SELECTIVE CONFLUENCE: COVERAGE-CONDITIONED RECOVERY FOR AGENT-MEMORY HANDOFFS

Anonymous authors
Paper under double-blind review

ABSTRACT

Cross-session agents use compact memories to continue work without replaying their full histories. A later decision may nevertheless require an exact identifier, status, or prior outcome that was unknown when the memory was written. Better retention and retrieval do not by themselves specify which evidence may authorize that decision. We introduce SELECTIVE CONFLUENCE, a handoff contract that keeps semantic memory advisory and recovers a claim-selected canonical record from an external archive. A local membership check controls release before model exposure; failed lookup or verification returns DEFER. Under explicit selection and decision assumptions, the authoritative evidence and the decision grounded in it are independent of the memory-update order. In offline reconstruction over 24 recorded-session tasks and five high-load writer states, with frozen exact selection, verified recovery increases correct decisions from 21/120 to 109/120, with no wrong action and positive mean gains on 23/24 tasks. Matched controls separate the information supplied by episode evidence from the integrity enforced by verification. Structured memory and a released memory-system control show that recovery gains depend on missing coverage, while an open-weight full stack exposes record addressing as a separate bottleneck. These results support a verifiable evidence path for exact commitments when compact memory cannot guarantee coverage.

1 INTRODUCTION

Cross-session agents carry compact textual memories forward while retaining detailed records outside the successor prompt (Packer et al., 2023; Wu et al., 2025; Shinn et al., 2023; Wang et al., 2025; Liu et al., 2026). A later task may require an exact identifier, status, date, or outcome that was unknown when memory was written. We study how successors can ground these commitments when summaries omit their support.

Memory systems increasingly learn what to store, link, update, retrieve, and discard (Xu et al., 2025; Yu et al., 2026; Tan et al., 2025; Li et al., 2026), while benchmarks test retrieval, temporal reasoning, knowledge updates, and abstention (Maharana et al., 2024; Wu et al., 2025; Cheng et al., 2026; Hu et al., 2025). Beyond information availability lies an authority question: which representation may justify an exact decision, and when should the system defer? Summary omission, irrelevant retrieval, and alteration of committed records can all prevent grounded decisions, but answer accuracy alone does not distinguish them.

We address this gap with SELECTIVE CONFLUENCE, a successor-facing authority contract. Figure 1 traces four stages: source acquisition, parallel handoff representations, claim-derived recovery, and local verification. Semantic memory provides working context, while the canonical archive preserves exact records independently of lossy writing. Only a selected record verified against an authenticated root is released; failed lookup or verification returns DEFER. The contribution is the interface that connects claim eligibility, evidence authority, and release, using established retrieval and Merkle components (Merkle, 1988). Under a stable relevant selection and a fixed record-grounded decision rule, the authoritative evidence and resulting commitment are independent of the writer's acquisition order.

In this bounded, target-blind handoff, the writer updates memory before seeing the future claim. Order acts through sequential lossy writes, unlike changes in evidence position within a long prompt (Liu et al., 2024) or inference-time permutation control (McIlroy-Young et al., 2024). Our diagnostics

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.

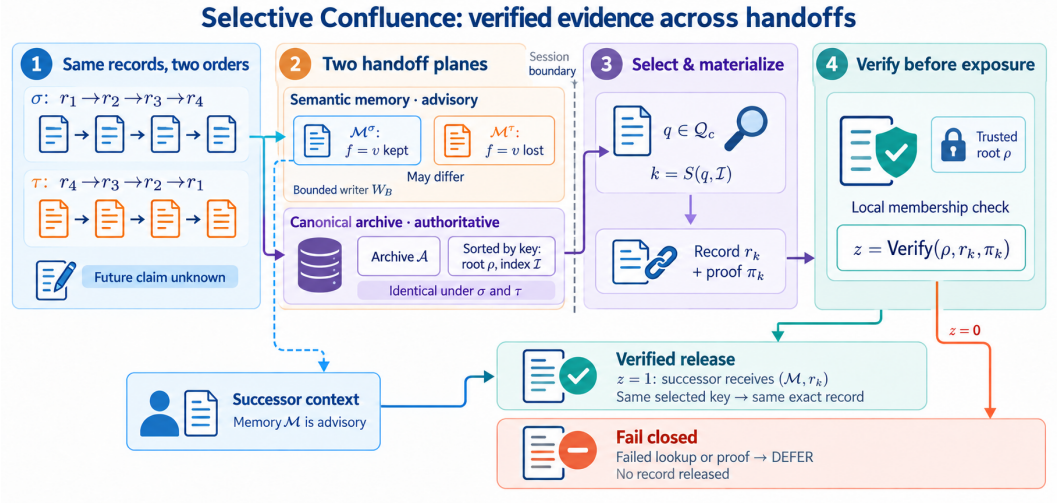


Figure 1: SELECTIVE CONFLUENCE handoff workflow. (1) The same records arrive in two orders before the future claim is known. (2) Bounded semantic memories may differ, while an independently constructed canonical archive has the same root and index under both orders. (3) After handoff, a commitment-bearing claim selects one key and materializes its record and proof. (4) Local membership verification gates record exposure. The lower branches distinguish verified release, with memory retained as advisory context, from failed lookup or verification, which withholds the record and returns DEFER. Same-key selection yields identical canonical evidence; decision invariance additionally requires a fixed record-grounded rule.

find fact-level disagreement across opposite orders and binding loss under overload. Yet a structured writer with sufficient capacity preserves all 96 bindings across all six opposite-order pairs. Recovery addresses uncertain coverage; it does not presume inevitable loss.

The evaluation tests the contract's layers separately. With exact selection, across five high-load writer states and 24 frozen recorded-session tasks, memory resolves 21/120 paired cells and verified recovery resolves 109/120; every writer improves, and 23/24 task directions are positive. Matched retrieval and receipt-free episode controls attribute most of the gain to recovered information, while an altered-record control tests the release gate. Larger-index routing, a local open-weight full stack, and Mem0 distinguish addressing failures from insufficient memory coverage and downstream evidence use. These are offline reconstruction decisions, not live task executions.

Our contributions are an explicit authority model for agent-memory handoffs, a conditional evidence-confluence property for canonical recovery, and an empirical decomposition of coverage, addressing, integrity, and evidence use. Together they identify when recovery helps and which failures verification can resolve. For non-parametric memory reuse, this yields a practical principle: preserve exact state with structure where feasible, and retain verifiable recovery when coverage is uncertain.

2 RELATED WORK

We connect agent-memory management, source recovery, and verification through the authority assigned to each representation at handoff.

Memory as managed agent state. Agent memory now includes synthesized experience, reusable workflows, managed context hierarchies, and persistent retrieval layers (Park et al., 2023; Shinn et al., 2023; Wang et al., 2025; Packer et al., 2023; Chhikara et al., 2025). A-Mem links notes, Agentic Memory jointly manages short- and long-term state, and Pensieve and Sculptor control model-visible context (Xu et al., 2025; Yu et al., 2026; Liu et al., 2026; Li et al., 2026). Reflective memory management also reasons about future and past information needs (Tan et al., 2025). We study when this continuing-agent state may authorize an exact commitment.

Memory evaluation. LoCoMo evaluates conversational memory (Maharana et al., 2024). LongMemEval tests extraction, reasoning, updates, and abstention (Wu et al., 2025). MemoryAgentBench stresses incremental interaction, while AMemGym evaluates policies on-policy (Hu et al., 2025; Cheng et al., 2026). Our evaluation additionally distinguishes support in bounded memory from recovered evidence and scores invalid release separately.

Compression and source revisitation. ReadAgent keeps compact gists and reopens selected passages when detail is needed (Lee et al., 2024). RAPTOR and HippoRAG organize external evidence hierarchically or associatively, while RECOMP suppresses unhelpful retrieval and compresses useful evidence (Sarthi et al., 2024; Gutiérrez et al., 2024; Xu et al., 2024). Self-RAG integrates retrieval and critique with generation (Asai et al., 2024). Our contract complements these context-reduction methods by assigning authority to recovered records through an explicit release rule.

Retrieval quality and evidence authority. LLatrieval uses an LLM to verify retrieved support, and Provenance checks whether generated facts follow from retrieved material (Li et al., 2024; Sankararaman et al., 2024). A Merkle check asks a narrower question: do the released bytes belong to a committed archive? Authenticity and relevance are distinct, so selection and integrity are scored separately.

Order and authority. Long-context use depends on evidence position, while Set-Based Prompting enforces invariance for specified inference-time permutations (Liu et al., 2024; McIlroy-Young et al., 2024). Here, order acts through bounded writes before handoff. We seek update-path-independent evidence for exact commitments, with non-commuting writes serving as a diagnostic.

Verification and memory poisoning. Pisces protects retrieval privacy (Liang et al., 2026); V3DB verifies vector search over a committed snapshot (Qiu et al., 2026). Our Merkle membership check (Merkle, 1988) proves neither relevance nor truth. AgentPoison directly poisons stored content; MINJA injects memories through queries (Chen et al., 2024; Dong et al., 2025). Poison admitted before our commitment still receives a valid proof. The contract complements admission control, source authentication, and semantic checking.

3 Problem Setup and Method

Bounded target-blind handoff. Let $\mathcal{A} = \{r_1, \ldots, r_n\}$ be canonical episode records. A writer observes order σ but not the later claim q, and updates

$$\mathcal{M}_t = W_B(\mathcal{M}_{t-1}, r_{\sigma(t)}), \qquad |\mathcal{M}_t|_{\text{char}} \le C, \quad |\mathcal{M}_t|_{\text{utf8}} \le B. \tag{1}$$

We use $C = 1{,}900$ characters and $B = 2{,}048$ UTF-8 bytes. The writer and successor are separate roles: the writer selects what persists in prompt-visible memory, while the successor answers a claim revealed after handoff. The source is then excluded from the successor prompt, though an external archive may persist. Order therefore acts through lossy state updates, not evidence position in the later prompt.

Coverage and diagnostic non-confluence. For an exact field–value claim, $\mathrm{Cov}(\mathcal{M}, q) = 1$ iff the binding appears in memory. Two orders are diagnostically non-confluent when

$$D(\mathcal{M}_n^{\sigma}, q) \neq D(\mathcal{M}_n^{\tau}, q) \quad \text{for the same canonical archive.} \tag{2}$$

For example, retaining a general account of an update need not preserve the exact status needed for a later action. Coverage tests whether that binding survives; it does not imply that the successor will interpret it correctly. We count fact-level rather than stylistic disagreement. This property is conditional on the writer, budget, source distribution, and successor realization.

Authority and evidence confluence. Let $\mathcal{Q}_c$ contain claims whose proof status or action depends on an exact stored binding. Semantic memory may provide context but is non-authoritative for $q \in \mathcal{Q}_c$. A handoff is evidence-confluent if every order over the same archive releases the same verified canonical record, or the same fail-closed outcome under order-independent failure conditions. Evidence confluence concerns what the interface releases. Decision confluence additionally requires

Table 1: Successor-facing handoff contract. Semantic state supports continuity; exact evidence is independently materialized and verified before release.

Component	Contents	Required property
Semantic memory $\mathcal{M}$	Bounded, target-blind state	Advisory; non-authoritative for $\mathcal{Q}_c$
Commitment index $\mathcal{I}$	Opaque record keys and query fields	Excludes answers and sealed evidence paths
Archive $\mathcal{A}$	Canonical exact records outside the prompt	Retained and governed independently of memory
Root ρ	Integrity commitment to $\mathcal{A}$	Order-canonical; authentically delivered
Selector S	Claim-derived record query	Selects without answer-oracle access
Release gate	(r, π) or DEFER	Verify before exposure; fail closed on rejection

the successor to follow a fixed rule grounded in the released record; stochastic model responses need not satisfy that stronger condition. Schema-valid responses are scored as CORRECT, WRONG-ACTION, or DEFER. We separately report serialized model-visible context under canonical JSON and zlib-9; archive/index storage, selector calls, and latency are excluded from this byte metric.

3.1 SELECTIVE CONFLUENCE

Selective Confluence separates a compact semantic plane from an authoritative commitment plane (Table 1). The former carries goals, summaries, and working context; the latter preserves exact records outside the prompt, without prescribing a storage engine. Panels 1–2 of Figure 1 show their common source: original records feed both planes, so the archive is not reconstructed from the bounded summary. The dashed context path carries semantic memory to the successor separately from exact-record recovery. Linked notes, learned memory policies, and source revisitation can implement the same interface (Xu et al., 2025; Yu et al., 2026; Lee et al., 2024).

Which claims require the commitment plane. Questions about plans, preferences, or broad context may use semantic memory alone. A claim enters $\mathcal{Q}_c$ when its proof status or permitted action depends on an exact stored binding, such as an identifier, status, date, or prior outcome. This policy is fixed before the successor decision; in our experiments it belongs to the task contract rather than being inferred after an answer is known.

Claim-derived recovery. Every $q \in \mathcal{Q}_c$ invokes the commitment plane even when the summary appears sufficient. The index exposes stable keys, record types, and permitted routing fields, but not correctness labels or sealed answer paths. As shown in panel 3 of Figure 1, a selector maps the claim to one key; the archive then materializes one record and its proof:

$$k = S(q, \mathcal{I}), \qquad (r_k, \pi_k) = \mathsf{Materialize}(k, \mathcal{A}). \tag{3}$$

Our selector is learned and schema-constrained; deterministic selection is also compatible. Self-RAG, RECOMP, LLatrieval, and ReadAgent already provide retrieval, critique, or source revisitation (Asai et al., 2024; Xu et al., 2024; Li et al., 2024; Lee et al., 2024). Our method instead specifies what happens after selection and which representation may justify the commitment.

Canonical commitment. Let $\mathrm{Enc}(r_i)$ be a deterministic serialization and k_i its stable key. We form leaves as

$$\ell_i = H(\texttt{record} \,\|\, k_i \,\|\, \mathrm{Enc}(r_i)), \qquad \rho = \mathrm{MerkleRoot}(\mathrm{sort}_k\{(k_i, \ell_i)\}_{i=1}^{n}). \tag{4}$$

Sorting by key makes the root independent of acquisition order. Concatenation denotes an unambiguous canonical key–record encoding; integrity assumes collision-resistant hashing. Another authenticated dictionary could replace the tree; the interface only requires stable keys, canonical bytes, and membership proofs relative to an authentic root (Merkle, 1988).

Verify before use. The local gate in panel 4 computes, before model exposure,

$$z = \mathsf{Verify}(\rho, r_k, \pi_k) \in \{0, 1\}. \tag{5}$$

The lower branches of Figure 1 show the resulting release policy. If $z = 1$, the successor receives semantic context and the authoritative record; otherwise the record is withheld and the system returns DEFER. Verification occurs before prompt construction. Proof metadata need not improve clean reasoning; it decides whether selected bytes may cross the authority boundary.

The release policy also limits replay. A non-commitment claim receives $\mathcal{M}$, a verified commitment claim receives $(\mathcal{M}, r_k)$, and a failed lookup or proof returns DEFER. The successor need not receive the full archive. This reduces successor-visible context, not archive maintenance, selection, or proof construction.

Conditional evidence confluence. If two orders contain the same archive, the index and root are deterministic, the selector returns the same relevant key, and materialization is unique, then both successfully release the same authoritative record for every $q \in \mathcal{Q}_c$. A fixed rule grounded in (q, r_k) is invariant even if semantic memories differ. Appendix A gives the proof and counting motivation. The local gate enforces record integrity; prompted successor behavior is evaluated separately. Selector error, stochastic evidence use, or an unauthenticated root can invalidate the corresponding evidence or decision guarantee.

Trust boundary. We assume authenticated root delivery and trusted archive construction. Membership verification detects post-commitment modification, not truth, relevance, freshness, selector correctness, or honest writes. LLatrieval checks retrieved support sufficiency, while Provenance estimates whether output follows from context (Li et al., 2024; Sankararaman et al., 2024). Memory-poisoning attacks (Chen et al., 2024; Dong et al., 2025) require separate defenses if malicious records are admitted before root creation. The separation makes failures attributable: a missing key is an addressing failure, a rejected path an integrity failure, and a wrong action after a valid relevant record an evidence-use failure.

4 Experimental Design

Correctness follows the registered exact-term or status/action rule, so a supported negative answer can still be scored as deferral. Table 2 maps each panel to a mechanism question; Appendix C describes the prompt contracts and evaluation schedules. Primary panels use provider-reported `gpt-5.6-sol/xhigh`; secondary arms use `gpt-5.5-2026-04-23/xhigh`, local Qwen3-8B-MLX-4bit (Yang et al., 2025), and Mem0 2.0.19 (Chhikara et al., 2025). Every cell is single-shot with zero SDK, application, or task retries, and all five repeated-writer realizations are retained irrespective of outcome.

Development diagnostics. Fixed-state order, deliberately overloaded capacity, and unequal-information recorded-session panels identify the mechanisms and controls needed for confirmation. They are described in Appendix C and are not the repeated-writer utility estimate.

Repeated-writer matched-path confirmation. We first select 12 tasks, four per domain, and generate five writers at two loads: three eight-episode shard memories and one 24-episode memory per realization. The GitLab and Shopping Admin sessions retain the realistic browser structure introduced by WebArena (Zhou et al., 2024), but all successor decisions are evaluated from frozen offline traces rather than live task execution. Six randomized routes compare low- and high-load memory; memory plus metadata-BM25; memory plus correct-episode trace with evaluator receipts removed; and exact selected records with or without verification. The exact selector makes this a coverage/evidence test, not selector learning. Before extension calls, we freeze the deterministic 12-task complement and reuse the same five high-load memories and exact records. The combined memory-versus-recovery analysis thus has all 24 tasks and 120 paired cells; matched-path controls remain on the stratified half.

Capacity-sufficient structured order control. Two synthetic quartets each contain 96 keyed commerce or CRM records. Three writer pairs per domain consume identical packets in opposite orders and return the cumulative key–value map. Complete states fit within 65,536 bytes, removing forced omission; exact canonical equality is the outcome.

Table 2: Evaluation panels and diagnostic roles.

Panel	Intervention	Task coverage	Primary outcome
Structured control	Opposite orders; sufficient key–value capacity	Commerce; CRM	Exact state equality; confluence
Repeated writers	Five retained writers; memory vs. verified records	24 natural tasks; three domains	Writer- and task-level gains
Matched paths	BM25, receipt-free trace, exact-record routes	Stratified 12-task subset	Evidence and retrieval controls
Integrity	Alter record after root creation	Three task families	Rejection; downstream deferral
Transfer	Primary vs. second model	Complete capacity arm	Direction under the same contract
Layer stress	Larger index; local open model; Mem0	Same three domains	Addressing, model, and memory limits

Layer-stress boundary panels. Addressability compares full hints, sparse routing keys, and a 120-episode hint-free index, including same-domain wrong-but-valid receipts. The stricter open-weight arm runs Qwen as five bounded writers, larger-index selector, and paired successor, never replacing a wrong or invalid selection with oracle evidence. The real-system arm writes 24 episodes through Mem0, retrieves top three memories for 12 tasks, and adds verified recovery; its exact-term score is primary and a failure-outcome audit is explicitly post-hoc.

Post-commitment alteration and analysis. For 24 claims, a support value changes after root creation; the unverified successor sees it, whereas the gate rejects it before prompt construction. This tests deterministic fail-closed implementation, not a security rate. We use exact paired tests, task-level signs, and the prospectively specified task–writer resampling; a secondary-run timeout remains missing infrastructure data.

5 RESULTS

Table 3 separates the questions tested by each panel: whether memory retains the binding, retrieval locates it, verification permits release, and the successor uses it correctly.

Table 3: Result map for saved artifacts, retaining all five repeated writers.

Panel	Observed outcome	Evidence boundary
Repeated writers	High-load memory 21/120; verified recovery 109/120; $+14$ to $+20$/writer	All five writers and 23/24 tasks gain
Matched paths	Memory+BM25 30/60; receipt-free trace 52/60; verified exact 54/60	Gain is not reducible to evaluator-receipt lookup
Structured order control	48/48 cumulative states exact; 6/6 pairs confluent	Sufficient key–value capacity removes order effects here
Post-commit alteration	Unverified: 24 wrong; gate: 24 deferrals	Deterministic release check under an authenticated root
Second model	Capacity 12/12 vs. 2/12; natural panel incomplete	Complete capacity replication; natural prefix excluded
Layer stress	Sparse/hint-free selection 12/12 vs. 1/12; open-weight full-stack recovery 15/60; Mem0 exact 8/12 in both routes	Addressing, writer compliance, and evidence use remain distinct from integrity

5.1 RECOVERY BENEFIT REPEATS ACROSS HIGH-LOAD WRITERS

Across five high-load memories and all 24 frozen tasks, semantic memory resolves 21/120 paired cells and verified recovery 109/120, with eleven deferrals and no wrong action. Every writer improves (Table 4). At task level, 23 directions are positive and one negative (two-sided sign test $p = 2.98 \times 10^{-6}$); the prospectively specified two-way task–writer 95% interval is $[0.558, 0.892]$.

As a descriptive cell-level summary, recovery alone is correct in 90 cells and memory alone in two (exact McNemar $p = 1.73 \times 10^{-24}$); the mean paired gain is 0.733. Inference about consistency instead uses the task sign test and the two-way interval, which varies both tasks and writers.

Table 4: Correct decisions and paired gains across 24 frozen tasks per writer. All five independently generated high-load memories are retained without an outcome gate.

Context / writer realization	1	2	3	4	5	Mean
High-load semantic memory	3	4	7	3	4	4.2
Verified exact recovery	23	22	21	21	22	21.8
Paired gain	+20	+18	+14	+18	+18	+17.6

On the stratified 12-task subset (Figure 2, left), exact terms appear in 59/120 low/high-load memory pairs: covered pairs resolve 42/59 and uncovered pairs 0/61. Low-load shards resolve 29/60 versus 13/60 for high-load memory, linking the load contrast to binding coverage. Coverage alone does not ensure correct use.

Across the full 24-task panel, GitLab moves from 9/60 to 60/60 and Shopping Admin from 0/30 to 30/30, but Salesforce moves from 12/30 to 19/30. All eleven unresolved cells and the only negative task direction are Salesforce cases; every unresolved cell is a deferral. In the matched-subset audit, successors describe the recorded failures correctly but map these supported negative answers to deferral. Recovery therefore repairs access to evidence while leaving a decision-policy problem. The result supports a coverage-conditioned gain, not equal domain difficulty, echoing memory benchmarks that separate retrieval from temporal and state-update reasoning (Wu et al., 2025; Cheng et al., 2026).

Matched retrieval and receipt-free evidence. Metadata-BM25 (Robertson & Zaragoza, 2009) raises high-load memory from 13/60 to 30/60 but misses every designated support record and makes three wrong actions. This is a lexical control, not a matched learned selector. A correct-episode trace without the evaluator receipt reaches 52/60 with three wrong actions, so ordinary browser evidence supplies nearly all tested information. Oracle routing tests evidence sufficiency, not retrieval accuracy. Verified exact recovery reaches 54/60 with zero wrong actions (Table 5).

Table 5: Matched paths on 12 tasks and five writers. Defer and wrong are counts out of 60. A route hit is a literal memory binding, designated BM25/exact record, or correct receipt-free trace episode.

Successor context	Correct	Defer	Wrong	Route hit	Route condition
Low-load semantic memory	29/60	31	0	35/60	Eight-episode writer shards
High-load semantic memory	13/60	47	0	24/60	One 24-episode memory
High memory + metadata BM25	30/60	27	3	0/60	Matched semantic baseline
High memory + routed trace	52/60	5	3	60/60	Evaluator receipt excluded
High memory + exact record	53/60	7	0	60/60	Clean; unverified
High memory + verified exact record	54/60	6	0	60/60	Authoritative recovery route

Clean unverified and verified exact routes reach 53/60 and 54/60. This one-cell difference is not evidence that proof metadata improves reasoning. Locating the episode supplies most observed information; the receipt distinguishes attempted actions from the audited outcome, while verification controls release.

5.2 Sufficient structure removes the order effect

All 48 structured states match their cumulative canonical union; 12 final lanes preserve 96/96 bindings, and all six opposite-order pairs are equal (Figure 2, right). A known schema and adequate budget can therefore prevent the tested order effect before recovery is needed. Learned memory managers can reduce loss (Xu et al., 2025; Yu et al., 2026; Liu et al., 2026); recovery addresses the residual case in which coverage cannot be guaranteed.

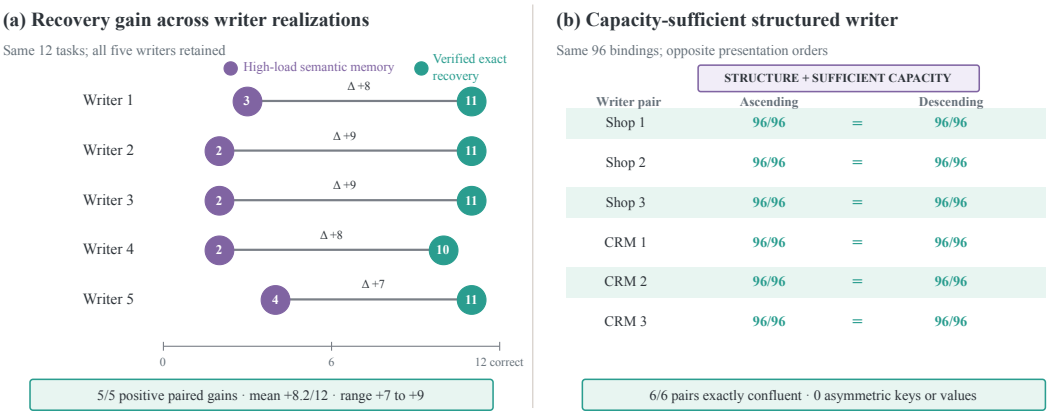


Figure 2: Repeated-writer recovery and the structured negative control. The left panel uses the original stratified matched-control subset, where verified exact recovery improves all five frozen 12-task writer realizations by seven to nine decisions; Table 4 reports the prospectively extended 24-task result. The right panel gives the capacity-sufficient control: with an explicit key–value schema, all six opposite-order writer pairs converge exactly.

5.3 Verification enforces the post-commitment trust boundary

Unverified altered bindings cause 24/24 wrong actions. The gate rejects all 24, sends none to the model, and yields 24 deferrals. This deterministic implementation check confirms rejection before prompt construction, without requiring the successor to recognize alteration. It does not estimate a stochastic security rate or establish source truth or relevance.

5.4 Cross-model capacity evidence

A second provider snapshot reproduces the capacity direction: 12/12 at load 16 and 2/12 at load 28. Its incomplete natural schedule is excluded from outcome evidence and recorded as missing infrastructure data. This supports the deliberately induced coverage crossover across snapshots, not the natural-task recovery magnitude.

5.5 Boundary panels separate addressing, memory, and evidence use

Addressability under archive growth. Full hints in the 24-episode archive and sparse claim-derived keys in the 120-episode archive each select exact support 12/12 times; removing address information from the larger index yields 1/12. Sparse-key and hint-free routes resolve 55/60 and 17/60. Retaining evidence does not ensure it is addressable. Wrong-but-valid receipts yield 12 memory-supported answers, 48 deferrals, and no wrong action. These answers also show that the prompted successor does not enforce an evidence-only commitment rule. Verification cannot reject authentic but irrelevant packages, and zero observed wrong actions is not a safety guarantee.

Open-weight full-stack replication. The local Qwen3-8B stack (Yang et al., 2025) completes 137/137 generations with five bounded memories after truncation, only two byte-unique. Its selector is exact/wrong-valid/invalid for 3/5/4 targets. Memory resolves 0/60; routed verified recovery resolves 15/60, gaining three for every writer with no wrong action. The positive direction transfers, but addressing limits the gain and low state diversity limits writer-level generalization. Invalid keys release no record; subsequent model outputs are retained diagnostically. The exact-routed stress still yields ten wrong actions (Appendix D), separating evidence delivery from correct use.

Real memory-system boundary. Mem0 (Chhikara et al., 2025) retrieves support terms for 10/12 targets. Mem0 top-3 and the same context plus recovery both score 8/12 (McNemar $p = 1$), so stronger memory can remove average exact-term gain. In a post-hoc audit, authenticated negative receipts change two failed Salesforce episodes from false completion to defer, moving outcome fidelity from 7/12 to 10/12. The distinction concerns what is scored: matching a requested term need

Table 6: Layer-stress boundaries. Mem0 compares top-3 memory with the same context plus verified recovery; its outcome audit is deterministic, post-hoc, and exploratory.

Boundary	Observed outcome	Evidence boundary
120-episode selector	Sparse keys 12/12; hint-free 1/12	Claim addressability remains necessary
Wrong-but-valid record	12 correct, 48 defer, 0 wrong / 60	No general safety guarantee
Local Qwen3-8B full stack	Selector exact 3/12; memory 0/60; routed recovery 15/60; 0 wrong	Direction transfers; addressing limits gain
Mem0 registered score	8/12 vs. 8/12; support hit@3 10/12	No average exact-term gain here
Mem0 outcome audit	Fidelity 7/12 vs. 10/12; false completion 2/2 vs. 0/2	Negative receipts guide authority use

not describe the recorded task outcome faithfully. This effect remains exploratory. Table 6 states both outcomes.

6 Discussion and Scope

The authority contract makes the responsibilities of established archives, selectors, and Merkle verification explicit. Memory supports continuity, selection locates a record, and local verification controls its release. These stages address different failures: a better summary cannot authenticate altered bytes, and a valid proof cannot make an irrelevant record useful. Better memory managers and retrievers remain complementary (Xu et al., 2025; Yu et al., 2026; Li et al., 2026; Gutiérrez et al., 2024).

The structured and Mem0 controls identify when recovery adds information. Complete structured state can prevent the tested order effect, while informative retrieved memory can remove average gain under the registered exact-term score. The observed gains arise when memory coverage is insufficient. The authority rule still applies to every commitment-bearing claim, regardless of apparent memory sufficiency. The free-text witness establishes that loss can occur under the tested contract, not that sequential handoff must lose information.

The contract applies to exact commitments over retained records, evaluated here through offline reconstruction. Its decomposition supports diagnosis across writers and retrieval systems without treating every failure as insufficient memory. The empirical scope comprises one closed writer/successor snapshot, a local open-weight stack, and one Mem0 configuration; their domain, routing, and state-diversity boundaries are reported with the corresponding results.

Deployment also requires a trusted archive and authenticated root; membership establishes neither truth, relevance, nor freshness. Pre-commitment poisoning (Chen et al., 2024; Dong et al., 2025) needs separate write-time defenses. Stable addressing and evidence-grounded decisions remain obligations beyond the local integrity gate, and zero observed wrong actions does not discharge them. Reported bytes measure successor-visible context only. Storage, proof construction, selector calls, and latency must be measured separately before drawing an end-to-end cost conclusion.

7 Conclusion

Selective Confluence assigns exact commitments an authority path that remains available when bounded semantic memory omits their support. It resolves claims to canonical records, verifies membership before exposure, and defers invalid releases. Across 24 tasks, five writers show consistent recovery gains; structured capacity removes the tested order effect, while larger-index, open-weight, and Mem0 panels separate addressing, integrity, and evidence use. Preserve known bindings with structure and retain verifiable recovery when coverage is uncertain. Stable relevant selection makes released evidence independent of acquisition order; invariant decisions additionally require a rule grounded in that record.

STATEMENTS

REPRODUCIBILITY STATEMENT

Sections 3–4 define scoring; Appendices A–F give proofs, protocols, schedules, exclusions, and traceability. The supplement includes repeated-writer/24-task extension, structured, selector-risk, open-weight, and Mem0 panels plus serializers, Merkle/BM25 code, schemas, figure data, and replay scripts. Scored cells bind prompt, response, schema, usage, and hash; all planned tasks/writers are retained without retries. Mem0 outcome fidelity is explicitly post-hoc.

AI USE STATEMENT

We used generative AI tools to assist with manuscript polishing, including improvements to grammar, phrasing, and readability. All AI-assisted revisions were reviewed and verified by the authors, who take full responsibility for the final content, for the scientific claims, and for every experimental result that is reported and discussed throughout this paper.

ETHICS STATEMENT

The study has no human-subject intervention or real-user deployment. It uses synthetic/test records and cleaned frozen traces from authorized environments, evaluated offline. Credentials, authentication state, keys, tokens, cookies, and rejected proofs are excluded; public artifacts remove operational identifiers and sensitive fields. Verification can block post-commitment alteration but does not certify source truth, legality, freshness, or ethics.

REFERENCES

Akari Asai, Zeqiu Wu, Yizhong Wang, Avi Sil, and Hannaneh Hajishirzi. Self-RAG: Learning to retrieve, generate, and critique through self-reflection. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/25f7be9694d7b32d5cc670927b8091e1-Abstract-Conference.html.

Zhaorun Chen, Zhen Xiang, Chaowei Xiao, Dawn Song, and Bo Li. AgentPoison: Red-teaming LLM agents via poisoning memory or knowledge bases. In *Advances in Neural Information Processing Systems*, volume 37, 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/eb113910e9c3f6242541c1652e30dfd6-Abstract-Conference.html.

Jiayang Cheng, Dongyu Ru, Lin Qiu, Yiyang Li, Xuezhi Cao, Yangqiu Song, and Xunliang Cai. AMemGym: Interactive memory benchmarking for assistants in long-horizon conversations. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/0856bc553d3e3b9827e5140d0ad3bf8d-Abstract-Conference.html.

Prateek Chhikara, Dev Khant, Saket Aryan, Taranjeet Singh, and Deshraj Yadav. Mem0: Building production-ready AI agents with scalable long-term memory. *arXiv preprint arXiv:2504.19413*, 2025. URL https://arxiv.org/abs/2504.19413.

Shen Dong, Shaochen Xu, Pengfei He, Yige Li, Jiliang Tang, Tianming Liu, Hui Liu, and Zhen Xiang. Memory injection attacks on LLM agents via query-only interaction. In *Advances in Neural Information Processing Systems*, volume 38, 2025. URL https://papers.nips.cc/paper_files/paper/2025/file/42a97bbd9844d2bf68596730af80bcdf-Paper-Conference.pdf.

Bernal Jiménez Gutiérrez, Yiheng Shu, Yu Gu, Michihiro Yasunaga, and Yu Su. HippoRAG: Neurobiologically inspired long-term memory for large language models. In *Advances in Neural Information Processing Systems*, volume 37, pp. 59532–59569, 2024. doi: 10.52202/079017-1902.

Yuanzhe Hu, Yu Wang, and Julian McAuley. Evaluating memory in LLM agents via incremental multi-turn interactions. *arXiv preprint arXiv:2507.05257*, 2025. URL https://arxiv.org/abs/2507.05257.

Kuang-Huei Lee, Xinyun Chen, Hiroki Furuta, John Canny, and Ian Fischer. A human-inspired reading agent with gist memory of very long contexts. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 26396–26415, 2024. URL https://proceedings.mlr.press/v235/lee24c.html.

Mo Li, L. H. Xu, Qitai Tan, Long Ma, Hongyong Song, Ting Cao, and Yunxin Liu. Sculptor: Empowering LLMs with cognitive agency via active context management. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/f9155a06ba1b20e40c38bf8541db456f-Abstract-Conference.html.

Xiaonan Li, Changtai Zhu, Linyang Li, Zhangyue Yin, Tianxiang Sun, and Xipeng Qiu. LLatrieval: LLM-verified retrieval for verifiable generation. In *Proceedings of NAACL-HLT*, pp. 5453–5471, 2024. doi: 10.18653/v1/2024.naacl-long.305.

Xiaojian Liang, Lushan Song, Shishuai Du, Weicheng Zhu, Li Hui Faith Tan, Jun Jie Sim, Haibing Jin, Zhenghao Wu, Yingting Liu, Xin Zhang, Jiang-Ming Yang, and Pu Duan. Pisces: Cryptography-based private retrieval-augmented generation with dual-path retrieval. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/6529ad2d7e2cf7bbc76785684b794d57-Abstract-Conference.html.

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173, 2024. doi: 10.1162/tacl_a_00638.

Xiaoyuan Liu, Tian Liang, Dongyang Ma, Deyu Zhou, Haitao Mi, Pinjia He, and Yan Wang. The pensieve paradigm: Stateful language models mastering their own context. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/8c54e9bfed4119c873f575d1d1e2f0a0-Abstract-Conference.html.

Adyasha Maharana, Dong-Ho Lee, Sergey Tulyakov, Mohit Bansal, Francesco Barbieri, and Yuwei Fang. Evaluating very long-term conversational memory of LLM agents. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics*, pp. 13851–13870, 2024. doi: 10.18653/v1/2024.acl-long.747.

Reid McIlroy-Young, Katrina Brown, Conlan Olson, Linjun Zhang, and Cynthia Dwork. Order-independence without fine tuning. In *Advances in Neural Information Processing Systems*, volume 37, pp. 72818–72839, 2024. doi: 10.52202/079017-2318.

Ralph C. Merkle. A digital signature based on a conventional encryption function. In *Advances in Cryptology—CRYPTO '87*, pp. 369–378, 1988. doi: 10.1007/3-540-48184-2_32.

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. URL https://arxiv.org/abs/2310.08560.

Joon Sung Park, Joseph 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*, pp. 1–22, 2023. doi: 10.1145/3586183.3606763.

Zipeng Qiu, Wenjie Qu, Jiaheng Zhang, and Binhang Yuan. V3DB: Audit-on-demand zero-knowledge proofs for verifiable vector search over committed snapshots. *arXiv preprint arXiv:2603.03065*, 2026. URL https://arxiv.org/abs/2603.03065.

Stephen Robertson and Hugo Zaragoza. The probabilistic relevance framework: BM25 and beyond. *Foundations and Trends in Information Retrieval*, 3(4):333–389, 2009. doi: 10.1561/1500000019.

Hithesh Sankararaman, Mohammed Nasheed Yasin, Tanner Sorensen, Alessandro Di Bari, and Andreas Stolcke. Provenance: A light-weight fact-checker for retrieval augmented LLM generation output. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pp. 1305–1313, 2024. doi: 10.18653/v1/2024.emnlp-industry.97.

Parth Sarthi, Salman Abdullah, Aditi Tuli, Shubh Khanna, Anna Goldie, and Christopher Manning. RAPTOR: Recursive abstractive processing for tree-organized retrieval. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/8a2acd174940dbca361a6398a4f9df91-Abstract-Conference.html.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: Language agents with verbal reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 36, pp. 8634–8652, 2023. doi: 10.52202/075280-0377.

Zhen Tan, Jun Yan, I-Hung Hsu, Rujun Han, Zifeng Wang, Long Le, Yiwen Song, Yanfei Chen, Hamid Palangi, George Lee, Anand Rajan Iyer, Tianlong Chen, Huan Liu, Chen-Yu Lee, and Tomas Pfister. In prospect and retrospect: Reflective memory management for long-term personalized dialogue agents. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics*, pp. 8416–8439, 2025. doi: 10.18653/v1/2025.acl-long.413.

Zora Zhiruo Wang, Jiayuan Mao, Daniel Fried, and Graham Neubig. Agent workflow memory. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 63897–63911, 2025. URL https://proceedings.mlr.press/v267/wang25bx.html.

Di Wu, Hongwei Wang, Wenhao Yu, Yuwei Zhang, Kai-Wei Chang, and Dong Yu. LongMemEval: Benchmarking chat assistants on long-term interactive memory. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/d813d324dbf0598bbdc9c8e79740ed01-Abstract-Conference.html.

Fangyuan Xu, Weijia Shi, and Eunsol Choi. RECOMP: Improving retrieval-augmented LMs with context compression and selective augmentation. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/bda88ed2892f5e61c9a9bf215c566913-Abstract-Conference.html.

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. 17577–17604, 2025. doi: 10.52202/085713-0593. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/19909c36f51abc4856b4560aff3d36d6-Abstract-Conference.html.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025. URL https://arxiv.org/abs/2505.09388.

Yi Yu, Liuyi Yao, Yuexiang Xie, Qingquan Tan, Jiaqi Feng, Yaliang Li, and Libing Wu. Agentic memory: Learning unified long-term and short-term memory management for large language model agents. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics*, pp. 21457–21483, 2026. doi: 10.18653/v1/2026.acl-long.981.

Shuyan Zhou, Frank F. Xu, Hao Zhu, Xuhui Zhou, Robert Lo, Abishek Sridhar, Xianyi Cheng, Tianyue Ou, Yonatan Bisk, Daniel Fried, Uri Alon, and Graham Neubig. WebArena: A realistic web environment for building autonomous agents. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/4410c0711e9154a7a2d26f9b3816d1ef-Abstract-Conference.html.

APPENDIX

A CLAIM-SEPARATION VIEW OF BOUNDED HANDOFFS

This appendix records the counting argument used only as a design lens in Section 3.1.

Let $\mathcal{P}$ be a set of source histories, where each history induces a proof signature $p \in \mathcal{P}$: the collection of exact record bindings required to answer a permitted future claim family $\mathcal{Q}$. The family is *claim separating* if, for every distinct p, p', some $q \in \mathcal{Q}$ requires a different scored decision under the two signatures. Here B counts bits, whereas Equation 1 uses a byte budget. A self-contained B-bit handoff is a map $h : \mathcal{P} \to \{0,1\}^B$.

Proposition 1 (claim-separation lower bound). If $|\mathcal{P}| > 2^B$ and $\mathcal{Q}$ claim-separates $\mathcal{P}$, no self-contained B-bit handoff can support the correct decision for every $p \in \mathcal{P}$ and every $q \in \mathcal{Q}$.

Proof. By the pigeonhole principle, there exist $p \neq p'$ with $h(p) = h(p')$. Claim separation gives a q whose correct decision differs under p and p'. A successor that receives only the common handoff state and q has identical inputs in both cases and therefore cannot return both correct decisions. □

It says that proof-complete self-containment requires a restriction somewhere: fewer future claims, fewer distinguishable signatures, error or abstention, or an external channel. SELECTIVE CONFLUENCE chooses the external-channel option while keeping release bounded to a claim-derived record.

Proposition 2 (conditional evidence confluence). Let $I(A)$ and $R(A)$ be an index and authenticated root that are deterministic functions of the canonical archive set A, independent of presentation order. For a commitment-bearing claim q, assume that $S(q, I(A))$ deterministically returns the same relevant key k, and that materialization returns the unique pair (r_k, π_k) accepted by $\mathsf{Verify}(R(A), r_k, \pi_k)$. Then the authoritative evidence released for q is identical for every source order over A. If a fixed downstream rule conditions its commitment only on (q, r_k), its decision is also identical.

Proof. Every order induces the same $I(A)$ and $R(A)$. The selector therefore returns the same k, materialization returns the same canonical r_k and valid path, and the gate releases the same record. Applying the same downstream rule to identical authoritative inputs gives the same decision. □

The proposition is deliberately conditional. An incorrect selector, unauthenticated root, non-canonical archive, stochastic decision rule, or policy that lets semantic memory override the exact record can reintroduce disagreement.

B OPERATIONAL DEFINITIONS

For structured panels, a claim has the form (e, f, v) for entity alias e, field f, and value alias v. Its exact support record is a canonical tuple (s, e, f, v), where s is an opaque source key. The sealed evaluator maps the tuple to one of two proof statuses and one action. Writers and successors never receive this evaluator mapping. Coverage is computed locally by exact tuple membership after writer output is frozen.

For natural recorded episodes, support is defined by the native evaluator/rubric surface attached to the source browser trajectory. Record boundaries follow a fixed type rule—request, retained browser step, or native evaluator receipt—before successor calls. The designated support record is the episode's evaluator receipt, which contains the recorded outcome. This makes the panel an offline routing and reconstruction diagnostic, not a test of whether the selector can infer an outcome from raw trace evidence. Outcomes use the same expected-term or native-rubric rule across semantic, BM25, and recovery conditions.

C Full Experimental Protocols

C.1 Shared contracts

Writer. The free-text writer receives a sequence of cleaned source packets and an instruction to preserve facts, unresolved commitments, and provenance useful to an unknown successor. It is explicitly told that the later target is unavailable. Output uses a fixed JSON schema with one `memory_text` field, a 1,900-character schema bound, and a local 2,048-byte UTF-8 check. Both constraints instantiate Equation 1; neither substitutes for the other. Earlier gated panels stop on a writer-contract failure; the repeated-writer confirmation instead marks that realization missing without replacement so a failed writer cannot be silently selected away. In the completed run all 20 memories are valid.

Successor. The successor receives one opaque claim or natural task instruction plus condition-specific context. It emits a schema-constrained proof status, action, and concise evidence reference. The local evaluator ignores prose style. Structured panels score the registered status/action pair; natural panels use the frozen expected-term or native-rubric rule. Unsupported actionable output is separated from deferral, and schema-invalid outputs are retained and reported separately where applicable.

Selector. The recovery selector receives the downstream claim and a commitment index. Each index row exposes `record_id`, collection/episode ID, domain, record type, a type-level description, and a truncated task hint; it does not expose the record payload. Its closed schema emits exactly one permitted record key plus a short selection basis. For example, an index row may describe an opaque key as "native evaluator/rubric receipt and recorded outcome" for a truncated task hint. The selector never receives the answer label or correct action, but these metadata make the 24-episode natural routing problem generous. Key accuracy is computed locally after the response; the successor may still defer after a valid record is released.

Canonical archive and proof. Records are serialized with sorted keys, normalized UTF-8, and fixed separators. Leaf hashes commit to record type, source key, and payload. Leaves are sorted by canonical key before pairwise Merkle aggregation, making the root independent of acquisition order. The proof package contains the canonical record, leaf key, sibling hashes, and side bits. Verification recomputes the path to the frozen root before prompt construction. Altered or malformed packages are rejected locally. For the natural-episode packages, SHA-256 hashes compact JSON with sorted keys and Unicode encoded directly as UTF-8. Record IDs are included within the serialized record. Distinct leaf and internal-node tags, each terminated by a null byte, separate the hash domains; an odd final node is duplicated at each level. Materialization requires exactly one matching record ID.

The frozen root is bound into each panel manifest and authorization hash before scored calls. This binds the experimental snapshot into the audit chain; a deployed system would additionally need an authenticated root channel and an append or update policy. The present contract treats a wrong-but-valid committed record as a selector/relevance error, not an integrity failure.

C.2 Order panels

The four-source bridge uses real test GitLab commit packets acquired before the panel was frozen. Two writer chains process the identical set in chronological and reverse-chronological order. Twelve opaque successor claims probe title, author, date, and parent-related facts. The final capsule and archive root are identical across order lanes; only the sequential semantic states differ.

The randomized confirmation uses synthetic Shopping Admin and Salesforce snapshots. A discovery run creates multiple acquisition permutations and identifies fact groups with differing lane coverage. Eleven groups and their higher-coverage lanes are then fixed before fresh opaque replica claims are evaluated; "direction" therefore means discovery-lane direction, not recency or primacy. The successor schedule interleaves conditions randomly. Stage A tests whether fresh aliases preserve those fixed-memory contrasts; Stage B runs selectors and recovery after that gate passes. All 243 planned calls complete with zero retries. No new writers are sampled in the confirmation.

C.3 LOAD AND CAPACITY PANELS

The endogenous crossover projects 28 heterogeneous packets—12 GitHub, eight commerce, and eight CRM packets—into equal-size eight-record cards. Each card has one queryable record and seven same-domain distractors, so the high-load writer faces 224 candidate bindings under a 1,900-character bound. Entity and value strings are replaced by deterministic random aliases. Seven low-load writers each see four cards; a claim is routed by its packet shard to the corresponding memory. One high-load writer sees all 28. The 24 claims are sampled before any writer call, balanced by domain and entailment status, and never sent to writers. Because exhaustive retention is infeasible by construction, this panel calibrates coverage loss and repair rather than discovering a capacity limit.

The fresh capacity panel regenerates aliases and writers. At a 1,900-character budget it compares loads 4, 8, 16, and 28 with 12 claims per condition. The budget sensitivity arm fixes load 28 and compares 950, 1,900, and 3,800 characters. These conditions are analyzed separately: the load contrast is confirmatory; the three-point budget arm is descriptive.

C.4 NATURAL RECORDED-SESSION PANEL

The panel contains 12 GitLab, six Shopping Admin, and six Salesforce tasks. A fixed serializer creates one request record, one record for every retained trajectory step, and one native evaluator-receipt record; cleaning removes secrets and normalizes browser state without using a successor response. The resulting archive contains 117 records, and record IDs, index rows, and the root are frozen before successor calls. One fresh high-load writer receives all 24 bundles.

BM25 uses the standard probabilistic relevance implementation (Robertson & Zaragoza, 2009) with fixed tokenization and returns the top three payload records without semantic memory. Recovery receives semantic memory and uses one schema-constrained selector over the metadata-rich index plus one locally verified evaluator receipt. The same 24 successor instructions and scoring functions are used, but the supplied information is not matched across routes. No online environment is accessed or modified during evaluation. A separate development run over the same episode family yields 22/24 for semantic memory, target-aware retrieval, verified recovery, and full-session context; it is reported as writer-realization sensitivity rather than pooled with the primary panel.

C.5 REPEATED-WRITER MATCHED-PATH CONFIRMATION

Twelve tasks are selected by a fixed SHA-256 ranking before new calls, four per domain. For each of five replicates, one writer sees all 24 episodes in a newly shuffled order and three writers each see one fixed eight-episode bundle. All five replicate IDs and all 12 tasks are retained regardless of score. The 360 successor calls are randomly interleaved across six conditions. Metadata BM25 ranks the same commitment-index rows visible to the learned selector and is always paired with the high-load semantic memory. The routed-trace route is given the top three trace steps from the correct episode after every native evaluator receipt has been removed. The two exact-record routes use the same frozen selector record; one labels it unverified, while the other releases the corresponding proof package only after local Merkle verification. The clean verified and unverified routes use independent single-shot decoding; their difference does not establish a systematic benefit from proof metadata.

The frozen selector is exact for all 12 tasks and its clean packages verify before successor calls. Consequently, this matched-path subset estimates utility conditional on correct record selection. It does not estimate learned retrieval accuracy. No scientific gate controls inclusion; the run completes 380/380 calls with 20/20 valid writers, zero retries, and all five realization-level gains reported.

Before the scale extension makes any model call, a deterministic ranking selects the twelve previously unevaluated tasks from the same frozen 24-task corpus. The extension reuses the five already frozen high-load memories, exact selector record IDs, and locally verified clean proof packages. It executes only the memory-only and verified-exact successor routes, producing 120 new calls. The protocol prospectively specifies the combined analysis over all 24 tasks and all five writers, including a task-level sign test and two-way task–writer resampling sensitivity. The extension completes 120/120 calls with zero retries; all tasks are remain in the analysis.

C.6 CAPACITY-SUFFICIENT STRUCTURED ORDER CONTROL

The structured writer uses a common record schema with source ID, collection, entity ID, and a canonical compact JSON value. Shopping and Salesforce quartets each contain four 24-record packets with 96 unique keys in total. Three fresh writer pairs per domain receive the same packets in ascending and descending source order. Each sequential response must contain exactly 24, 48, 72, or 96 records, and the local evaluator compares the entire keyed map against the cumulative canonical union. The full states occupy 25,892 and 26,242 bytes, below the 65,536-byte limit. The outcome is exact state equality; no successor claim or sealed answer is involved.

C.7 LAYER-STRESS BOUNDARY PANELS

Addressability and wrong-but-valid evidence. The selector-risk panel holds the 12 natural tasks and five high-load memories fixed. It compares three index surfaces: the original 24-episode/117-record metadata-rich index; a 120-episode/213-record index with four deterministic, target-derived routing keys per episode; and the same larger index with task hints and routing keys removed. The selector emits one record ID before local scoring. Successors then receive the selected verified record, an exact-record ceiling, or a prospectively fixed same-domain receipt from the wrong episode that still verifies against the frozen root. All 36 selector and 300 successor cells are retained.

Local open-weight model. Qwen3-8B-MLX-4bit runs locally in non-thinking mode over compacted versions of the same 12-task surface. An exact-routed arm gives five target-blind writers the 1,900-character and 2,048-byte JSON contract; malformed outputs are retained as empty memory and exact verified records are supplied to the recovery successor. A stricter full-stack arm separates memory payload from serialization: the system retains at most 1,900 characters and 2,048 UTF-8 bytes of each raw writer output, then the same Qwen checkpoint selects one record ID from the 120-episode/213-record sparse-key index. A selected record is released only if its proof verifies; wrong-but-valid records are not corrected, and invalid selections release no evidence. Five writer, 12 selector, and 120 paired successor cells give 137 generations, all retained with no network access or retries. This boundary evaluates the evidence-release gate with a prompted successor: even after an invalid key, a no-record successor call is retained and scored. It therefore does not enforce the contract's hard-defer decision branch. For this full-stack arm, writer and successor decoding use temperature 0.7, top-p 0.8, and top-k 20, with generation caps of 768 and 384 tokens, respectively. The selector uses greedy decoding with a 64-token cap; thinking is disabled throughout.

Mem0 real-system writer. Mem0 2.0.19 ingests 24 natural episodes through its released add/search API, using local Qwen3-8B-MLX-4bit extraction and BAAI/bge-small-en-v1.5 embeddings in embedded Qdrant. For each of 12 tasks, the successor receives either Mem0 top-3 or the same retrieval plus the independently verified exact-record package. The registered score requires the frozen exact target terms. The post-hoc outcome-fidelity audit reads only public native evaluator fields: a successful episode passes under the registered exact score; a failed episode passes only if the successor defers or explicitly states a negative outcome. The audit performs no model call and does not replace the registered score.

C.8 POISONED-BINDING INTERVENTION

For each claim, the exact value in its support record is replaced by a different opaque token with the same grammar. All other fields remain unchanged. The altered package is constructed after the clean archive root is committed, so its Merkle path must fail. In the unverified arm, only the altered record is supplied. In the verified arm, verification precedes prompt construction; a failed package results in a local defer event and no successor model call over that record. This is a deterministic post-commitment control and does not model write-time poisoning.

C.9 STATISTICAL ANALYSIS

Paired successor outcomes are binary for exact correctness. For two routes A and B, we report discordant counts (n_{10}, n_{01}) and the exact two-sided McNemar probability under $\mathrm{Binomial}(n_{10} + n_{01}, 1/2)$. The randomized order stress uses a one-sided sign test over 11 discovery-defined fact-

Table 7: Conceptual comparison of components and handoff contracts. Checks indicate compatibility or evaluated coverage, not new primitives; order/load includes the distinct questions identified in the final column. These systems are not experimental baselines.

System or family	Adapt. memory	Select. source	Order/load	Local integrity	Distinction
A-Mem / Agentic Memory	✓	✓	—	—	Optimizes continuing-agent state
ReadAgent	✓	✓	—	—	Revisits source passages
Set-Based Prompting	—	—	✓	—	Inference-time permutation invariance
LLatrieval / Provenance	—	✓	—	—	Model/NLI support judgments
V3DB	—	✓	—	✓	Proves committed search execution
SELECTIVE CONFLUENCE	✓	✓	✓	✓	Bounded successor handoff contract

group directions. This test belongs only to the Appendix C order-panel development diagnostic and is not used for the main repeated-writer estimate. For the repeated-writer panel, the primary stability summary is the complete vector of five paired 24-task gains, its mean and range, and the number positive. The combined analysis additionally uses the 24 observed task clusters for a two-sided sign test and resamples task and writer clusters with replacement; crossed cells are not treated as 120 independent writers. Pooled-cell McNemar values use an independent-pair reference and serve as descriptive contrasts because tasks and writers recur across cells; task and writer summaries carry the main stability interpretation. We report exact denominators and do not impute missing responses. Communication comparisons use canonical JSON, UTF-8 byte counts, and zlib level 9 for model-visible episode context; selector calls and persistent storage are excluded.

D ADDITIONAL RESULTS

D.1 MECHANISM DECOMPOSITION

The direct mechanism panel contains 27 claims and four conditions. Canonical semantic memory is correct on 11; commitment metadata without the exact record is correct on 3; the unverified clean exact record and verified clean recovery are each correct on all 27. Verified recovery beats metadata-only by 24 paired decisions and loses none ($p = 1.19 \times 10^{-7}$). Clean proofs verify 27/27, and all 27 single-field post-commitment alteration controls fail locally. The clean exact-record routes have zero prediction disagreements, showing that proof metadata controls release rather than changing clean semantic inference.

D.2 COMMUNICATION ACCOUNTING

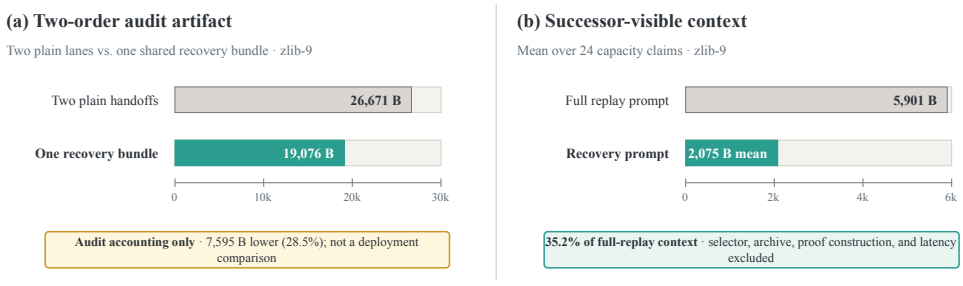


Figure 3: **Serialized context accounting under a common lossless codec.** *Left:* 26,671 bytes is the sum of two plain order lanes, while 19,076 bytes is one shared proof-recovery audit bundle; this is an artifact-accounting comparison, not a deployment saving. *Right:* recovery exposes 2,075 bytes to the successor versus 5,901 for full replay, with both at 24/24. Persistent archive/index storage, selector tokens and call, proof construction, and latency are excluded.

D.3 CAPACITY AND MEMORY-BUDGET SENSITIVITY

At fixed budget, the fresh capacity sweep yields exact coverage and successor correctness of 12/12 at loads 4, 8, and 16, then 2/12 at load 28. At fixed load 28, budgets of 950, 1,900, and 3,800 characters yield 1/12, 2/12, and 3/12. Since the high-load construction cannot retain all 224 candidate bindings as literal text, these values calibrate which queried bindings survive for this writer rather than establish a general capacity result.

D.4 WRITER-REALIZATION SENSITIVITY

A separate complete 123-call panel over the recorded-session family produced 22/24 correct outcomes for semantic memory, target-aware retrieval, verified recovery, and full-session context. It passed every writer, transport, proof, and budget gate and is retained as a valid null recovery-gain result. The registered prediction that recovery would outperform semantic memory was a scientific hypothesis, not a condition for panel admission. This panel used three target-blind memories covering eight episodes each; the 7/14/21 diagnostic used one memory covering all 24 episodes. We therefore do not describe their difference as writer randomness alone or pool them as exchangeable samples. Together they motivate a load/coverage crossover test with repeated writers and outcome-independent inclusion.

D.5 REPEATED-WRITER AND MATCHED-PATH CONFIRMATION

The confirmatory panel completes all 380 planned calls at an estimated API cost of $8.67. All 20 writer outputs satisfy both the 1,900-character and 2,048-byte contracts. Per-writer high-load versus verified-recovery scores are $(3, 11)$, $(2, 11)$, $(2, 11)$, $(2, 10)$, and $(4, 11)$ out of 12. The corresponding gain vector is $(8, 9, 9, 8, 7)$; no zero or reverse realization is excluded.

The prospectively frozen complement completes 120/120 successor calls at an estimated cost of $1.80. Combining both halves yields 24 tasks and five writer states: high-load memory resolves 21/120 paired cells and verified recovery resolves 109/120. Writer-level gains are $(20, 18, 14, 18, 18)$ out of 24. At the task level, 23 directions are positive and one is negative (two-sided sign test $p = 2.98 \times 10^{-6}$). A fixed-seed two-way resampling over task and writer clusters gives a 95% percentile interval $[0.558, 0.892]$ for the mean paired gain. The corpus-native domain split is 12 GitLab, six Shopping Admin, and six Salesforce tasks; recovery scores 60/60, 30/30, and 19/30, respectively, with zero wrong actions.

On the original 12-task matched subset, domain-stratified verified recovery is 20/20 for GitLab, 20/20 for Shopping Admin, and 14/20 for Salesforce, with the remaining six Salesforce decisions deferring and no wrong action. The receipt-free routed-trace route is 20/20, 20/20, and 12/20, respectively, but makes three wrong Salesforce actions. The metadata-BM25 route is 11/20, 8/20, and 11/20 across the same domains and makes three wrong Salesforce actions. These splits show where the remaining uncertainty lies without changing the global scoring rule.

Salesforce failure audit. All six verified-route deferrals have explicit negative-outcome receipts: five are repetitions of one failed episode and one is a repetition of another. The successor summarizes the recorded failure correctly but maps that supported negative answer to DEFER; these are decision-policy errors rather than missing evidence. All three routed-trace wrong actions are repetitions of one episode whose actor trace carries an incorrect target-field interpretation. The native evaluator receipt contains the corrected target but is excluded by this route, so these errors arise from non-authoritative trace content rather than an absent episode. This distinction explains why exact recovery fails safely on the former cases while receipt-free trace can commit incorrectly on the latter.

Clean exact-record delivery is 53/60 unverified and 54/60 after verification. There are three verified-only and two unverified-only correct pairs, consistent with independent single-shot successor variation rather than a systematic clean-proof benefit. The proof contract is evaluated by its release semantics and altered-record control, not by this one-decision difference.

D.6 Claim addressability and wrong-but-valid evidence

All 336 planned calls complete with zero retries at an estimated cost of $6.20. Full task hints and sparse routing keys each select the exact record 12/12 times in the 24- and 120-episode indices, respectively; the hint-free 120-episode index selects it 1/12 times. End-to-end correct/defer/wrong counts across five memories are 54/6/0 for full hints, 55/5/0 for sparse keys, and 17/43/0 for hint-free routing. Sparse keys beat hint-free routing in 38 paired cells and lose none (exact two-sided McNemar $p = 7.28 \times 10^{-12}$).

The forced wrong-but-valid route yields 12/60 correct, 48/60 defer, and 0/60 wrong actionable decisions. All 12 correct cells already contain the expected term in semantic memory; the wrong receipt does not supply the answer. Four of 300 successor responses cite a record ID not visible in their condition, so correctness and evidence-citation compliance are reported separately. These results support claim addressability as a necessary contract layer, not the proposition that irrelevant authenticated evidence is harmless.

D.7 Open-weight and real-memory boundaries

The exact-routed local Qwen3-8B ablation completes 125/125 generations without network access, runtime errors, or retries. Two of five writer outputs satisfy both size and JSON constraints; three exhaust the generation cap before producing valid JSON and are retained as empty memories. Memory-only correct/defer/schema-invalid counts are 0/49/11. Verified recovery produces 41 correct, nine defers, and ten wrong actions; every writer's paired gain is positive at $(9, 7, 9, 8, 8)$. By domain, recovery is 19/20, 17/20, and 5/20 correct for GitLab, Shopping Admin, and Salesforce. This conditions on exact evidence and demonstrates that proof delivery alone does not repair weak serialization or successor interpretation.

The stricter open-weight full stack completes 137/137 local generations with five valid bounded memories and no runtime errors or retries. The five seeds produce two unique byte-level memory states. Its model selector over 120 episodes is exact for 3/12 tasks, wrong-but-valid for five, and invalid for four. Memory-only correct/defer/schema-invalid counts are 0/30/30; model-routed verified recovery produces 15 correct, 35 defers, 10 schema-invalid outputs, and zero wrong actions. Every writer gains exactly three correct decisions. Conditional on exact selection, recovery resolves 10/15; under wrong-but-valid selection it resolves 5/25, but all five concern one task already covered by memory. This end-to-end result transfers the positive direction while identifying addressability as the limiting layer.

Mem0 extracts 24 memories from 24 episodes. Support hit@3 is 4/4 for GitLab, 3/4 for Shopping Admin, and 3/4 for Salesforce. Registered correct/defer/wrong counts are 8/2/2 for both Mem0 top-3 and Mem0 plus verified exact recovery; the two discordant tasks cancel (exact McNemar $p = 1$). One verified receipt repairs the Shopping `Processing` target, while a failed Salesforce receipt changes a target-term-matching but false completion into defer. A second failed Salesforce receipt produces the same safety correction without changing registered correctness.

The deterministic post-hoc audit contains ten natively successful and two natively failed targets. On successful targets, registered correctness changes from 7/10 to 8/10. On failed targets, false completion changes from 2/2 to 0/2 and safe defer from 0/2 to 2/2. The combined outcome-fidelity diagnostic is therefore 7/12 versus 10/12. Because this criterion was defined after inspecting the registered-score mismatch, it is reported only as construct-validity evidence.

D.8 Structured capacity control

All 48 sequential updates match the exact cumulative key–value union. Both final lanes in each of six pairs preserve 96/96 records, producing 12 exact final states and six canonically equal opposite-order pairs. There are no order-asymmetric missing keys, duplicate keys, or value changes. The run completes 48/48 zero-retry calls for an estimated $4.37. This negative control narrows the paper's order claim: sufficient structured state prevents the observed failure; selective recovery addresses settings in which that guarantee is unavailable.

Table 8: Secondary robustness and sensitivity evidence. These rows are not pooled with the complete primary panels.

Evidence	Result	Reporting boundary
High-load memory budget	950/1900/3800 chars: 1/12, 2/12, 3/12	Descriptive capacity calibration; no dose-response law
Eight-episode memory construction	All four routes: 22/24	Valid null recovery gain; differs from the 24-episode construction in load as well as realization
Second-model capacity	Load 16: 12/12; load 28: 2/12	Complete replication of the stress direction
Second-model natural arm	Stopped before the frozen panel completed	Informatively censored; retained as execution audit only

D.9 Second-model evidence

The secondary run uses the provider-reported `gpt-5.5-2026-04-23` identifier with the same target-blind contract. The capacity arm is complete. The natural schedule stopped when response creation timed out before an output or usage record existed. The zero-retry policy prevented later cells from running, so the observed prefix is informatively censored. We preserve its receipts for audit but do not score or interpret that prefix as a cross-model effect.

E Artifact Integrity and Independent Audit

Before execution, content hashes bind the public panel, sealed oracle, schedule, schemas, model configuration, call budget, statistical gates, and exclusion rules. Each scheduled model cell has one prompt, response, event stream, usage receipt, and local score; SDK, application, and task retries are zero. Independent reconstruction checks the archive root, canonical record, and proof path before release. Rejected packages never reach the model.

A separate audit program replays immutable records to recompute hashes, schedule uniqueness, blindness constraints, outcome counts, exact tests, common-codec bytes, budgets, and secret-pattern checks. Each panel applies its frozen failure policy. Fatal artifact or transport failures seal a run where specified; planned proof rejections remain scored controls, and invalid or missing outputs follow the stated retention rules without replacement. Partial runs do not enter complete-panel estimates.

Headline estimates use complete, independently audited primary panels. Development runs, including valid null results, remain available separately. The repeated-writer audit recomputes 380 call receipts, 20 writer contracts, 360 successor rows, six condition totals, and five writer gains. The structured-order audit checks 48 cumulative maps, 12 final lanes, and six confluence comparisons. The addressability audit checks 36 selectors, 300 successor cells, all 300 proof paths, and three index surfaces. The exact-routed open-weight audit checks 125 local generations, five writer states, paired seeds, and exact packages. The Mem0 audit checks 25 local calls, retrievals, 24 successors, and the registered score; a separate zero-model audit computes post-hoc outcome fidelity. The second-model capacity arm is complete and audited; its censored natural prefix is retained only as an execution record. Prompts and artifacts contain no credentials, cookies, OAuth tokens, private keys, or authentication state.

F Main-Claim Traceability

Table 3 maps claims to saved panels, not an additional meta-analysis. Each row retains its own task family, outcome definition, and communication boundary.