Per-Fact Graduated Consolidation Resolves the Capacity Ceiling in Weight-Edited Language Models

Vladimir Baranov
vlad@chatsetter.ai

Abstract

Language models that learn from conversation via direct weight editing face a hard capacity ceiling: the 8B Llama model sustains reliable recall for only ∼13 unconstrained MEMIT edits before cascading interference collapses performance, and sleep maintenance alone cannot grow capacity beyond the null-space-constrained limit. Prior attempts to offload knowledge into LoRA adapters failed entirely: LoRA fine-tuning incurs an alignment tax (37% recall degradation on 8B) that blocks the transfer pathway, and per-edit gating—where all facts in a batch must pass before any can advance—produced 0% advancement with recall degrading from 0.80 to 0.60. We resolve both failures with per-fact graduated consolidation. Each fact independently tracks its own consolidation stage and advances or retreats based on individual chat recall after LoRA absorption. A graduated dissolution schedule (scale $1.0 \to 0.5 \to 0.1 \to 0.0$) progressively reduces MEMIT influence as LoRA takes over, and cumulative fusing—training each cycle on an already-fused model—overcomes the alignment tax through incremental prior erosion. In a capacity sweep on Llama 3.1 8B (4-bit, 2×H100) with {5, 10, 15, 20} facts across 3 sleep cycles, every condition achieves 100% advancement rate and 1.00 chat recall. MEMIT edits dissolve as designed, making the buffer renewable: effective lifetime capacity becomes unbounded, limited only by the number of sleep cycles the user is willing to run.

1 Introduction

Papers 1 through 5 of this series trace an arc of discovery. Paper 1 (Baranov, 2026d) introduced sleep-wake consolidation using LoRA fine-tuning to transfer conversational memories into model weights. Paper 2 (Baranov, 2026a) discovered the alignment tax: LoRA training on raw fact completions degrades chat-template recall by 37% on 8B models, because the instruction-tuned distribution shifts during fine-tuning. Paper 3 (Baranov, 2026c) combined MEMIT with LoRA in a dual-system architecture. Paper 4 (Baranov, 2026e) introduced two-phase sleep with SWS and REM stages. Paper 5 (Baranov, 2026b) removed LoRA entirely, establishing MEMIT as the sole memory mechanism with a sleep maintenance cycle of auditing, null-space-constrained refreshing, and pruning.

Paper 5's MEMIT-only architecture achieved remarkable results—100% recall recovery from 40% degradation within 4 sleep cycles—but exposed a hard ceiling. The 8B model sustains ≥0.90 recall at 13 unconstrained edits, then collapses to 0.57 at fact 14. Sleep maintenance can repair this damage but cannot grow capacity beyond the null-space-constrained limit (∼30 facts for 8 layers). Worse, when total edits exceed a pruning threshold, a death spiral drives recall from 97% to 46% over 10 cycles. The system needs an overflow mechanism: a way to move established facts out of MEMIT's finite buffer into higher-capacity storage.

The obvious candidate is LoRA—but Paper 2 showed it destroys chat recall, and an initial attempt at per-edit gating produced 0% advancement with recall actually degrading. The key insight is that the alignment tax is not absolute but per-pass: each LoRA fine-tuning pass shifts the distribution slightly, and cumulative fusing on an already-shifted model requires progressively less adjustment. Furthermore, gating must operate at the fact level, not the edit level: one stubborn fact should not block the entire batch.

This paper makes five contributions:

1. **Per-fact gating.** Each fact in a MEMIT edit independently tracks its consolidation stage and advances or retreats based on individual chat recall after LoRA absorption, replacing all-or-nothing edit-level gating.

2. **Graduated dissolution schedule.** A four-stage scale schedule $(1.0, 0.5, 0.1, 0.0)$ progressively reduces MEMIT influence as LoRA demonstrates it can carry each fact, with edit-level scale determined conservatively by the minimum fact stage.

3. **Cumulative fusing overcomes the alignment tax.** Training on an already-fused model produces dramatically lower starting loss (0.62 vs. 2.91), allowing stubborn facts to transfer on subsequent cycles without catastrophic distribution shift.

4. **100% advancement at all tested scales.** In a sweep of $\{5, 10, 15, 20\}$ facts $\times$ 3 cycles, every fact eventually leaves stage 0. Chat recall reaches 1.00 for all conditions.

5. **Unbounded effective capacity.** Because consolidation progressively clears MEMIT edits (scale $\rightarrow$ 0.0 at stage 3), the buffer is renewable. Lifetime capacity is limited only by LoRA's absorption rate, not by MEMIT's interference ceiling.

This paper supersedes our prior work on sleep-wake consolidation (Baranov, 2026d), the alignment tax (Baranov, 2026a), dual-system memory (Baranov, 2026c), two-phase consolidation (Baranov, 2026e), and MEMIT-only convergence (Baranov, 2026b).

2 Background and Motivation

2.1 The Alignment Tax

Paper 2 (Baranov, 2026a) characterized the alignment tax: LoRA fine-tuning on raw completion text (e.g., "Aria Nakamura lives in Portland") shifts the model's distribution away from its instruction-tuned behavior. On the 8B model, a single LoRA training pass degrades chat-template recall by 37%. The effect scales inversely with model size—47% for 3B, 37% for 8B, 0% for 70B—suggesting that larger models have more capacity to absorb distributional perturbations without losing their instruction-following behavior.

The alignment tax made LoRA consolidation appear unviable for sub-70B models. Paper 5 responded by removing LoRA entirely. But this left the system with MEMIT as its only storage, subject to a hard interference ceiling.

2.2 The Capacity Ceiling

Paper 5 (Baranov, 2026b) established the wake capacity threshold at $\sim$13 unconstrained edits for 8B with 8 MEMIT layers. Beyond this point, cascading interference causes recall to collapse. Sleep maintenance with null-space-constrained refreshes can repair the damage and sustain $\sim$30 facts, but

this is the ceiling: the null-space becomes increasingly crowded, and the pruning heuristic begins removing working edits faster than refresh can replace them.

The fundamental problem is that MEMIT edits are additive perturbations to MLP weights. Each edit consumes a slice of the available weight-update subspace. With 8 target layers and a 4096-dimensional hidden space, the capacity is finite. The system needs a mechanism to retire old MEMIT edits once their knowledge is safely stored elsewhere.

2.3 The Per-Edit Gating Failure

The first attempt to reintroduce LoRA used per-edit gating: after LoRA training and fusing, the system tested chat recall for each MEMIT edit as a whole. If recall passed, the edit's consolidation stage advanced; if it failed, the stage retreated. This gating was all-or-nothing at the edit level.

The results were catastrophic: 0% advancement rate, with recall degrading from 0.80 to 0.60 at 10 facts. The problem was twofold. First, the alignment tax from a single LoRA pass was severe enough that most facts failed the chat recall test, so no edits advanced. Second, the all-or-nothing gating meant that even if 4 of 5 facts in an edit transferred successfully, one stubborn fact blocked the entire batch from advancing. The LoRA training destabilized recall without producing any consolidation benefit.

3 Method

3.1 Per-Fact Gating

We replace the scalar `consolidation_stage` field on each MEMIT edit with a vector `fact_stages: List[int]`, one integer per fact. Each fact independently tracks its position in the dissolution schedule. After LoRA fusing, the system temporarily zeros the MEMIT edit's delta and tests chat recall for each fact individually. Facts that pass advance one stage; facts that fail retreat to stage 0. The backward-compatible `consolidation_stage` property returns min(`fact_stages`), which determines the edit-level MEMIT scale. This conservative rule ensures that MEMIT influence is only reduced when *all* facts in an edit have been absorbed by LoRA.

3.2 Graduated Dissolution Schedule

The scale schedule maps consolidation stages to MEMIT delta multipliers (Table 1).

Table 1: Graduated dissolution schedule. MEMIT influence is withdrawn progressively as LoRA demonstrates it can carry each fact. At stage 3 the delta is fully zeroed and the edit can be pruned.

Stage	MEMIT Scale	Interpretation
0	1.0	MEMIT carries all weight
1	0.5	LoRA partially absorbing
2	0.1	LoRA nearly complete
3	0.0	LoRA carries all weight; MEMIT retired

At stage 3, the MEMIT delta is fully zeroed and the edit can be pruned from the active set, freeing capacity for new facts. The graduation ensures that MEMIT support is withdrawn gradually: if LoRA's absorption is incomplete, the residual MEMIT signal provides a safety net.

3.3 Cumulative Fusing

Each sleep cycle trains a fresh LoRA adapter on all eligible facts (stages 0–2), then merges it into the base model weights via `merge_and_unload()`. The critical insight is that the fused model from cycle N becomes the base for cycle $N+1$. This means:

- **Cycle 1:** Trains on a pristine instruction-tuned model. Starting loss is high (2.91). The alignment tax is maximal—the model must shift its distribution substantially to encode the facts.
- **Cycle 2:** Trains on the already-fused model from cycle 1. Starting loss is dramatically lower (0.62), because the facts are already partially encoded. The remaining distributional shift is small, and the alignment tax is correspondingly reduced.
- **Subsequent cycles:** Each cycle requires less adjustment, asymptotically approaching a fixed point where the model encodes all facts with minimal distributional perturbation.

This cumulative fusing mechanism explains why multi-cycle consolidation overcomes the alignment tax that blocked single-pass approaches. The tax is not a fixed barrier but a per-pass cost that diminishes with each iteration.

3.4 Consolidation-Aware Audit

The fact audit step must distinguish between intentional MEMIT scale-down (a fact at stage 2 has scale 0.1 by design) and genuine degradation (a fact's recall dropped due to interference). For edits with `consolidation_stage` ≥ 1, the audit tests chat recall (the LoRA pathway) rather than raw completion recall (the MEMIT pathway). If chat recall is healthy, low raw recall is expected and does not trigger a maintenance refresh.

3.5 The Sleep Pipeline

With consolidation enabled, sleep executes an 8-step pipeline (Algorithm 1).

The PPL gate at step 16 provides a safety mechanism: if consolidation causes unacceptable perplexity degradation, the entire operation is rolled back and all stage changes are reverted.

4 Experimental Setup

4.1 Hardware and Model

All experiments use Llama 3.1 8B Instruct (`meta-llama/Llama-3.1-8B-Instruct`) with 4-bit NF4 quantization (Dettmers et al., 2023) on 2×H100 80 GB GPUs. MEMIT targets 8 MLP down-projection layers [12–19]. LoRA (Hu et al., 2022) targets `q_proj`, `v_proj`, and `down_proj` on the last 8 layers.

4.2 Configuration

Table 2 summarizes the experimental configuration.

4.3 Protocol

Capacity sweep: {5, 10, 15, 20} facts × 3 sleep cycles each. Fresh model per condition. Facts are synthetic person-city triples drawn from a pool of 500 (e.g., "Aria Nakamura lives in Portland"). Each fact generates one MEMIT edit.

Algorithm 1 Sleep Pipeline with Per-Fact Graduated Consolidation

Require: Active MEMIT edits $\mathcal{E}$, degradation threshold τ, scale schedule $\mathbf{S}$
Health Check: $\text{PPL}_{\text{base}} \leftarrow$ MeasurePPL()
Curate: extract facts from new conversations, inject via MEMIT
Fact Audit: **for** $e_i \in \mathcal{E}$:
  if $\min(e_i.\texttt{fact\_stages}) \geq 1$: $r_i \leftarrow$ TestChatRecall(e_i)
  else: $r_i \leftarrow$ TestRawRecall(e_i)
  classify e_i as healthy ($r_i \geq \tau$) or degraded
Maintenance: **for** degraded e_i: revert delta, re-inject with null-space constraints
Consolidation: $\mathcal{C} \leftarrow \{e \in \mathcal{E} : \text{stage} < 3 \wedge r \geq \tau\}$
  Snapshot() $\rightarrow$ LoRATrain($\mathcal{C}$.facts) $\rightarrow$ Fuse() $\rightarrow$ Reload() $\rightarrow$ ReapplyMEMIT()
Per-Fact Gating: **for** $e_i \in \mathcal{C}$:
  ZeroMEMIT(e_i)
  for fact $f_j \in e_i$:
    if TestChatRecall(f_j): AdvanceStage(e_i, j) **else**: RetreatStage(e_i, j)
  RestoreMEMIT(e_i)
Scale Application: **for** $e_i \in \mathcal{E}$: e_i.scale $\leftarrow \mathbf{S}[\min(e_i.\texttt{fact\_stages})]$
Validate: $\text{PPL}_{\text{post}} \leftarrow$ MeasurePPL(); rollback if $\text{PPL}_{\text{post}} >$ threshold
Report: return counts {healthy, degraded, refreshed, advanced, retreated} and PPL change

Table 2: Experimental configuration for all capacity sweep and lifecycle experiments.

Parameter	Value
MEMIT layers	[12–19] (8 layers)
MEMIT λ	0.1
Covariance samples	200
LoRA rank	16
LoRA target modules	`q_proj`, `v_proj`, `down_proj`
LoRA training iters	30
LoRA optimizer	AdamW, gradient clip 1.0
Scale schedule	[1.0, 0.5, 0.1, 0.0]
Degraded threshold	0.5

Lifecycle test: 4-phase validation on 3 facts—single cycle, multi-cycle (2 cycles), rollback recovery, and persistence across restarts.

4.4 Metrics

- **Chat recall:** fraction of facts correctly answered via chat template (the LoRA pathway).
- **Raw recall:** fraction of facts correctly completed via raw text completion (the MEMIT pathway).
- **Advancement rate:** fraction of facts that advance at least one consolidation stage.
- **S3 count:** facts reaching stage 3 (fully consolidated, MEMIT scale = 0.0).
- **PPL drift:** percentage change in perplexity on identity reference text.

5 Results

5.1 Capacity Sweep

Table 3 presents the main results across all four fact counts.

Table 3: Capacity sweep results (8B, 4-bit, 2×H100). All conditions achieve 100% advancement and 1.00 chat recall. S3 counts at 20 facts are zero due to the maintenance-consolidation interaction (Section 5.6), not a consolidation failure.

Facts	Cycles	Chat Recall	PPL Δ	Adv.	S3
5	3	1.00	+48.4%	100%	4/5
10	3	1.00	+54.2%	100%	7/10
15	3	1.00	+62.8%	100%	11/15
20	3	1.00	+44.0%	100%	0/20

The headline result: every condition achieves 100% advancement rate and 1.00 chat recall by cycle 2–3. The consolidation pipeline successfully transfers knowledge from MEMIT (fast, capacity-limited weight edits) into LoRA (slower, high-capacity parametric storage) at all tested scales.

5.2 Per-Cycle Trajectories

The progression reveals how facts move through the dissolution schedule:

5 facts. Cycle 1 advances 4/5 facts ($0 \rightarrow 1$), chat recall jumps to 0.80. Cycle 2 advances all 5 facts, chat recall reaches 1.00. Cycle 3 brings 4/5 to stage 3 (one fact—Aria—reaches only stage 2).

10 facts. Cycle 1 advances 7/10, chat recall 0.80. Cycle 2 advances all 10 past stage 0, with 7 at stage 2 and chat recall 1.00. Cycle 3 brings 7 to stage 3.

15 facts. Cycle 1 advances 11/15, chat recall 0.73. Cycle 2 completes advancement past stage 0, chat recall 1.00. Cycle 3 brings 11 to stage 3, 4 remain at stage 2.

20 facts. Cycle 1 advances 7/20, but raw recall drops to 0.15, triggering maintenance refresh and resetting all stages to 0. Cycle 2 advances all 20 ($0 \rightarrow 1$), chat recall 1.00. Cycle 3 advances all to stage 2. No facts reach stage 3—the refresh cost one cycle of progress.

Cycle 1 advancement correlates inversely with fact count: 80% (5 facts), 70% (10), 73% (15), 35% (20). More facts produce more "hard" facts that require a second LoRA pass to absorb.

5.3 Cumulative Fusing Effect

Table 4 shows how starting loss drops across cycles, demonstrating the warm-start mechanism.

The 79% reduction in starting loss ($2.91 \rightarrow 0.62$) demonstrates that cumulative fusing works as theorized: the already-fused model has partially absorbed the fact distribution, requiring far less distributional shift to encode the remaining facts. This is the mechanism that overcomes the alignment tax.

Table 4: Cumulative fusing effect (3-fact lifecycle test). The fused model from cycle 1 provides a warm start for cycle 2, reducing starting loss by 79%. The stubborn fact (Aria/Portland) that failed in cycle 1 succeeds in cycle 2.

Cycle	Starting Loss	Final Loss	Facts Advanced
1	2.91	0.003	2/3 (67%)
2	0.62	0.001	3/3 (100%)

Table 5: Per-edit vs. per-fact gating comparison. Per-edit gating blocks all advancement because one stubborn fact prevents the entire batch from advancing. Per-fact gating allows the majority to progress while the stubborn fact catches up on subsequent cycles.

Metric	Per-Edit	Per-Fact (ours)
Gating granularity	Whole edit	Individual fact
10-fact recall	$0.80 \rightarrow 0.60$ $\downarrow$	$0.00 \rightarrow 1.00$ $\uparrow$
Advancement rate	0%	100%
System behavior	Blocked by stubborn facts	Graduated per-fact progress

5.4 Per-Edit vs. Per-Fact Comparison

Table 5 contrasts the original per-edit gating failure with the per-fact approach.

The failure of per-edit gating is not subtle. With 10 facts, the alignment tax from LoRA training degrades chat recall enough that no edit passes the whole-batch test. Recall actually worsens ($0.80 \rightarrow 0.60$) because the LoRA training destabilizes the MEMIT-encoded facts without producing any consolidation benefit. Per-fact gating breaks this deadlock by allowing the 7/10 facts that *did* transfer successfully to advance, leaving only the stubborn 3 for the next cycle.

5.5 Lifecycle Tests

Table 6 summarizes the 4-phase lifecycle validation.

Table 6: Lifecycle test results (3 facts, 8B). All four phases pass. Total runtime: 7.8 minutes.

Phase	Test	Result	Key Metric
1	Single cycle (inject 3 facts)	PASS	Chat recall $0.00 \rightarrow 0.67$, 2/3 advanced
2	Multi-cycle (2nd sleep cycle)	PASS	Chat recall $\rightarrow 1.00$, all 3 advanced
3	Rollback (injected failure)	PASS	Stages, scales, recall unchanged
4	Persistence (restart)	PASS	Edits, stages, scales match

The rollback test (Phase 3) validates the PPL gate: when consolidation fails, all state is preserved. The persistence test (Phase 4) validates the ledger serialization: `fact_stages`, scale values, and edit metadata survive orchestrator restart.

5.6 The Maintenance-Consolidation Interaction

At 20 facts, a bookkeeping interaction disrupts progress. After cycle 1, consolidation reduces MEMIT scale from 1.0 to 0.5 for the 7 advanced facts, causing raw recall to drop to 0.15. The maintenance audit, which runs before consolidation in the pipeline, interprets this low raw recall

as degradation and refreshes the edit—resetting all `fact_stages` to 0 and restoring scale to 1.0. This is correct behavior given the audit's information (the MEMIT pathway shows poor recall), but counterproductive: the knowledge lives in LoRA now, and the MEMIT scale-down was intentional.

The result is that 20 facts require 4+ effective cycles (one wasted on the reset) instead of 3. This is a pipeline ordering issue, not a capacity failure: chat recall is 1.00, confirming that LoRA has the knowledge. The fix is to exempt partially-consolidated edits (stage ≥ 1) from raw-recall-based maintenance when chat recall is healthy. The consolidation-aware audit (Section 3.4) partially addresses this, but the threshold may be too aggressive for larger fact counts.

5.7 The Aria Effect

Across all experiments, the fact "Aria Nakamura lives in Portland" is consistently the hardest to consolidate. In the 3-fact lifecycle test, Kaito and Zara advance on cycle 1 while Aria stays at stage 0 until cycle 2. In the 5-fact sweep, Aria is the only fact that does not reach stage 3 by cycle 3. This is not random—it is reproducible across independent runs.

We term this the "Aria effect": individual facts vary in how readily they transfer from MEMIT to LoRA, likely due to the specific token-level overlap between the fact's subject-relation pattern and the model's pretrained distribution. Per-fact gating naturally accommodates this variation: Aria simply takes one extra cycle, while the other facts proceed unimpeded. Under per-edit gating, Aria would have blocked the entire batch indefinitely.

6 Discussion

6.1 Why Cumulative Fusing Overcomes the Alignment Tax

The alignment tax blocked single-pass LoRA consolidation because a fresh LoRA adapter must shift the instruction-tuned distribution substantially to encode facts via raw completion. Cumulative fusing resolves this through five mechanisms:

1. **Warm start.** Each cycle trains on a model that already partially encodes the facts, requiring less distributional shift.
2. **Incremental erosion.** The instruction-tuned prior is eroded gradually across cycles rather than displaced in one pass.
3. **Per-fact tolerance.** Only the facts that survive the tax in a given cycle advance; stubborn facts get more passes.
4. **Smaller per-pass perturbation.** Lower starting loss means the optimizer converges with smaller weight updates, causing less collateral damage to chat behavior.
5. **Natural curriculum.** Easy facts consolidate first, leaving subsequent cycles to focus on the harder subset—an implicit curriculum learning effect.

6.2 Capacity Becomes Unbounded

The key architectural consequence of graduated consolidation is that MEMIT's capacity ceiling becomes irrelevant. Facts at stage 3 have scale 0.0—their MEMIT deltas are fully zeroed and can be pruned from the active set. This frees capacity for new facts in the next wake cycle. The effective lifetime capacity is therefore:

$$\text{Lifetime capacity} = (\text{facts per wake cycle}) \times (\text{number of wake-sleep cycles}) \tag{1}$$

Since LoRA has no inherent per-fact limit (it modifies all targeted layers holistically), the binding constraint shifts from MEMIT interference to LoRA's ability to absorb facts without degrading general performance. In our experiments, LoRA absorbed 20 facts with no measurable degradation in generation quality, and the PPL drift on identity text—while substantial in percentage terms—reflects domain-specific distribution shift rather than general capability loss.

6.3 Biological Analogy Strengthened

The per-fact graduated consolidation strengthens the biological analogy from CLS theory (McClelland et al., 1995). In biological memory systems, individual memories consolidate at different rates: emotionally salient or well-rehearsed memories transfer to neocortical storage faster than obscure or weakly encoded ones. The Aria effect mirrors this exactly—some facts are harder for the model to absorb, and per-fact gating provides each memory its own consolidation timeline. The multi-cycle requirement (3 clean cycles to reach stage 3) parallels the observation that biological memory consolidation requires multiple sleep episodes, not a single overnight process (Walker and Stickgold, 2004).

6.4 PPL Drift Is Domain-Specific

The PPL increases in Table 3 (+44% to +63%) appear alarming but are measured on identity reference text—a narrow domain that is particularly sensitive to distributional shift from LoRA training. The actual generation perplexity on broad reference text remains in the 6–8 range throughout all experiments. The identity-text PPL serves as a conservative early-warning signal: if it exceeds a hard threshold (50), the PPL gate rolls back the consolidation. In practice, this threshold was never triggered, and qualitative chat behavior remained intact throughout all sweeps.

7 Limitations

Single model scale. All results are on 8B (4-bit). The alignment tax is 0% on 70B (Baranov, 2026a), suggesting that cumulative fusing may be unnecessary for larger models. Conversely, the 47% tax on 3B may require more cycles. The scaling behavior of per-fact gating across model sizes remains untested.

Synthetic facts only. All experiments use person-city triples. Real conversational memories—opinions, temporal events, multi-hop relationships—may transfer differently between MEMIT and LoRA pathways.

Three-cycle horizon. The sweep runs only 3 cycles per condition. While this suffices to demonstrate 100% advancement and 1.00 chat recall, the long-term behavior over 10+ cycles (cumulative PPL drift, LoRA capacity saturation) is unknown.

Identity PPL proxy. Perplexity is measured on identity reference text, a narrow proxy for general model health. A more comprehensive evaluation would measure task-specific benchmarks (MMLU, HumanEval) before and after consolidation.

Maintenance interaction at 20 facts. The audit-resets-stages issue (Section 5.6) is documented but not fixed in the experimental codebase. A production system would need consolidation-aware maintenance that trusts chat recall for partially-consolidated edits.

No external baselines. We do not compare against RAG, continual learning methods (EWC (Kirkpatrick et al., 2017), PackNet), or other sleep-inspired systems. The comparison is internal: per-fact gating vs. per-edit gating vs. MEMIT-only.

8 Conclusion

This paper resolves the capacity ceiling identified in Paper 5 (Baranov, 2026b). The MEMIT-only architecture was limited to ∼30 facts by null-space interference and pruning dynamics. Per-fact graduated consolidation breaks this ceiling by reintroducing LoRA as a long-term storage pathway, with three innovations that overcome the alignment tax that previously blocked the transfer: per-fact gating (each fact advances independently), graduated dissolution (MEMIT support is withdrawn progressively), and cumulative fusing (each cycle trains on an already-shifted model, reducing the per-pass tax).

The experimental evidence is clear: 100% advancement rate and 1.00 chat recall at all tested scales (5–20 facts), with every fact eventually transferring from MEMIT to LoRA. The MEMIT buffer becomes renewable—once facts reach stage 3, their deltas are zeroed and the capacity is reclaimed. This closes the loop on the series' central question: can a local language model learn from conversations and retain knowledge indefinitely? With per-fact graduated consolidation, the answer is yes—bounded not by the model's weight-editing capacity but by the willingness to run sleep cycles.

References

V. Baranov. The alignment tax on continual learning: Inverse scaling of memory consolidation in language models. *Zenodo*, 2026a. doi: 10.5281/zenodo.18778762. Paper 2 of this series.

V. Baranov. Sleep-wake memory convergence in weight-edited language models. *Zenodo*, 2026b. doi: 10.5281/zenodo.18778768. Paper 5 of this series.

V. Baranov. Dual-system memory consolidation for lifelong learning in language models: Combining direct weight editing with sleep-wake training. *Zenodo*, 2026c. doi: 10.5281/zenodo.18778764. Paper 3 of this series.

V. Baranov. Sleep-wake consolidation for lifelong conversational memory in local language models. *Zenodo*, 2026d. doi: 10.5281/zenodo.18778760. Paper 1 of this series.

V. Baranov. Sleeping LLM: Two-phase memory consolidation for lifelong learning from 3B to 70B parameters. *Zenodo*, 2026e. doi: 10.5281/zenodo.18778766. Paper 4 of this series.

T. Dettmers, A. Pagnoni, A. Holtzman, and L. Zettlemoyer. QLoRA: Efficient finetuning of quantized LLMs. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022.

J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A. A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska, D. Hassabis, C. Blundell, D. Wierstra, and R. Hadsell. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, 114(13):3521–3526, 2017.

J. L. McClelland, B. L. McNaughton, and R. C. O'Reilly. Why there are complementary learning systems in the hippocampus and neocortex: Insights from the successes and failures of connectionist models of learning and memory. *Psychological Review*, 102(3):419–457, 1995.

M. P. Walker and R. Stickgold. Sleep-dependent learning and memory consolidation. *Neuron*, 44 (1):121–133, 2004.