CIRRUS: REVOCABLE DURABILITY CONTRACTS FOR CROSS-SESSION MEMORY

Anonymous authors
Paper under double-blind review

ABSTRACT

A procedure can be correct when written to an agent's memory, superseded in a later session, and still steer every query until its durable entry is withdrawn. Queries answered from that superseded memory value incur *correction debt*, which we operationalize as stale failure over revision-eligible targets. CIRRUS (Contradiction-Informed Reversible Retention Under Sessions) makes the lifetime of each fact, preference, and procedure explicit: a durability contract binds versioned source evidence, calibrated future utility and revision risk, dependent retrieval keys, and an executable post-grant revocation predicate. A shared-candidate 2×2 factorial isolates the lifecycle mechanism. With learned admission fixed, withdrawal raises delayed task success by 3.4 points and lowers stale failure from 11.9% to 5.1%; with withdrawal fixed, learned admission adds 1.7 success points over transparent rules. The complete system reaches 72.6% success, 4.2% unsupported answers, and 5.1% stale failures on 5,400 CONSOLBENCH episodes at 15 aggregate A100 GPU-min per update window, improving success by 4.5 points over compressive memory at the same update compute. The ordering persists under procedure changes, multi-hop selective forgetting, and unlabeled conflicts: on STALE, frozen CIRRUS scores 63.1% versus 56.7% for CUPMem. Durable memory therefore benefits from treating promotion as a revocable, evidence-carrying contract rather than a terminal write.

1 INTRODUCTION

Consider a procedure that succeeds in two sessions, becomes durable, and is later replaced before a delayed task. If its first version remains retrieval-eligible, once-valid evidence becomes a repeated source of wrong action. We call each revision-eligible query that fails from such a superseded value *correction debt*. On CONSOLBENCH, flat summarization improves success over episodic retrieval (67.8% versus 65.2%) yet raises stale failure from 8.1% to 12.3%. The resulting systems question is when remembered state should stop acting as current.

Long-horizon agents retrieve, reflect on, compress, and organize prior interaction (Park et al., 2023; Zhong et al., 2024; Lewis et al., 2020; Rae et al., 2020; Packer et al., 2023; Chhikara et al., 2025; Xu et al., 2025). These operations determine what can enter context, but an individual durable value also needs a lifecycle: evidence for promotion, an estimate of future service, an identity that survives paraphrase, and a later observation that terminates active use. Retrieval relevance cannot express that lifecycle because an obsolete value can remain highly relevant; whole-store rollback is unnecessarily broad when one procedure or preference changes.

CIRRUS implements the lifecycle without changing the acting model. A *durability contract* attaches independent support, future-utility and revision-risk scores, source lineage, dependent retrieval keys, status, and a grant epoch to each candidate fact, preference, or procedure. Its temporal evidence graph keeps successive values separate. Admission can promote, defer, or discard a version; post-grant evidence can revoke the active root and its dependent closure; a corrected cluster must earn a new grant. Promotion and withdrawal are thus executable operations over the same versioned object.

The central experiment follows this object from write to revision to withdrawal. On 5,400 held-out CONSOLBENCH episodes, a shared-candidate 2×2 factorial varies learned versus transparent admission and reversible versus irreversible withdrawal while fixing candidates, types, graphs, provenance, exposure, update compute, and seeds. Withdrawal under learned admission raises

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

success by 3.4 points and removes 6.8 stale-failure points; learned admission under reversible control adds another 1.7 success points. Complete-system comparisons then establish utility at the 15 GPU-min operating point, while revision stress and frozen transfers to API-Bank procedures (Li et al., 2023), MemoryAgentBench (Hu et al., 2026), and STALE (Chao et al., 2026) test whether the same contract survives different revision interfaces.

The contribution is one auditable lifecycle in which an entry carries the evidence needed both to acquire and to lose durable status. CONSOLBENCH scores its delayed utility, unsupported use, and correction debt against a current-value ledger; the factorial isolates withdrawal, while evidence-graph and scoring ablations locate support quality and admission selectivity. The complete system reaches 72.6% success with 4.2% unsupported answers and 5.1% stale failures, and scores 63.1% on unlabeled STALE conflicts, 6.4 points above CUPMem.

2 RELATED WORK AND TECHNICAL POSITIONING

2.1 READ SELECTION, WRITE TRANSFORMATION, AND WRITE LIFETIME

Retrieval and paging systems select evidence for the current context (Lewis et al., 2020; Guu et al., 2020; Borgeaud et al., 2022; Izacard et al., 2023; Asai et al., 2024; Packer et al., 2023); summarization and compression transform older context under capacity pressure (Rae et al., 2020; Munkhdalai et al., 2024; Qian et al., 2025). Their operational unit is a query context, summary, or latent state. CIRRUS instead acts on a versioned fact, preference, or procedure whose sources justify a grant and whose post-grant evidence can revoke that exact version. End-to-end comparisons measure the resulting system, while the shared-candidate factorial identifies revocation without representation differences.

MemoryBank uses time and importance to reinforce or forget conversational memories (Zhong et al., 2024); Mem0 extracts and updates persistent items, and A-MEM links and evolves structured notes (Chhikara et al., 2025; Xu et al., 2025). These systems establish selective and organized memory. The durability contract adds a supervised future-service target, an explicit post-grant evidence epoch, dependency-aware revocation, and a new grant for a corrected version. RecMem triggers consolidation from semantic recurrence (Dai et al., 2026); CIRRUS treats recurrence as support and separately estimates whether the candidate will be useful and whether its value is likely to change. The distinctive object is therefore not a richer note but a version whose authority can be granted, traced, withdrawn, and earned again.

2.2 EXPLICIT ROLLBACK AND INFERRED INVALIDATION

ChronoMem versions whole-memory snapshots and maps an explicit rollback request to a historical state (Su et al., 2026). CIRRUS instead resolves a contradiction attached to an entry identity and invalidates only dependent keys; Appendix C compares the two interfaces directly. MemoryAgentBench tests selective forgetting under incremental interaction (Hu et al., 2026), whereas STALE withholds revision labels and CUPMem propagates inferred state invalidation (Chao et al., 2026). We therefore use RecMem on MemoryAgentBench, CUPMem on STALE, and ChronoMem in explicit-versus-implicit correction as lifecycle-native external comparators. The core factorial adds mechanism-level identification to these system comparisons by enabling withdrawal alone on one fixed candidate representation.

3 DURABILITY CONTRACTS UNDER REVISION

3.1 STATE, CONTRACTS, AND DECISIONS

An *episode* is one task-bearing interaction with an observable outcome; a *session* is an ordered sequence of episodes between consolidation boundaries. The agent stores raw, source-addressable episodes in $\mathcal{E} = \{e_1, \ldots, e_N\}$ and promoted facts, preferences, and procedures in $\mathcal{D} = \mathcal{D}^f \cup \mathcal{D}^p \cup \mathcal{D}^r$. A *candidate span* is grounded text extracted from one or more episodes and canonicalized into an entry identity. Each candidate lifecycle is represented by a durability contract

$$d_c = (c, \tau_c, S_c, q_c, \rho_c, \Delta_c, z_c, g_c),$$

where τ_c is the memory type, S_c contains source pointers, q_c is the admission certificate, ρ_c is the revocation predicate, Δ_c records dependent retrieval keys and entries, $z_c \in \{\texttt{active}, \texttt{deferred}, \texttt{demoted}, \texttt{discarded}\}$ is lifecycle status, and g_c is the boundary at which the current value version became active. A *revision event* supplies later evidence that changes the currently valid value or procedure; a *revision storm* links multiple such events within a short session interval.

Let $\mathcal{Q}_{\text{rev}}$ be delayed targets for which a value was superseded before the query. Set $D_q = 1$ when q is wrong and its answer or action is grounded in that superseded value, and $D_q = 0$ otherwise. The correction-debt rate is

$$\text{CD} = \frac{100}{|\mathcal{Q}_{\text{rev}}|} \sum_{q \in \mathcal{Q}_{\text{rev}}} D_q.$$

Thus the reported stale-failure percentage is the empirical correction-debt rate. This query-level definition measures the repeated behavioral burden created while a retired value remains influential and makes withdrawal directly comparable inside the factorial.

At each boundary, the policy chooses $a_c \in \{\texttt{promote}, \texttt{defer}, \texttt{discard}\}$:

$$a_c = \texttt{promote} \iff \hat{u}(c) > \theta_u \wedge \hat{r}(c) < \theta_r \wedge \text{supp}_t(c) \succeq \boldsymbol{\theta}_s^{\text{type}(c)}. \tag{1}$$

The support vector $\text{supp}_t(c) = (n_t(c), m_t(c))$ counts non-duplicate supporting instances and their distinct source sessions for the currently valid value cluster; $\succeq$ requires both coordinates to meet the type-specific threshold. Thus $q_c = (\hat{u}(c), \hat{r}(c), n_t(c), m_t(c))$. Promotion stores every supporting pointer, records the current boundary as g_c, and instantiates ρ_c from post-grant evidence. A candidate that fails exactly one clause of Equation 1 but has no admission-time contradiction above 0.2 remains deferred for at most 3 windows; a second failed clause, a hard contradiction, or expiration closes that candidate version as discarded while preserving its raw sources. Later evidence starts a new version rather than reopening the discarded contract.

Withdrawal has two ordered steps. First, a post-grant contradiction that satisfies ρ_c demotes the contradicted root. Second, CIRRUS traverses Δ_c and demotes every active entry whose stored evidence or successful execution path depends on that root; duplicate roots and dependents are applied once per boundary. Both steps mark the affected durable version inactive and remove its retrieval keys, while all timestamped source spans remain in $\mathcal{E}$. After 1 complete window, a corrected value can be promoted only from a new consistent cluster that independently satisfies Equation 1; its new g_c prevents evidence against the retired version from immediately demoting the correction.

3.2 CANDIDATE IDENTITY AND EVIDENCE GRAPH

Extraction is restricted to pre-boundary evidence with an addressable source. The extractor emits a maximal declarative span for a grounded world or user claim, an explicit stated or selected option for a preference, and the minimal tool-and-argument subsequence ending in a successful task outcome for a procedure. Conflicting type cues defer the candidate. Canonical keys normalize entity aliases, argument names, temporal qualifiers, and procedure signatures; within one type and key, exact matches and MinHash Jaccard similarity at least 0.8 define duplicate spans. Duplicates within one session collapse to one instance. Exact canonical-key agreement establishes a shared slot. For non-identical keys, embedding search only proposes same-type neighbor pairs; DeBERTa-v3-large (MNLI) (He et al., 2021) creates an entailment or contradiction edge only above 0.8 confidence. It does not merge a pair by itself. Pairs below threshold and pairs with ambiguous type or slot identity remain separate and cannot contribute mutual support.

For each canonical slot or procedure signature, CIRRUS maintains a directed temporal graph $G_c = (V_c, E_c)$: nodes are source spans or successful action subsequences, value-consistent nodes form clusters, and edges point from earlier evidence to later evidence. Node features record session, timestamp, episode success, memory type, and source pointer. For the latest consistent cluster $K_t(c)$,

$$n_t(c) = |\{v \in K_t(c) : \text{entail}(v, c) > 0.8\}|,$$
$$m_t(c) = |\{\text{session}(v) : v \in K_t(c), \text{entail}(v, c) > 0.8\}|.$$

Entailment-connected nodes in the same slot and value form one consistent cluster; contradiction-connected nodes occupy competing temporally ordered value clusters. Admission-time conflict is

the maximum contradiction from evidence later than the cluster's first span. For an active entry, the executable revocation score is time-conditioned:

$$\mathrm{contra}_t^+(c) = \max_{v \in V_c : \mathrm{time}(v) > g_c} \mathrm{contradict}(v, c).$$

We define the maximum over an empty post-grant set as zero. One post-grant contradiction above 0.7 triggers root demotion. The graph retains older contradictions for audit, but they are outside contra_t^+ after a corrected value receives a new grant epoch. If several later spans disagree, temporal precedence selects the candidate value and the highest-confidence latest consistent cluster supplies its support. A directed dependency edge $c \to d$ means that d's stored source list or successful execution path references c; Δ_c stores these outward dependents for transitive invalidation. Query-time retrieval scores raw episodic spans and active durable entries in one common dense space and returns the joint top k under the shared token cap. Demoted entries cannot serve as durable premises, although their source spans remain eligible as timestamped episodic evidence.

3.3 Typed Contracts and Learned Scores

Facts use $\boldsymbol{\theta}_s^f = (3 \text{ instances}, 2 \text{ sessions})$, admission-time contradiction below 0.2, and 48 hours without a value change. Preferences use $(2, 2)$ consistent choices with explicit user-stated or user-chosen provenance. Procedures use $(2, 2)$ successful executions of the same canonical tool-and-argument pattern and aggregate downstream success above 70%. "Recent evidence" in the demotion pass means a source first observed after g_c and no later than the current consolidation boundary.

Each memory type has two scorers, one for utility and one for revision risk; the shared-scorer ablation in Table 8 replaces these six type-specific heads with one utility and one risk head. A scorer consumes the pooled frozen Llama-3.1-8B candidate embedding, source diversity, both support coordinates, admission-time contradiction, recency, and originating outcome. Each is a three-layer MLP with 768 and 256 hidden units, GELU activations, and 0.1 dropout. For utility supervision, every training candidate is replayed against eligible queries in the next 5 sessions twice with the actor seed, query, non-candidate context, and token budget fixed: once with the candidate eligible for retrieval and once with it masked. The label is positive when inclusion changes task failure to success and negative otherwise. Risk labels mark whether the candidate's value is superseded within the next 10 sessions. All heads use the extracted candidates of their type, inverse-frequency class weights in binary cross-entropy, and Adam at 3×10^{-4} for 30K steps.

Scorer development and endpoint evaluation use disjoint scenario identities. Candidate-lifecycle examples in the scorer-development pool are split chronologically 70/15/15; label lookahead is truncated at each partition boundary. Development data calibrate each score by isotonic regression and choose thresholds that maximize task success subject to development-set unsupported, stale-failure, native state-cap, and update-compute constraints. The 5,400 ConsolBench endpoint episodes are a separate held-out evaluation set, not the 15% scorer-test share. No ConsolBench target or future development/test event enters scorer labels, calibration, thresholds, candidate features, or pre-query retrieval state.

4 Evaluation: Benchmarking Revision Lifecycles

4.1 ConsolBench Construction and Lifecycle Estimands

ConsolBench is a controlled benchmark constructed from typed scenario specifications rather than a resplit of an external conversation collection. A specification defines an initial fact, preference, or executable tool procedure; the interactions that reveal or exercise it; any later replacement; and a delayed task whose success depends on the currently valid version. The scenario—all of its sessions, value versions, revisions, and delayed targets—is the indivisible sampling and split unit. This prevents paraphrases or pre- and post-revision views of one state from crossing development and evaluation.

The endpoint set contains 5,400 held-out task episodes: 2,400 personalization, 1,800 factual-dependence, and 1,200 procedural-reuse episodes. Thirty percent of the factual-dependence scenarios contain revision storms, defined as at least two linked replacements of one canonical slot before its delayed target. A target enters the endpoint set only when its structured state ledger yields one current value or executable procedure and the delayed task has a deterministic success predicate.

Algorithm 1 CIRRUS durability-contract update

Require: Episodic store $\mathcal{E}$, durable store $\mathcal{D}$, deferred queue $\mathcal{Q}$
Extract, canonicalize, and deduplicate grounded candidates
Update temporal evidence graphs with new support and contradictions
for each candidate c in the current boundary and $\mathcal{Q}$ **do**
  Compute type, $\hat{u}(c)$, $\hat{r}(c)$, and $\mathrm{supp}_t(c)$
  if Equation 1 holds **then**
    Promote c with source pointers and grant epoch g_c; remove it from $\mathcal{Q}$
  else if exactly one grant clause fails and no hard contradiction is present **then**
    Defer c in $\mathcal{Q}$
  else
    Discard c from $\mathcal{Q}$ while retaining its raw sources
  end if
end for
for each $d \in \mathcal{D}$ touched by recent evidence **do**
  if $\mathrm{contra}_t^+(d) > 0.7$ **then**
    Demote root d and each active entry in its transitive dependent closure
    Remove their durable retrieval keys and start re-promotion cooldowns
  end if
end for
return Updated $\mathcal{E}$, $\mathcal{D}$, and $\mathcal{Q}$

Table 1: CONSOLBENCH validity chain. Gold state and revision metadata are used for benchmark construction and scoring, never exposed to a memory method.

Stage	Constructed object	Leakage and validity control
State source	Typed fact, explicit preference, or executable tool-and-argument procedure with a current-valid-value ledger	One scenario identity owns every realization and value version
Interaction history	Timestamped source episodes grouped into ordered sessions; personalization scenarios span 10–20 sessions	Only evidence observed before a query enters memory or retrieval
Revision	A later interaction replaces the ledger value; linked replacements form a revision storm	Methods observe ordinary interactions, not revision or stale labels
Delayed target	A later answer or tool task whose task-specific success predicate uses the current ledger value	Superseded values are retained only as distractors and stale-error references
Supervision split	Separate scenario identities for scorer development and the 5,400-episode endpoint set	Scorer labels and lookahead stop at partition boundaries; thresholds freeze before evaluation

Unsupported and stale annotations are then derived independently from the evidence and supersession ledgers. The 70/15/15 split in Section 3 applies to separate scorer-development examples, so it does not partition or reduce this 5,400-episode denominator.

A procedure trajectory illustrates the scored object: two successful source sessions license promotion; a later successful replacement creates a competing value cluster and post-grant contradiction; the next boundary demotes the old root and its dependents. The replacement must independently meet the support rule. A delayed target using the old signature after replacement is stale, while a pre-replacement target is correct under the earlier ledger state. Facts and preferences follow the same lifecycle with type-specific support.

Task success measures the delayed episode objective. An unsupported answer lacks entailment from the current episodic-plus-active-durable evidence pool or conflicts with entailed evidence. Stale

Table 2: Controlled resources at the primary operating point. Native state caps are not equated across representation types; all rows share query context, retrieval depth, actor calls, splits, and tuning protocol.

Method	GPU-min	Native post-state cap	Available input
Episodic Retrieval	0	5,400 episodes	Raw spans and timestamps
Flat Summarization	15	1024 tokens/window	Raw spans and timestamps
Replay Memory	15	2048 traces	Raw spans, outcomes, timestamps
Compressive Memory	15	1024 latents/window	Raw spans and timestamps
Light Offline FT	15	7.4M parameters	Raw spans and outcomes
Factorial controls	15	2600 entries	Shared candidates, types, evidence, provenance
CIRRUS	15	2600 entries	Candidates, types, evidence, provenance

failure, equivalently the empirical correction-debt rate, is a wrong answer grounded in a value superseded before the target. Task success and unsupported-answer rate use all 5,400 targets; stale failure uses only post-revision targets with a supersession opportunity. The central mechanism estimand is the scenario-paired learned-reversible minus learned-irreversible effect: candidates, graphs, provenance, compute, and all non-withdrawal operations are identical. Its designated mechanism endpoint is correction debt, with task success and unsupported use completing the outcome vector. The paired complete-system contrast with Compressive Memory at 15 GPU-min estimates the combined deployed stack, while the factorial supplies withdrawal-specific attribution.

The Static Chat control contains 900 episodes outside CONSOLBENCH and has no delayed dependence. Per-type results use follow-up memory probes rather than the episode-completion objective, so they are not algebraic components of aggregate task success.

4.2 MODELS, BASELINES, AND CONTROLLED RESOURCES

The acting agent is Llama-3.1-70B (Grattafiori et al., 2024); Llama-3.1-8B supplies frozen candidate embeddings, and DeBERTa-v3-large (MNLI) supplies method-side evidence edges. Runs use 8×A100-80GB, CUDA 12.4, and PyTorch 2.3.1. All methods receive the same 4096 retrieved-token context, retrieval $k = 8$, one actor inference call per query, split, tuning budget, and pre-boundary episodic exposure. The matched experimental dimensions are query access and update compute; storage is reported in each representation's native unit rather than treated as a common-capacity quantity.

Episodic Retrieval applies dense retrieval over raw traces and incurs zero *consolidation* cost, not zero total inference cost. Flat Summarization uses Llama-3.1-8B with deterministic decoding to replace the oldest 25% of traces by a 1024-token summary. Replay Memory retains 2048 outcome-stratified traces. Compressive Memory uses a 7.4-million-parameter learned bottleneck. Light Offline FT applies rank-16 LoRA to attention projections with matched optimizer FLOPs. Every non-episodic method receives 15 aggregate A100 GPU-min per window. Candidate extraction, graph updates, and scoring are charged to CIRRUS.

The per-window ledger starts when a consolidation boundary opens and ends when the updated state is queryable. It includes trace encoding, summarization or replay selection, optimizer steps, CIRRUS extraction and NLI passes, graph and dependency updates, and scorer inference. One-time training and calibration of learned components occur on development data under the common tuning protocol and are outside every method's per-window ledger. Consequently, "matched budget" below denotes the controlled update-time and query-time estimand, not equal serialized bytes or equal total research compute.

4.3 UNCERTAINTY AND EVALUATOR INDEPENDENCE

Table 4 reports means and 95% percentile-bootstrap intervals from five full runs. A 10-seed audit additionally shares scenarios, actor randomness, and scorer seeds. Its hierarchical bootstrap samples

Table 3: Durability-contract factorial on 10 paired seeds. The axes vary authority grant (transparent or learned) and withdrawal (irreversible or reversible); every cell shares candidates, types, evidence graphs, provenance, budgets, exposure, and compute. Values are means ± 95% hierarchical-bootstrap intervals.

Scoring	Write lifecycle	Success ↑	Unsupported ↓	Stale ↓
Transparent	Irreversible	67.7 ± 0.8	5.9 ± 0.4	12.4 ± 0.6
Transparent	Reversible	70.9 ± 0.7	5.1 ± 0.4	6.7 ± 0.5
Learned	Irreversible	69.2 ± 0.7	4.8 ± 0.4	11.9 ± 0.6
Learned	Reversible (CIRRUS)	72.6 ± 0.6	4.2 ± 0.3	5.1 ± 0.4

scenario identities first, retaining every session, episode, and revision, then samples scorer seeds and forms paired differences. Five-run intervals describe absolute performance; ten-seed intervals support paired contrasts. Appendix A gives denominators and intervals for the remaining analyses.

Unsupported and stale labels use an entailment evaluator instantiated separately from the method-side graph. We further audit 600 answers sampled across methods, memory types, revision status, and predicted label. Method names and memory-store metadata are removed, answer order is randomized, and two expert annotators independently apply written definitions of support, supersession, and source precedence before adjudicating disagreements. They inspect only generated answers and supplied evidence records; no user conversation or annotator attribute is an experimental variable. Table 7 reports agreement and detector validity, and GPT-4o (OpenAI, 2024) provides a heterogeneous evaluator check. DeBERTa evidence checks are inputs only to CIRRUS and the shared factorial, and evaluation labels are produced independently for every method.

5 RESULTS: REVOCATION CHANGES THE MEMORY LIFECYCLE

5.1 WITHDRAWAL COLLAPSES QUERY-LEVEL CORRECTION DEBT

The learned-admission pair is the direct test of revocable durability. Enabling withdrawal adds 3.4 task-success points (95% interval [2.6,4.2]), changes unsupported answers by -0.6 points ([-0.9,-0.3]), and removes 6.8 correction-debt points ([-7.6,-6.0]). Under transparent admission, the corresponding cell-mean effects are +3.2, -0.8, and -5.7 points. The same-direction effects under both grant rules identify withdrawal as the operation that changes stale use.

Within reversible control, learned utility and risk scoring add 1.7 success points and reduce unsupported and stale outcomes by 0.9 and 1.6 points; paired intervals are [1.1,2.3], [-1.2,-0.6], and [-2.3,-0.9]. Under irreversible writes, the corresponding cell-mean effects are +1.5, -1.1, and -0.5 points. Withdrawal therefore supplies the larger correction effect, while learned admission improves selectivity.

The component ablations complete the mapping from operation to failure. Removing promotion lowers task success by 7.4 points (72.6 to 65.2); disabling root and cascade withdrawal raises correction debt by 6.8 points (5.1 to 11.9); and removing the evidence graph raises unsupported answers by 2.7 points (4.2 to 6.9). Appendix Table 8 reports the complete outcome vector, including shared-scorer and single-score controls.

5.2 COMPLETE-SYSTEM UTILITY AT FIXED UPDATE COMPUTE

CIRRUS improves delayed task success by 4.8 points over Flat Summarization and 4.5 points over Compressive Memory, the strongest complete-system baseline in Table 4, while reducing unsupported answers by 41% relative to summarization. The paired 10-seed intervals for CIRRUS minus summarization are [4.0,5.6] on task success, [-3.4,-2.4] on unsupported answers, and [-8.1,-6.3] on stale failure; against Compressive Memory, they are [3.8,5.2], [-3.1,-2.1], and [-7.5,-5.9]. These comparisons establish the utility of the complete contract stack under matched update compute and query access; Table 3 supplies the representation-controlled attribution.

Table 4: Complete-system CONSOLBENCH utility, provenance, and correction endpoints. Means ± 95% intervals over five full runs; cost is aggregate A100 GPU-min per consolidation window.

Method	Success ↑	Unsupported ↓	Stale ↓	Cost
Episodic Retrieval	65.2 ± 0.9	9.8 ± 0.6	8.1 ± 0.5	0
Flat Summarization	67.8 ± 0.8	7.1 ± 0.5	12.3 ± 0.7	15
Replay Memory	66.4 ± 0.9	8.4 ± 0.6	9.7 ± 0.6	15
Compressive Memory	68.1 ± 0.7	6.8 ± 0.5	11.8 ± 0.6	15
Light Offline FT	67.3 ± 1.0	7.5 ± 0.6	13.1 ± 0.8	15
CIRRUS	72.6 ± 0.6	4.2 ± 0.3	5.1 ± 0.4	15

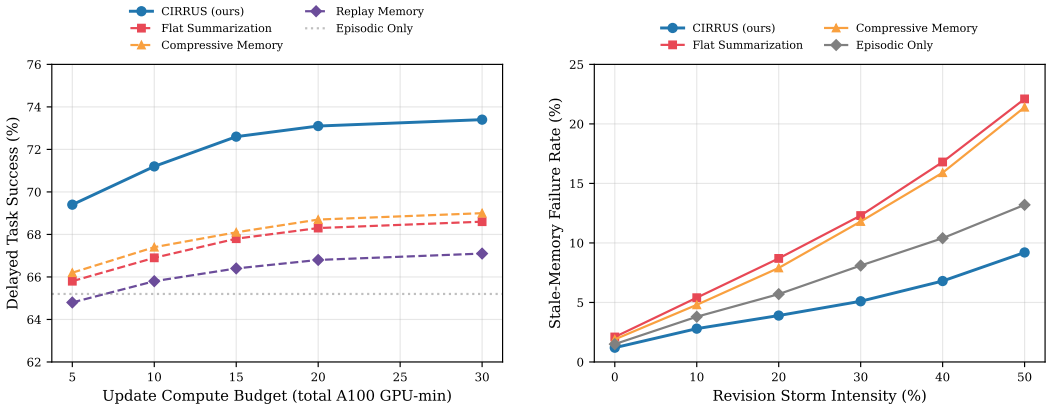

Figure 1: Left: delayed task-success means across controlled update-compute budgets. Right: stale-failure means as revision intensity increases. The curves are descriptive sweeps; paired operating-point uncertainty appears in Tables 3 and 4.

Across the plotted 5–30 A100 GPU-min range, the mean ordering is unchanged, separating the observed advantage from one favorable update budget. Revision pressure exposes the lifecycle difference: at 50% intensity, stale failure is 9.2% for CIRRUS, 11.3% for the transparent reversible gate, and 22.1% for summarization. Among revision-bearing episodes, CIRRUS improves 61.8%, ties 26.7%, and harms 11.5%; the highest-intensity quintile contributes 46.2% of the stale-failure reduction.

5.3 REVISION WITHOUT THE CONSOLBENCH INTERFACE

CIRRUS has the highest reported value in every row of Table 5. On API-Bank-derived procedures, its paired success difference from Compressive Memory is [1.8,3.4] points and from the reversible heuristic is [0.8,2.2]; stale-failure intervals for the same contrasts are [-4.4,-2.6] and [-2.0,-0.6]. MemoryAgentBench changes both the interaction construction and target, with the largest separation on multi-hop selective forgetting, where a stale premise can corrupt a reasoning chain. STALE removes explicit revision labels: CIRRUS reaches 68.7% state-resolution, 61.5% premise-resistance, and 59.2% policy-adaptation accuracy. This last result tests whether contradiction evidence can be inferred from ordinary subsequent interaction rather than supplied as a revision event.

5.4 REVISION PRESSURE LOCALIZES THE GAIN

Facts and procedures show the largest gains because repeated evidence and later contradiction directly inform their lifecycle. Preference gains remain positive. On Static Chat, where no delayed state is needed, the 0.6-point CIRRUS–summarization difference is small relative to the overlapping marginal 95% intervals. Revision-bearing probes account for 78.4% of the aggregate probe gain, connecting the type analysis to revision rather than generic chat quality.

Table 5: External revision interfaces. API reports task success; MemoryAgentBench reports FactConsolidation single-hop / multi-hop accuracy; STALE reports overall accuracy. Values are percentages; full intervals and metrics appear in Appendix C.

Workload	Strongest task baseline	Reversible heuristic	CIRRUS
API-Bank-derived procedures	Compressive 66.5	67.6	69.1
MemoryAgentBench FactConsolidation-SH	RecMem 66.5	68.9	73.4
MemoryAgentBench FactConsolidation-MH	RecMem 12.8	17.1	23.8
STALE implicit conflicts	CUPMem 56.7	55.8	63.1

Table 6: Follow-up memory-probe success (%). Static Chat is a separate 900-episode negative control. These probe means do not aggregate to the episode-level outcome in Table 4.

Method	Facts	Preferences	Procedures	Static Chat
Episodic Retrieval	61.3 ± 1.1	68.4 ± 0.9	64.8 ± 1.2	78.2 ± 0.7
Flat Summarization	64.1 ± 0.9	70.2 ± 0.8	67.1 ± 1.0	78.8 ± 0.6
Compressive Memory	64.9 ± 0.8	70.5 ± 0.7	67.8 ± 0.9	79.0 ± 0.6
Heuristic reversible gate	67.8 ± 0.8	72.1 ± 0.7	69.8 ± 0.9	79.1 ± 0.6
CIRRUS	69.3 ± 0.7	73.3 ± 0.6	71.4 ± 0.8	79.4 ± 0.6
Δ vs. summarization	+5.2	+3.1	+4.3	+0.6

6 DISCUSSION: THE VALUE OF A REVOCABLE WRITE

The write-to-revision trajectory explains the result more precisely than the complete-system ranking. With representation and admission fixed, withdrawal removes 6.8 correction-debt points and adds 3.4 success points. The root operation and its deterministic dependent closure are both part of this lifecycle intervention. A separate ablation raises unsupported answers by 2.7 points when the evidence graph is removed, and learned grant criteria add 1.7 success points over transparent reversible rules. The endpoints therefore track distinct contract obligations: evidence supports promotion, prediction selects future service, and post-grant contradiction ends stale authority.

The contract pays where stored state remains useful long enough to be revised. Revision-bearing probes explain 78.4% of the aggregate probe gain, revision stress widens the separation, and the 0.6-point Static Chat difference is small relative to overlapping intervals. The semi-online schedule preserves 77.1% of the offline gain over summarization; API procedures, multi-hop FactConsolidation, and unlabeled STALE conflicts preserve the ordering without refitting. Storage purge succeeds for 97.8% of entries before one cycle and 82.3% beyond five cycles, favoring correction before dependencies accumulate.

The evidence covers session-delimited updates, delayed factual or procedural dependence, interaction-visible revisions, and a Llama-3.1-70B actor. The factorial identifies lifecycle operations; complete-system comparisons match update compute and query access across native state units. Residual risk concentrates in low-confidence contradictions and long dependency chains, with 11.5% of revision-bearing episodes harmed. In this regime, transparent withdrawal captures most stale-state benefit and learned scoring adds utility and support selectivity.

7 CONCLUSION

Durable memory is a lifecycle decision, not a terminal write. On the shared CIRRUS representation, withdrawal cuts correction debt from 11.9% to 5.1% and adds 3.4 success points; learned admission adds another 1.7. The complete system reaches 72.6% success with 4.2% unsupported answers on 5,400 delayed targets, and frozen-contract evaluations preserve the ordering across three external revision interfaces. Versioned evidence lets an agent retain useful experience while ending a superseded value's influence.

AI USE STATEMENT

Generative AI models were used as follows: Llama-3.1-70B acted as the evaluation agent and Llama-3.1-8B supplied frozen candidate embeddings (Section 4); GPT-4o provided the heterogeneous evaluator check (Section 4 and Appendix A). Generative AI tools also assisted with language and presentation editing during manuscript preparation. No other AI-assisted research or preparation tasks are documented in the available manuscript-preparation record.

REFERENCES

Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. Self-RAG: Learning to retrieve, generate, and critique through self-reflection. In *The Twelfth International Conference on Learning Representations*. OpenReview.net, 2024. URL https://openreview.net/forum?id=hSyW5go0v8.

Sebastian Borgeaud, Arthur Mensch, Jordan Hoffmann, Trevor Cai, Eliza Rutherford, Katie Millican, George Bm Van Den Driessche, Jean-Baptiste Lespiau, Bogdan Damoc, Aidan Clark, Diego De Las Casas, Aurelia Guy, Jacob Menick, Roman Ring, Tom Hennigan, Saffron Huang, Loren Maggiore, Chris Jones, Albin Cassirer, Andy Brock, Michela Paganini, Geoffrey Irving, Oriol Vinyals, Simon Osindero, Karen Simonyan, Jack Rae, Erich Elsen, and Laurent Sifre. Improving language models by retrieving from trillions of tokens. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 2206–2240. PMLR, 2022. URL https://proceedings.mlr.press/v162/borgeaud22a.html.

Hanxiang Chao, Yihan Bai, Rui Sheng, Tianle Li, and Yushi Sun. STALE: Can LLM agents know when their memories are no longer valid?, 2026. URL https://arxiv.org/abs/2605.06527.

Prateek Chhikara, Dev Khant, Saket Aryan, Taranjeet Singh, and Deshraj Yadav. *Mem0: Building Production-Ready AI Agents with Scalable Long-Term Memory*, volume 413 of *Frontiers in Artificial Intelligence and Applications*, pp. 2993–3000. IOS Press, October 2025. ISBN 9781643686318. doi: 10.3233/FAIA251160. URL https://ebooks.iospress.nl/doi/10.3233/FAIA251160.

Zijie Dai, Shiyuan Deng, Sheng Guan, Yizhou Tian, Xin Yao, Xiao Yan, and James Cheng. RecMem: Recurrence-based memory consolidation for efficient and effective long-running LLM agents. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens (eds.), *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 32353–32376, San Diego, California, United States, July 2026. Association for Computational Linguistics. ISBN 979-8-89176-395-1. doi: 10.18653/v1/2026.findings-acl.1619. URL https://aclanthology.org/2026.findings-acl.1619/.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, et al. The Llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024. doi: 10.48550/arXiv.2407.21783. URL https://arxiv.org/abs/2407.21783.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. REALM: Retrieval-augmented language model pre-training. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 3929–3938. PMLR, 2020. URL https://proceedings.mlr.press/v119/guu20a.html.

Pengcheng He, Jianfeng Gao, and Weizhu Chen. DeBERTaV3: Improving DeBERTa using ELECTRA-style pre-training with gradient-disentangled embedding sharing. *arXiv preprint arXiv:2111.09543*, 2021. doi: 10.48550/arXiv.2111.09543. URL https://arxiv.org/abs/2111.09543.

Yuanzhe Hu, Yu Wang, and Julian McAuley. Evaluating memory in LLM agents via incremental multi-turn interactions. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/fd1eff9dd295df50a41f2521942fa31d-Paper-Conference.pdf.

Gautier Izacard, Patrick Lewis, Maria Lomeli, Lucas Hosseini, Fabio Petroni, Timo Schick, Jane Dwivedi-Yu, Armand Joulin, Sebastian Riedel, and Edouard Grave. Atlas: Few-shot learning with retrieval augmented language models. *Journal of Machine Learning Research*, 24(251):1–43, 2023. URL https://www.jmlr.org/papers/v24/23-0037.html.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems*, volume 33, pp. 9459–9474, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/6b493230205f780e1bc26945df7481e5-Abstract.html.

Minghao Li, Yingxiu Zhao, Bowen Yu, Feifan Song, Hangyu Li, Haiyang Yu, Zhoujun Li, Fei Huang, and Yongbin Li. API-Bank: A comprehensive benchmark for tool-augmented LLMs. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 3102–3116. Association for Computational Linguistics, 2023. doi: 10.18653/v1/2023.emnlp-main.187. URL https://aclanthology.org/2023.emnlp-main.187/.

Tsendsuren Munkhdalai, Manaal Faruqui, and Siddharth Gopal. Leave no context behind: Efficient infinite context transformers with infini-attention. *arXiv preprint arXiv:2404.07143*, 2024. URL https://arxiv.org/abs/2404.07143.

OpenAI. GPT-4o system card. *arXiv:2410.21276*, 2024. URL https://arxiv.org/abs/2410.21276.

Charles Packer, Sarah Wooders, Kevin Lin, Vivian Fang, Shishir G Patil, Ion Stoica, and Joseph E Gonzalez. MemGPT: Towards LLMs as operating systems. *arXiv preprint arXiv:2310.08560*, 2023. doi: 10.48550/arXiv.2310.08560. URL https://arxiv.org/abs/2310.08560.

Joon Sung Park, Joseph C O'Brien, Carrie Jun Cai, Meredith Ringel Morris, Percy Liang, and Michael S Bernstein. Generative agents: Interactive simulacra of human behavior. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*. Association for Computing Machinery, 2023. doi: 10.1145/3586183.3606763. URL https://doi.org/10.1145/3586183.3606763.

Hongjin Qian, Zheng Liu, Peitian Zhang, Kelong Mao, Defu Lian, Zhicheng Dou, and Tiejun Huang. MemoRAG: Boosting long context processing with global memory-enhanced retrieval augmentation. In *Proceedings of the ACM Web Conference 2025*, pp. 2366–2377. Association for Computing Machinery, 2025. doi: 10.1145/3696410.3714805. URL https://doi.org/10.1145/3696410.3714805.

Jack W Rae, Anna Potapenko, Siddhant M Jayakumar, Chloe Hillier, and Timothy P Lillicrap. Compressive transformers for long-range sequence modelling. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=SylKikSYDH.

Yongye Su, Wujiang Xu, Chaoji Zuo, and Elisa Bertino. ChronoMem: Version control and semantic rollback for large language model agent memory, 2026. URL https://arxiv.org/abs/2607.27773.

Wujiang Xu, Zujie Liang, Kai Mei, Hang Gao, Juntao Tan, and Yongfeng Zhang. A-Mem: Agentic memory for LLM agents. In *Advances in Neural Information Processing Systems*, volume 38, pp. 20004–20031. Neural Information Processing Systems Foundation, Inc., 2025. doi: 10.52202/085713-0593. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/19909c36f51abc4856b4560aff3d36d6-Paper-Conference.pdf.

Wanjun Zhong, Lianghong Guo, Qiqi Gao, He Ye, and Yanlin Wang. MemoryBank: Enhancing large language models with long-term memory. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 19724–19731, 2024. doi: 10.1609/aaai.v38i17.29946. URL https://doi.org/10.1609/aaai.v38i17.29946.

Table 7: Independent evaluator audit over 600 stratified answers. Detector statistics use adjudicated expert labels; deltas are Flat Summarization minus CIRRUS unsupported-answer rates in points.

Audit statistic	Value	95% interval
Inter-annotator κ	0.82	[0.78,0.86]
Unsupported precision	0.91	[0.87,0.94]
Unsupported recall	0.87	[0.82,0.91]
Unsupported F1	0.89	[0.85,0.92]
Stale-label F1	0.86	[0.81,0.90]
Human-adjudicated delta	2.6	[1.7,3.5]
GPT-4o-evaluated delta	2.4	[1.5,3.3]

Table 8: Complete contract ablation matrix. Values are means $\pm$ 95% paired hierarchical-bootstrap intervals.

Variant	Success ↑	Unsupported ↓	Stale ↓
CIRRUS	72.6 ± 0.6	4.2 ± 0.3	5.1 ± 0.4
No promotion	65.2 ± 0.9	9.8 ± 0.6	8.1 ± 0.5
No demotion / learned irreversible	69.2 ± 0.7	4.8 ± 0.4	11.9 ± 0.6
No evidence graph	70.1 ± 0.8	6.9 ± 0.5	6.0 ± 0.5
Shared scorer across types	71.3 ± 0.7	4.9 ± 0.4	6.0 ± 0.5
Utility only	70.8 ± 0.8	5.4 ± 0.5	7.1 ± 0.6
Revision risk only	69.7 ± 0.8	5.0 ± 0.4	6.3 ± 0.5

A STATISTICAL PROTOCOL AND RAW DENOMINATORS

The 10-seed shared-representation audit contains 54000 episode-runs, 18240 revision-eligible query-runs, and 284600 canonical candidate instances. For each bootstrap replicate, scenario identities are sampled with replacement as the highest level; every session, episode, and revision in a sampled scenario is retained. Paired method differences are computed within the replicate before sampling scorer seeds. We report percentile intervals from the resulting paired distribution. Multiple endpoints are presented as an evidence hierarchy rather than independent discovery tests; no claim depends on a thresholded p-value.

The per-type table evaluates 3600 factual, 4800 preference, and 3600 procedure follow-up probes. Of these, 1080, 1344, and 912 follow revision-bearing trajectories. Static Chat contributes 1800 probes from its separate 900-episode control set.

B COMPLETE CONTRACT FACTORIAL AND ABLATION OUTCOMES

All factorial cells use 15 aggregate A100 GPU-min, the 2600-entry state cap, and the common query budget. The transparent irreversible cell admits candidates by the same typed support, recency, and contradiction thresholds but cannot withdraw an accepted entry. The learned irreversible cell uses the CIRRUS utility and risk scores but likewise makes admission permanent. The reversible cells share contradiction-triggered root demotion followed by deterministic dependent-closure demotion and differ only in transparent versus learned grant criteria. Rounded CIRRUS aggregates coincide with Table 4; paired effects and intervals come from the 10-seed audit.

C EXTERNAL REVISION BENCHMARKS

C.1 API-BANK-DERIVED PROCEDURE REVISIONS

We split API-Bank source traces before adaptation, keep every chain derived from one source trace in one partition, sessionize held-out traces into multi-step tool chains, and place externally specified tool or argument changes between sessions. The result contains 720 held-out episodes and 214 revision

Table 9: Candidate-level calibration by memory type. Precision and recall are computed against future held-out use and revision outcomes; ECE is expected calibration error.

Type	Promotion P	Future-use R	Demotion P	Demotion R	ECE
Facts	0.88	0.74	0.91	0.79	0.036
Preferences	0.82	0.69	0.86	0.75	0.049
Procedures	0.85	0.72	0.89	0.77	0.041

Table 10: API-Bank-derived procedure revision results over 10 paired seeds. Values are means ± 95% hierarchical-bootstrap intervals.

Method	Success ↑	Unsupported ↓	Stale ↓
Flat Summarization	66.0 ± 0.9	7.4 ± 0.6	11.0 ± 0.8
Compressive Memory	66.5 ± 0.8	7.0 ± 0.5	10.4 ± 0.7
Heuristic reversible gate	67.6 ± 0.8	6.2 ± 0.5	8.2 ± 0.6
CIRRUS	69.1 ± 0.7	5.7 ± 0.4	6.9 ± 0.5

events. Revision specifications and task-success checks use API signatures rather than outputs from a memory method. 10 paired seeds use Llama-3.1-70B, identical session boundaries, and the controlled update-compute and query budgets. The type-specific CIRRUS scorers, isotonic maps, and promotion thresholds are frozen from CONSOLBENCH; adaptation only maps API schema fields into the existing procedure key. DeBERTa evidence checks are method inputs only for CIRRUS and the factorial controls, while evaluation follows the independent protocol from Section A.

Against Compressive Memory, CIRRUS has paired intervals [1.8,3.4], [-1.9,-0.7], and [-4.4,-2.6]. Against the reversible heuristic, the intervals are [0.8,2.2], [-0.9,-0.1], and [-2.0,-0.6].

C.2 MEMORYAGENTBENCH FACTCONSOLIDATION

We use the official single-hop and multi-hop splits, feed chunks incrementally, and consolidate only at published chunk boundaries. Every method uses one Llama-3.1-70B actor and the common 4096-token query budget; each method that performs consolidation is capped at 15 aggregate A100 GPU-min per boundary. RecMem follows recurrence-triggered consolidation; the heuristic and CIRRUS receive the same canonical candidates used in the factorial. CIRRUS reuses the frozen CONSOLBENCH scorers, calibration maps, thresholds, and type rules; no MemoryAgentBench answer or forgetting label is used for fitting.

C.3 STALE IMPLICIT CONFLICTS

STALE supplies 400 scenarios and 1,200 queries across state resolution, premise resistance, and implicit policy adaptation. No method receives revision labels. CUPMem uses its benchmark-native state adjudication; ChronoMem is omitted here because STALE provides no explicit rollback request. All other methods process the same incremental interactions under controlled context and, where applicable, update-compute budgets. CIRRUS uses the CONSOLBENCH-frozen scorers, calibration, thresholds, and contradiction rule; only the benchmark's incremental interactions can trigger a state change.

C.4 EXPLICIT ROLLBACK VERSUS AUTOMATIC CORRECTION

ChronoMem and CIRRUS expose complementary interfaces. On explicit user-requested whole-state rollback, ChronoMem reaches $84.6 \pm 1.0\%$ consistency and CIRRUS reaches $80.8 \pm 1.1\%$. When the correction must be inferred from subsequent evidence, the ordering reverses: CIRRUS reaches $64.2 \pm 1.2\%$ versus ChronoMem's $49.7 \pm 1.4\%$. Snapshot rollback is strongest when the target state is named; item-level evidence adjudication is strongest when invalidation is implicit.

Table 11: MemoryAgentBench FactConsolidation accuracy (%). Values are paired-scenario means ± 95% bootstrap intervals over 10 scorer seeds.

Method	Single-hop ↑	Multi-hop ↑
Full Context / RAG	63.8 ± 1.1	8.9 ± 0.8
Flat Summarization	64.7 ± 1.0	11.3 ± 0.9
Compressive Memory	65.4 ± 0.9	12.1 ± 0.9
RecMem	66.5 ± 0.9	12.8 ± 0.8
Heuristic reversible gate	68.9 ± 0.8	17.1 ± 1.0
CIRRUS	73.4 ± 0.7	23.8 ± 1.1

Table 12: STALE overall accuracy and stale-response rate (%). Values are scenario-paired means ± 95% bootstrap intervals.

Method	Overall ↑	Stale response ↓
Full Context / RAG	51.1 ± 1.4	31.7 ± 1.6
Flat Summarization	49.7 ± 1.5	33.0 ± 1.7
Compressive Memory	51.4 ± 1.4	30.8 ± 1.6
RecMem	53.1 ± 1.3	28.9 ± 1.5
CUPMem	56.7 ± 1.2	24.1 ± 1.4
Heuristic reversible gate	55.8 ± 1.3	23.5 ± 1.3
CIRRUS	63.1 ± 1.1	18.7 ± 1.2

D SCHEDULES, THRESHOLDS, AND OPERATIONAL COST

This section separates update scheduling from the grant-and-withdrawal mechanism. The cadence sweep holds total update compute fixed, the threshold rows perturb one decision setting at a time, and the systems ledger reports the realized state and query latency at the primary operating point.

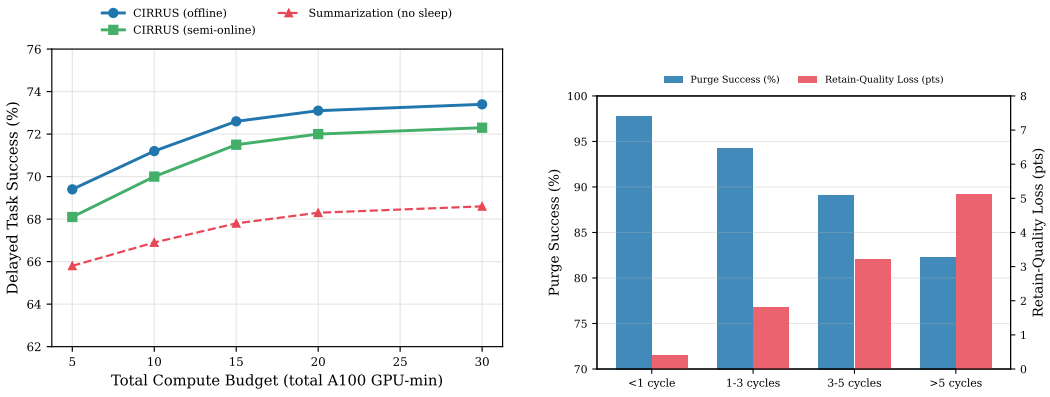

Figure 2: Left: offline and every-five-session consolidation under matched total compute. Right: purge success and retain-quality loss by durable-entry age.

Table 13: Cadence and threshold robustness over 10 paired seeds. Values are success / stale failure (%); intervals are 95% hierarchical-bootstrap half-widths.

Setting	Success ↑	Stale ↓
Every 1 session	70.7 ± 0.8	6.2 ± 0.5
Every 3 sessions	71.2 ± 0.7	5.8 ± 0.5
Every 5 sessions	71.5 ± 0.7	5.6 ± 0.4
Every 10 sessions	70.3 ± 0.8	6.6 ± 0.6
Irregular {2,6,4,8}	70.8 ± 0.8	6.1 ± 0.5
Lower $\theta_u = 0.55$	72.1 ± 0.7	5.5 ± 0.4
Higher $\theta_r = 0.35$	71.9 ± 0.7	5.4 ± 0.4
Lower $\theta_s^f = 2$	71.7 ± 0.8	6.0 ± 0.5
DeBERTa-v3-base edges	71.6 ± 0.8	5.8 ± 0.5

Table 14: Systems cost at the matched operating point. State is the active post-consolidation footprint; latency includes retrieval and one actor call.

Method	Active state	Tokens/query	p50 (s)	p95 (s)
Episodic Retrieval	5400 traces	4096	1.82	3.94
Flat Summarization	1024 tokens/window	4096	1.54	3.21
Replay Memory	2048 traces	4096	1.73	3.62
Compressive Memory	1024 latents/window	4096	1.59	3.38
Light Offline FT	7.4M parameters	4096	1.48	3.17
CIRRUS	2480 entries	4096	1.63	3.46

The 2600-entry value in Table 2 is the enforced capacity; the observed active footprint at this operating point is 2480 entries. Across the audit, CIRRUS records 5316 promotions and 1611 total demotions: 1157 contradicted roots and 454 active dependents in their deduplicated closures. The transparent reversible gate records 5682 promotions and 1184 total demotions: 812 roots and 372 dependents. These reconciled counts show that learned risk admits fewer entries and that the withdrawal intervention includes the same root-then-closure rule in both reversible cells, without changing query context or actor calls.

Table 15: Storage-level revocation by entry age. Purge success and retain-quality loss are percentages / points; latency is seconds.

Entry age	Purge success ↑	Retain loss ↓	Latency
<1 cycle	97.8	0.4	2.1
1–3 cycles	94.2	1.8	3.7
3–5 cycles	89.1	3.2	5.4
>5 cycles	82.3	5.1	8.9

Ordinary withdrawal and storage purge test different operations. Withdrawal demotes a contradicted root and its active dependent closure, removes their durable retrieval keys, and preserves source evidence in $\mathcal{E}$. The purge test follows an authorized identity-level removal request: it deletes the named source spans, invalidates every dependent key in Δ_c, and measures whether the remaining retained entries preserve their task quality. The age trend therefore measures dependency entanglement, not the contradiction detector used for ordinary withdrawal.

E Operational Evidence Matrix

Table 16 connects each claim-bearing failure to the state it perturbs and the comparison that identifies its effect. The first two rows correspond to the shared-representation intervention; the third is the separate storage-purge operation defined above.

Table 16: Durability-contract evidence matrix connecting protected state, perturbation, enforcement boundary, comparator, utility, and observed failure.

Contract obligation	Capability, assumptions, and enforcement boundary	Strongest control, setting, utility, and observed failure
Evidence-backed authority grant	An isolated trace may be incorrect; source sessions and evidence scores are auditable inputs. Admission is enforced at the episodic-to-durable boundary.	No-evidence-graph ablation on held-out episodes. Unsupported answers rise 4.2 to 6.9 while full CIRRUS retains 72.6 success. Failure: correlated or misclassified support.
Executable authority withdrawal	Later evidence may contradict an accepted fact; temporal order and source identity are available. Demotion is enforced before retrieval keys are rebuilt.	Learned irreversible factorial and 50% revision stress. Stale failure rises 5.1 to 11.9 without demotion and reaches 22.1 for summarization at 50%. Failure: implicit contradiction below the NLI trigger.
Dependency-aware storage removal	An authorized request names a durable identity; dependency pointers are trusted. Purge and dependent-key invalidation occur inside the external store.	Age-stratified revocation with retained-task quality. Purge success falls from 97.8 to 82.3 as dependencies accumulate. Failure: old entries entangled with downstream promoted state.

F Implementation Details

Facts, preferences, and procedures use $\theta_u = 0.6$, $\theta_r = 0.3$, instance-support thresholds 3, 2, and 2, and distinct-session thresholds 2, 2, and 2. Admission-time contradiction must remain below 0.2; time-conditioned post-grant contradiction above 0.7 triggers demotion. Thresholds are calibrated once on the temporal development partition and frozen. Candidate extraction operates only on grounded pre-boundary spans. Summarization, compression, replay, and LoRA consume the same trace pool and session boundaries. The 15 GPU-min cost denotes aggregate device time per update window rather than 15 minutes on each of eight GPUs.

The Flat Summarization prompt is: "Consolidate the following pre-boundary interactions into durable statements. Preserve source-qualified facts, preferences, procedures, and revisions; do not add unsupported information." Decoding uses temperature 0 and top-p 1.0. Compressive Memory uses two projection layers around its learned bottleneck and trains only those parameters. Replay samples equally across success, failure, memory type, and revision-bearing strata. LoRA updates query, key, value, and output attention projections for the number of optimizer steps that exhausts the same FLOP ledger.

The acting model receives only the method-produced memory context and the current query. Evaluator prompts receive the current evidence pool, current state, and answer, but not method identity. For STALE, all revision and conflict labels are withheld from memory methods; only benchmark scoring uses scenario annotations.