrin, R. M. (1981). Search of associative memory. *Psychological Review, 88,* 93–134.

Rabbitt, P. M. A. (1981). Sequential reactions. In D. H. Holding (Ed.), *Human skills* (pp. 153–175). New York: Wiley.

Ratcliff, R. (1978). A theory of memory retrieval. *Psychological Review, 85,* 59–108.

Regan, J. E. (1981). Automaticity and learning: Effects of familiarity on naming letters. *Journal of Experimental Psychology: Human Perception and Performance, 7,* 180–195.

Rosenbloom, P. S., & Newell, A. (1986). The chunking of goal hierarchies: A generalized model of practice. In R. S. Michalski, J. G. Carbonell, & T. M. Mitchell (Eds.), *Machine learning: An artificial intelligence approach* (Vol. 2, pp. 247–288). Los Altos, CA: Morgan Kaufmann.

Ryan, C. (1983). Reassessing the automaticity-control distinction: Item recognition as a paradigm case. *Psychological Review, 90,* 171–178.

Schacter, D. L. (1987). Implicit memory: History and current status. *Journal of Experimental Psychology: Learning Memory, and Cognition, 13,* 501–522.

Schneider, W. (1985). Toward a model of attention and the development of automatic processing. In M. I. Posner & O. S. Marin (Eds.), *Attention and performance XI* (pp. 475–492). Hillsdale, NJ: Erlbaum.

Schneider, W., & Detweiler, M. (1987). A connectionist/control architecture for working memory. In G. H. Bower (Ed.), *The psychology of learning and motivation, Vol. 21,* (pp. 53–119). San Diego, CA: Academic Press.

Schneider, W., Dumais, S. T., & Shiffrin, R. M. (1984). Automatic and control processing and attention. In R. Parasuraman & R. Davies (Eds.), *Varieties of attention* (pp. 1–27). San Diego, CA: Academic Press.

Shiffrin, R. M., & Schneider, W. (1977). Controlled and automatic human information processing: II. Perceptual learning, automatic attending, and a general theory. *Psychological Review, 84,* 127–190.

Siegler, R. S. (1987). The perils of averaging data over strategies: An example from children's addition. *Journal of Experimental Psychology: General, 116,* 250–264.

Siegler, R. S. (1988). Strategy choice procedures and the development of multiplication skill. *Journal of Experimental Psychology: General, 117,* 258–275.

Soetens, E., Deboeck, M., & Hueting, J. (1984). Automatic aftereffects in two-choice reaction time: A mathematical representation of some concepts. *Journal of Experimental Psychology: Human Perception and Performance, 10,* 581–598.

Treisman, A., & Gormican, S. (1988). Feature analysis in early vision: Evidence from search asymmetries. *Psychological Review, 95,* 15–48.

Weisel, T. N., & Hubel, D. H. (1963). Single cell responses in striate cortex of kittens deprived of vision in one eye. *Journal of Neurophysiology, 26,* 1003–1017.

Wickens, C. D. (1984). Processing resources in attention. In R. Parasuraman & R. Davies (Eds.), *Varieties of attention* (pp. 63–102). San Diego, CA: Academic Press.

Winkelman, J. H., & Schmidt, J. (1974). Associative confusions in mental arithmetic. *Journal of Experimental Psychology, 102,* 734–736.

Zbrodoff, N. J., & Logan, G. D. (1986). On the autonomy of mental processes: A case study of arithmetic. *Journal of Experimental Psychology: General, 115,* 118–130.