NO LONGER CURRENT, STILL BINDING: DIAGNOSING WHERE AGENT MEMORY FAILS TO MAINTAIN DECISION CONSTRAINTS

Anonymous authors
Paper under double-blind review

ABSTRACT

An event can end while the constraint it creates remains in force. The standard long-horizon benchmarks do not ask whether a memory system handles this: across LoCoMo and LongMemEval, 0 of 2,486 questions require applying a resolved event as a constraint to a decision the question does not name. We introduce CCD, a controlled diagnostic evaluation of whether agent memory retains constraints after events end, releases them when lifted, and rejects a release record that belongs to another patient. It comprises 53 synthetic Chinese clinical cases in four variants that differ in what the follow-up records; 44 pass a rule-based gold audit for release decisions. On 43 eligible cases, with the question, answerer, prompt and retrieval budget fixed, we vary only the supplied follow-up record. The natural context releases 17; supplied as the sole follow-up record, the patient's own lifting record releases 32, its resolution record 8. Another patient's lifting record falsely releases 30 of 43 still-restricted cases. Both contrasts hold on a second backbone. On a natively run production memory system, term-matched binding evidence reaches the answerer in 11 of 53 corpora. An oracle-guided maintenance pass over a frozen store then removes evidence that is still required, a further loss caused by the imposed policy. Release requires lifting evidence that is retrieved and attributed to the correct case.

1 INTRODUCTION

An assistant that has followed someone for months acts on a history that keeps being updated. Some updates report that an episode has ended, while others confirm that a restriction arising from it has been lifted. For example, recovery from myopathy does not by itself lift a restriction on restarting the drug that caused it. Reading the first as the second permits an action that is still restricted; missing the second keeps enforcing a restriction that has already been lifted. The two errors point in opposite directions, and a system can commit either one while answering correctly about the patient's present state.

Figure 1 illustrates the distinction with a suspected peanut allergy. A person developed a rash after eating peanuts and was advised to avoid them pending allergy evaluation. The rash later disappeared, but the advice to avoid peanuts remained in effect. An assistant asked for snack suggestions should therefore still exclude peanuts. Now consider an alternative follow-up stating that an allergy specialist ruled out peanut allergy and lifted the restriction. The earlier reaction still occurred, but it no longer justifies excluding peanuts. Keeping the original event fixed and changing the follow-up lets us test whether the assistant retains a restriction that still applies and stops applying one that has been lifted.

Two lines of work bear on this. One maintains the validity of what is stored. TEPA (Zhou et al., 2026) revokes a precedent when fresher evidence contradicts it under the same key; related systems down-weight or exclude entries that a later record supersedes (Wang, 2026); and a benchmark scores both detecting an invalidated state and acting on the update downstream (Chao et al., 2026). The question they answer is which stored content should still count as current, and by their criterion an event that simply ran its course is not a contradicted one. The other line asks whether a stored constraint is applied when the follow-up question does not name it. LoCoMo-Plus (Li et al., 2026)

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.

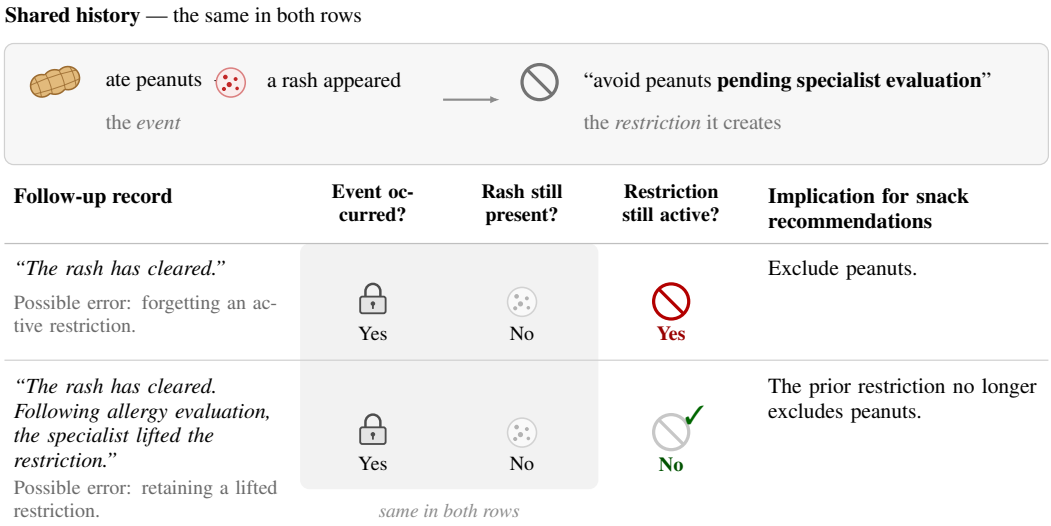

Figure 1: Recovery alone does not release a restriction; an explicit lifting does (illustrative example). In both histories the person developed a rash after eating peanuts and later recovered. The advice to avoid peanuts remains active in the upper row, but is explicitly lifted after specialist evaluation in the lower row. An assistant suggesting snacks should exclude peanuts while the restriction remains active, and should stop treating the earlier reaction as a reason to exclude them once it is lifted.

is the nearest, and poses exactly that retrieval problem, but its released items concern constraints whose source state has not ended, and no lifecycle policy runs over its store. Neither line tests what happens to a constraint as its source event changes state: when the event resolves, when a record formally lifts the constraint, and when the lifting belongs to a different patient.

Whether an event has ended does not settle whether it still constrains a particular decision, and an event-status label records only the former. Suppressing records solely because an event has ended can therefore remove evidence that later decisions still need. None of the designs whose code or mechanism description we could read explicitly labels records whose source event has ended but whose constraint remains in force.

We audited all 2,486 questions across LoCoMo (Maharana et al., 2024) and LongMemEval (Wu et al., 2025) under the protocol of Appendix D. None requires applying a constraint from a resolved event without an explicit cue to that event in the question. Yet 8 of the 10 conversations in the released LoCoMo corpus contain resolved events with continuing constraints. Although 38 questions refer to these events, none tests whether their constraints still guide a later decision. High scores on these benchmarks therefore do not establish that a system can apply a constraint after its source event has ended.

Accuracy alone is insufficient for two reasons. First, evaluating only cases where a restriction remains active can reward an assistant that always refuses the requested action. Paired evaluation requires the assistant to retain the restriction while it applies and stop applying it after an explicit release. Second, the same error can arise at different stages. An assistant may continue enforcing a lifted restriction because the release record was never stored, because a recovery record was retrieved in its place, because the update was assigned to the wrong patient or event, or because the record was available but not used correctly. We combine memory-pipeline tracing with controlled evidence substitution to separate these failures.

We introduce CCD, a controlled diagnostic evaluation of constraint persistence, release, and update attribution in agent memory. Paired histories keep the index event unchanged while varying whether it remains active, resolves, or has its restriction lifted. An additional variant replaces the target event with a benign episode, to test whether the assistant imposes a restriction that no record supports. Evidence substitution holds the question, answerer, prompt, and retrieval budget fixed while changing the follow-up record supplied to the answerer. The answerer is `gemini-2.5-flash`; the cross-backbone checks use `claude-sonnet-5` (subsection 5.3) and `gemma-4-31b-it` (sub-

section 5.1). We also trace evidence through a production memory pipeline and apply maintenance policies to frozen store snapshots to measure the additional evidence loss caused by those policies.

Our contributions are:

- **Task and diagnostic evaluation.** We distinguish event status from continuing constraints and introduce CCD to test constraint persistence, release, and update attribution. An audit of 2,486 questions across LoCoMo and LongMemEval finds none testing this implicit use of resolved events (section 2).
- **Empirical diagnosis and strategy comparison.** We compare memory maintenance and retrieval strategies, trace evidence loss, and isolate the effects of supplied updates. Retrieval crowding accounts for most of the term-matched evidence loss in the evaluated system. Supplying the case's own release record raises correct release to $32/43$, but another patient's release record falsely releases $30/43$ still-restricted cases (section 4).
- **Supporting analysis: rank-based suppression.** We derive the condition under which query-independent suppression can clear the event while retaining the evidence other queries need, and find it unmet in 21 of the 26 reachable cases; the same form estimates eviction thresholds from retrieval logs (section 4).

2 Problem Formulation

Three predicates. For a memory m recording a clinical event and a decision-bearing query q at time t, we distinguish:

$$\textsc{Occurrence}(m) \in \{0,1\} \quad \text{did the event ever happen (immutable)} \tag{1}$$

$$\textsc{Current}(m,t) \in \{0,1\} \quad \text{is the state still active at } t \tag{2}$$

$$\textsc{Binding}(m,q,t) \in \{0,1\} \quad \text{does it constrain the decision behind } q \tag{3}$$

Occurrence is unchanged by resolution: an event that has ended still happened. Current can change over time. Binding also depends on the query, and can remain 1 after Current becomes 0. A policy that suppresses resolved records without preserving their decision-relevant evidence implicitly assumes that only current events can constrain a decision, that is Binding $\Rightarrow$ Current. The target cell is Occurrence$=1 \wedge$ Current$=0 \wedge$ Binding$=1$, which we write C0B1.

We diagnose failures along three dimensions: supply, whether the required evidence reaches the answerer; attribution, whether an update applies to the correct person, event and action scope; and use, whether the answerer then applies or releases the constraint correctly. These dimensions can overlap. We examine them through ingestion-log tracing and controlled evidence substitution.

Evaluation criteria. Five quantities recur. Resolution retrieval is scored against the sentence asserting the state ended, and lifting retrieval against the sentence recording the release; neither is scored on any gold unit that happens to be retrieved.

Quantity	Counted when	Valid denominator
Target-history retrieval	the index-event unit is in the top-k	53
Resolution-evidence retrieval	explicit resolution evidence is in the top-k	50
Lifting-evidence retrieval	explicit release evidence is in the top-k	53
Decision correctness	the answer applies or releases as gold requires	44 on release
Paired-joint correctness	one case is correct on *both* Resolved and Released	44

Two denominators for the same variant. Every Released corpus carries a lifting sentence, so lifting-evidence retrieval is scorable on all 53. Release *decisions* additionally need a usable binary gold label, and 9 corpora carry a lifting that contradicts the relation's own stated basis, which declares the constraint permanent; release and paired-joint therefore use 44 (audit and repair in subsection D.2). Three Resolved follow-ups do not state that the index event resolved; they are dropped from the resolution row (hence its 50) and kept in the target-history row's 53 and in the Resolved decision denominators. Every evaluation subset used later is indexed in Appendix B.

Suppression as rank translation. Let a retriever produce a total order over N memory units for query q. A lifecycle mechanism marks a subset $R \subseteq [N]$ as resolved and applies an integer shift $s \in \mathbb{Z}_{\geq 0}$:

$$\text{key}_q(i) \;=\; \text{rank}_q(i) \;+\; s \cdot \mathbf{1}[i \in R], \qquad \text{re-sort by key}_q, \text{ keep the top } k. \tag{4}$$

The base order is whatever the stack already uses (BM25 here, a dense encoder on the production stack), and Equation 4 models suppression applied to that base ranking. The base ranking depends on q, while the same suppression strength s is used across queries. Ties preserve the base order. Adding s to a sort key moves a memory by a data-dependent distance: how far it falls depends on how many of its neighbours were marked too; Corollary 1 turns this quantity into an eviction threshold. The model spans no suppression (s=0), finite rank penalties, and deletion-equivalent top-k retrieval ($s \to \infty$, identical whenever k unsuppressed units remain).

Evidence retention and removal. The task requires event evidence to remain retrievable under Q_{bind} and resolution evidence under Q_{state}. When both are initially retrievable, s=0 satisfies these requirements. Suppression adds a goal of the policy, not of the task: removing the resolved event from the top-k results for Q_{state}. Retrieving the event alongside its resolution can still answer the state correctly; what subsection 4.2 prices is whether a policy pursuing this removal can do so without evicting evidence the task requires.

3 The CCD Diagnostic Set and Experiments

Controlled evaluation. Aggregate scores cannot separate missing evidence from wrong attribution, or either from failure to use what was retrieved. CCD uses paired histories, controlled retrieval experiments and evidence substitution to examine these sources separately.

Construction. CCD contains 53 Chinese clinical cases over 14 relations and a 40-case English DevOps extension over 10, all generated with `gemini-2.5-flash`; the release-axis gold passed the rule-based consistency audit of subsection D.2. Six of the fourteen clinical relations are held out by design, spanning an oncology domain absent from the rest. Each case is a shared history, an index event and a follow-up session. Four variants separate an event ending from its constraint being lifted. ONGOING: the state has not ended, so the constraint remains active. RESOLVED: the state has ended and the constraint it left has not been lifted, the target cell. RELEASED: it has been lifted, and the correct answer flips to *release* (the artefacts name it `DISCHARGED`). NEVER: the event is absent from the constructed history, which tests whether the model imposes the restriction with no record of the event. Three queries ask about the current state (Q_{state}), a decision the history bears on (Q_{bind}), and something unrelated (Q_{irr}).

ONGOING, RESOLVED and RELEASED share one index-event session: we substitute the follow-up sentence and rewrite that session's units to agree. Across the 53 clinical cases the index-event text is unchanged in all 106 ONGOING–RESOLVED and RESOLVED–RELEASED pairs. NEVER replaces index event and follow-up wholesale, so no record asserts or denies the event.

Maintenance and retrieval arms. We compare mechanisms on a shared retrieval pipeline with a default budget of k=20 units. The maintenance policies are append-only retention, rank-based suppression across strengths under broad and narrow label scopes (its $s \to \infty$ endpoint is deletion-equivalent), and type-based protection from suppression. The retrieval strategies are a query-conditioned slot channel, one-hop event-to-outcome expansion, and RETRIEVE-MORE, which suppresses nothing at $3k$ budget. The suppressor receives *oracle* resolution labels, isolating the policy's effect from errors in detecting resolution.

Evidence substitution and store tracing. Except in the natural-context arm, substitution removes every outcome unit from the context and inserts one designated record at a fixed position, with question, answerer, prompt and budget held fixed: the case's own lifting sentence, its resolution sentence, another patient's lifting sentence, or nothing. Further conditions add subject, event and action-scope labels to every unit (subsection 5.3).

On a production stack we trace where the evidence goes with `mem0` running natively, under its own extraction and conflict resolution, then apply deletion policies to frozen snapshots of that same store to measure the additional loss each causes (subsection 4.1).

Scoring. We report the retrieval, decision and paired-joint metrics of section 2. Resolution and lifting retrieval are scored on the sentence reporting the state ended and the sentence recording the release. Eligibility is assessed per metric and every result carries its denominator. Decision correctness is scored by deterministic matching on the answer's conclusion.

4 DIAGNOSTIC RESULTS

4.1 WHERE THE EVIDENCE IS LOST: THE NATIVE RUN, THEN THE SAME STORE FORKED

We trace losses in a native run, then measure an imposed policy on frozen stores.

(A) Running natively, where does the evidence go? On `mem0`, running with its own extraction and conflict resolution and no oracle labels (116 corpora: the 53 RESOLVED and 53 ONGOING arms plus 10 NEVER; $\sim$250k memory operations), the store snapshots and ingestion log decompose the 53 corpora: event evidence is term-matched in 42 final stores, 3 wrote it before the maintenance pass deleted it, and 8 wrote it in words our term-matching scorer cannot see. No store failed to write event-session content. Of the 42, 11 reach the top-20 and 31 are crowded out: $11/53$ is what this scorer observes. The 31-to-3 split attributes most of the loss visible to this scorer to retrieval crowding rather than native maintenance; consistently, the reachable count is the same under ONGOING and RESOLVED (paired discordance $2/2$, p=1.0).

(B) Holding the store fixed, what does a maintenance pass add? The log locates losses but does not isolate the effect of an imposed maintenance policy, so we freeze each post-ingestion store and fork only the policy over the identical snapshot, replaying retrieval with the stack's own embedder, validated byte-exact against the native run's recorded top-20 on all 159 rows (53 RESOLVED corpora $\times$ three queries). Broad deletion then takes binding evidence from $11/53$ to 0 and resolution-record coverage from 31 to 0 of the 53 stores (coverage counts all 53; the three corpora without an explicit resolution sentence can only score as absent, so at most 50 could contribute); narrow deletion keeps the resolution record and still collapses binding to $1/53$.

The *oracle* policy of (B) is not `mem0`'s native policy, whose conflict machinery shows no detectable separation of resolved from ongoing event content (UPDATE/DELETE volume 28 vs. 19, p=0.24). Broad deletion causes a measured additional loss in all 11 corpora whose binding evidence was retrievable from the frozen stores; this is also the most this experiment can lose, since the other 42 were outside the retrieved top-20 before any forked policy ran, for the reasons (A) separates.

The state-axis harm reaches the answer layer. A separate answer-layer test applies the resolved suppressor to ONGOING with the resolved label forced onto units whose state has not ended (the corpus is unchanged; only the label differs from the oracle run), emptying the follow-up state evidence ($16 \rightarrow 0$ of the 38 in-vocabulary cases; subsets are indexed in Appendix B) and moving the answers ($24/38 \rightarrow 14/38$; discordant $12/2$, McNemar p=$\mathbf{0.0129}$). On RESOLVED the same emptying leaves accuracy unmoved, because gold coincides with the model's default.

4.2 WHAT A SUPPRESSION PASS WOULD COST

A single query-independent shift usually cannot satisfy both retrieval goals. Write $s_{\mathrm{ev,bind}}$, $s_{\mathrm{ev,state}}$ and $s_{\mathrm{res,state}}$ for the smallest shifts that evict the event evidence under Q_{bind}, the event evidence under Q_{state}, and the resolution evidence under Q_{state} ($+\infty$ where eviction is impossible; Appendix E). The first diagnostic, the *retention trade-off*, asks for a shift that clears the event from Q_{state} while keeping it under Q_{bind}: one exists iff $s_{\mathrm{ev,state}} < s_{\mathrm{ev,bind}}$. On the 26 cases where, before any suppression, the index-event unit is in Q_{bind}'s top-k and the resolution unit is in Q_{state}'s top-k, no such shift exists in $\mathbf{21/26}$: the penalty is the same under both queries, so the outcome can differ only through the two rankings, which rarely diverge enough (over all 53 the condition fails in 41, or 48 if the resolution record must also survive, which adds $s < s_{\mathrm{res,state}}$).

A second diagnostic prices the goal of an aggressive suppressor that accepts losing binding retrieval: evict the event while keeping the resolution record under Q_{state}. Its window is empty in $17/26$ of the same cases (Theorem 1, analysed in Appendix E). Reported counts use exact per-case thresholds; the estimate below reads them from two logged statistics. Equation 4 is an analysis model: in the production and published designs we examined (Mem0 (Chhikara et al., 2025), Letta (Packer et al., 2023), LangMem, A-Mem (Xu et al., 2025), MemOS (Li et al., 2025), Memobase, Zep (Rasmussen et al., 2025), TEPA (Zhou et al., 2026), CUPMEM (Chao et al., 2026), DeMem (Zou et al., 2026), RD-Forget (Li & Li, 2026) and MemoryLACE (Yacoubi et al., 2026)), the lifecycle mechanisms we could inspect act at its endpoints, append-only retention or deletion-equivalent removal; we found none operating in the interior.

Whether a memory survives Equation 4 is predictable from a log: its pre-shift rank, k, and the fraction of units above it that were marked. Appendix E derives the eviction condition exactly under Equation 4 and turns it into a threshold *estimate* computed from those logged quantities; against the exact condition the estimate has a median absolute error of half a rank position, and is stable under a dense retriever. Eviction is governed by how many of a unit's higher-ranked neighbours were marked: on the EHR-grounded synthetic-dialogue benchmark MedLoCoMo (Zhang et al., 2026), per-probe redundancy correlates with the eviction threshold at Spearman $+\mathbf{0.465}$, against -0.072 for corpus size.

On a production retrieval stack run locally (LangMem's `InMemoryStore`, oracle labels) the three policy families land on the poles of this trade-off: deletion removes the event and the resolution evidence alike; append-only preserves binding but also retrieves the benign replacement and surfaces the event on irrelevant queries; and overwrite, which replaces on fact identity with no validity check, destroys binding *even when nothing has resolved*. On that stack, applying the penalty in score space rather than rank space behaved as an on/off switch at the one setting we tested: λ:1.0→0.9 moved the median gold rank from 49 to 868 on MedLoCoMo text.

5 PROTECTING BINDING EVIDENCE AND TESTING RELEASE

Increasing suppression strength does not recover the target cell: its count falls $14 \to 12 \to 6 \to 0$ of 38 as suppression strengthens, while ONGOING, none of whose units the oracle labels resolved, never moves. We therefore test two ways to preserve binding evidence: selecting it after seeing the query (subsection 5.1), and exempting binding-typed facts from suppression (subsection 5.2). We then test what evidence is needed to stop applying a constraint once it has been lifted (subsection 5.3).

5.1 A QUERY-CONDITIONED CHANNEL DOES NOT BEAT RETRIEVE-MORE

We implement a slot-reserving retriever keeping a few positions for a query-conditioned binding channel exempt from suppression, motivated by the decision-oriented budget view of Zou et al. (2026) (the slot channel itself is ours). It recovers the target cell's required evidence against its own ablation ($p<0.0001$) without the budget baseline's over-retrieval cost, but on JC_2, joint citation over

Table 1: **Column (1) alone would favour RETRIEVE-MORE; the off-target columns show the cost.** Columns (1)–(3) count the index-event unit, n=53, or in NEVER its benign replacement, whose retrieval is not itself an error; each row retrieves at the budget its *Budget* column states. Column (4) counts explicit resolution evidence, n=50, scored on the unit *asserting the state ended*. *Suppression* is SLOT minus the binding channel; RETRIEVE-MORE is the same skeleton at 3× budget.

Mechanism	(1) Required Q_{bind}/RES ↑	(2) Substitute Q_{bind}/NEV	(3) Irrelevant Q_{irr}/RES ↓	(4) Resolution Q_{state}/RES ↑	Budget
Semantic only	21/53	10/53	18/53	17/50	k
+ expansion, recency	23/53	5/53	13/53	13/50	k
+ **suppression** (*ablation*)	**0/53**	0/53	0/53	**0/50**	k
RETRIEVE-MORE	**49/53**	34/53	48/53	27/50	**3k**
SLOT (broad current)	31/53	2/53	12/53	0/50	k
SLOT (narrow current)	31/53	2/53	12/53	**13/50**	k

Table 2: **Retention and release come apart: no arm here does both reliably.** All cells score decisions: RLS counts answers that stop applying the restriction once it has genuinely been lifted, RES answers that keep applying it while it still holds, and NEV answers that do not invent one that never existed. *Joint* requires both (*deny* on RESOLVED, *proceed* on RELEASED), so a uniformly cautious answerer scores 0 on it. Release and *Joint* use the 44 corpora carrying a usable binary release label; retention and floor use all 53 (section 3). All arms share the candidate pool, answerer and judge; the retrieval budget is k, except RETRIEVE-MORE at $3k$.

Arm	RLS/44 stop applying	RES/53 keep applying	NEV/53 floor	RES/Q_{st} state	Joint/44 both
Append	$6/44$	$52/53$	$2/53$	$16/53$	$6/44$
Lifecycle (broad)	$4/44$	$49/53$	$4/53$	$6/53$	$1/44$
TYPED	$\mathbf{9/44}$	$50/53$	$3/53$	$21/53$	$6/44$
TYPED+fwd	$\mathbf{9/44}$	$50/53$	$4/53$	$\mathbf{31/53}$	$\mathbf{7/44}$
Retrieve-more ($3k$)	$7/44$	$49/53$	$4/53$	$30/53$	$6/44$

a case's two queries (cite the history on the binding query, abstain on the irrelevant one), it does not outperform RETRIEVE-MORE: suppress nothing, retrieve $3\times$ as much. Required-evidence retrieval alone would miss how differently the two arms behave: RETRIEVE-MORE reaches $49/53$ on it, yet takes the benign replacement in $34/53$ against SLOT's $2/53$ (Table 1); restricting the resolved label to the event mention moves the state column from $0/50$ to $13/50$ without changing the other three retrieval counts. The query analyzer is a single point of failure: a missed binding query silently reduces the method to its ablation, and this failure replicates on a second model family. The retrieved sets have little overlap (median Jaccard 0.08), yet the two arms separate little on JC_2.

5.2 CHANGING WHAT IS SUPPRESSED: TYPED EXEMPTION

Typed exemption preserves binding evidence while suppressing transient clutter at a fixed retrieval budget. It exempts binding-typed units from suppression, while *transient* resolved units, the clutter that suppression legitimately targets, remain suppressible. CCD's pool excludes resolved-structured noise, so we inject six such sessions per corpus, oracle-typed, every policy on identical corpora. On LangMem's `InMemoryStore`, the stack of subsection 4.2, over the transient-injected corpora, typed exemption holds binding at append-only's level (29 vs. $28/53$) at the same budget, and clears the transient leakage: append-only lets an injected transient into the top-20 in $30/53$ corpora under Q_{state}, TYPED in none. Neither recency pinning nor salience exemption matches this combination on either stack: the constraining fact is structurally old and need not be salient. The pattern reproduces on the BM25 skeleton under rank translation, including a recency-boosted regime where append-only leaks transients $53/53$.

These retrieval results use oracle type labels; replacing them with a classifier loses 2–3 of 53 target cases. At the answer layer, typed exemption improves over suppression on JC_2 under both providers but does not outperform retrieve-more: no detected difference in Chinese, a loss in English.

5.3 THE RELEASE AXIS: WHICH EVIDENCE, AND WHOSE EVENT

Typed exemption protects binding evidence but does not specify when a constraint should be released; the RELEASED condition tests that separate requirement. Does the answerer distinguish a lifting record from a resolution record? Substituting the record into a fixed context tests this directly (Table 3): removing every follow-up record collapses release to $4/43$ against the natural context's $17/43$, the case's own lifting sentence restores $32/43$, and its resolution sentence reaches $8/43$ (discordant $25/1$, McNemar $p<10^{-4}$).

Does it check whose lifting record it received? A foreign lifting releases as readily on RELEASED ($33/43$) and falsely releases $30/43$ RESOLVED cases. A separate run that labels the owner on every unit re-measures that second cell at $30/43$ and still leaves $25/43$ ($7/2$ discordant against its own baseline, p=0.18): this run establishes no reliable improvement, and most false releases remain. The second backbone shows the same pattern: over the same 43 cases `claude-sonnet-5` likewise separates lifting from resolution, accepts a foreign lifting as readily as the case's own, and again

Table 3: **The supplied follow-up record changes release decisions.** (A) is the natural context; (A_0) removes every follow-up record. Starting from (A_0), conditions (B)–(D) place one record in the same final slot, with question, answerer, prompt and budget held fixed (n=43 of the scorable 44 that also have a donor and a resolution-only unit). †(B) on RESOLVED inserts a lifting that never occurred, a manipulation check rather than a safety cell; (D) is the safety cell there. Contexts are the typed+fwd arm's retrievals under the symmetric answer prompt, so cells pair within this table and row (A) is not Table 2's Append row.

Record placed in the final slot	releases on RLS	retains on RES
(A) nothing removed, natural context	17/43	38/43
(A_0) all follow-up records removed	4/43	39/43
(B) this case's explicit lifting sentence	**32/43**	13/43†
(C) this case's resolution-only sentence	8/43	37/43
(D) same relation, another patient's lifting	33/43	13/43

shows no reliable change from ownership labelling. What makes a foreign record acceptable is not identifiable here: event class and licensed action are coupled one-to-one in CCD, so no condition varies one while holding the other fixed (Appendix A).

Missing lifting evidence is a major bottleneck: with a lifting sentence in the retrieved context a ground-truth release verdict releases and retains in 11/12 cases, without it no tested condition exceeds 3/25 (subsection D.3), and read-time expansion raises lifting recall from 3 to 13 of 53. A query-conditioned release gate tested on the same set establishes no reliable gain over relevance-only selection (subsection D.3).

The NEV floor of Table 2 is low in every arm (2–4 of 53). In sampled answers the refusals cite other records in the history or general caution; none fabricates the absent index event. The rate is also prompt-sensitive: on an empty context, two phrasings of the same question move refusal from 34 to 15 of 53. The floor prices the answerer's default caution, which the paired *Joint* metric is designed to cancel.

Release requires two things the tested memory pipelines do not reliably provide: retrieving the lifting record, and linking it to the correct case.

6 RELATED WORK

Implicit constraints and scope. LoCoMo-Plus (Li et al., 2026) is the nearest evaluation: it scores whether models "retain and apply latent constraints" when the later query is not similar to the record that must be recalled. Its constraints are "user state, goals, or values", it states that it "does not

Table 4: **What the neighbouring evaluations ask.** (1) does any item's gold require *applying* a constraint whose originating state has ended (the C0B1 cell); (2) is the gold a downstream decision rather than the record itself; (3) is the later query semantically dissimilar to the record that must be recalled; (4) is a deletion, suppression or expiry policy the varied factor. ✓/✗ follow the cited work's own abstract or text; – means the work does not state it and we did not audit its items, so it is not a ✗. Column (1) for LoCoMo and LongMemEval is our own audit (0/2,486).

Evaluation	**(1)** ended state still binds	**(2)** answer is not the record	**(3)** cue–trigger disconnect	**(4)** store policy is the varied factor
LoCoMo (Maharana et al., 2024)	✗	✗	✗	✗
LongMemEval (Wu et al., 2025)	✗	✗	✗	✗
STALE (Chao et al., 2026)	–	✓	–	✗
Memora (Uddin et al., 2026)	✗	✗	–	✗
ForgetEval (Yang, 2026)	✗	✗	–	✓
MemSyco-Bench (Xiang et al., 2026)	–	✓	–	–
LoCoMo-Plus (Li et al., 2026)	✗	✓	✓	✗
CCD (this paper)	✓	✓	✓	✓

model long-term belief revision", and no lifecycle policy acts on its store. We vary the originating event's state and test the effect of suppressing records marked resolved. MemSyco-Bench's *respect memory's applicable scope* task (Xiang et al., 2026) is the nearest published probe of the attribution layer; our substitution arms ask, more narrowly, whether a lifting record belongs to the case being answered (Table 4).

Updates and forgetting. Memora (Uddin et al., 2026) introduces a metric "that penalizes reliance on obsolete or invalidated memory"; STALE (Chao et al., 2026) scores whether the *updated* state is acted on downstream, in scenarios framed as conflicts; ForgetEval (Yang, 2026) checks store mutations, a case passing iff the must-not-contain set is absent from the top-10. These benchmarks test responses to outdated information or changes to stored facts; they do not isolate the decision required by a resolved event whose restriction remains in force. On the two standard long-horizon sets that decision is unqueried: $0/2{,}486$ questions across LoCoMo (Maharana et al., 2024) and LongMemEval (Wu et al., 2025) require applying a resolved event as a constraint.

Memory maintenance mechanisms. Conflict-based revocation treats staleness as supersession: TEPA (Zhou et al., 2026) revokes a precedent when fresher evidence contradicts it under the same key, and Zep (Rasmussen et al., 2025) marks contradicted edges invalid while retaining them. In our RESOLVED cell the event's current state changes while its occurrence remains true and its restriction remains binding; revoking a superseded state alone does not determine whether the binding evidence should be retained, and none of the designs we examined types an event that has resolved and still constrains. DeMem (Zou et al., 2026) organizes limited memory around decision loss and motivates the slot-based comparison of subsection 5.1 (DAM (Sun et al., 2025) extends it to sequential decisions); Li & Li (2026) separate stored records from query-conditioned use. Parametric editing, knowledge-conflict and time-sensitive QA, and validity- or time-typed stores treat which stored value currently holds (Meng et al., 2022; 2023; Xu et al., 2024; Chen et al., 2021; Zhang & Choi, 2021; Wang, 2026; Wang et al., 2026; Sun & He, 2026; Tao et al., 2026; Su et al., 2026); tiered architectures offer protected storage (Sumers et al., 2024; Packer et al., 2023; Wang & Chen, 2025), and constraint force can erode while its fact survives (Nakayashiki, 2026; Sun et al., 2026). We measure the effect of suppressing resolved-event evidence and test whether exempting binding-typed records preserves it.

7 CONCLUSION

A constraint can outlive the event that created it, and a maintenance rule that reads only event state cannot tell the two apart. On frozen stores from a native `mem0` run, an imposed broad-deletion pass removes the reachable binding evidence ($11/53 \to 0$). At a fixed retrieval budget, typed exemption protects binding evidence while suppressing transient clutter, under oracle type labels. Release requires a different judgment: in controlled context substitutions, the case's own lifting sentence leads to release in $32/43$ cases against $8/43$ for its resolution sentence, yet another patient's lifting sentence triggers $30/43$ false releases on RESOLVED. These results point to three requirements for agent memory: preserve the evidence behind a constraint, retrieve the records that lift it, and link those records to the correct case.

AI USE STATEMENT

Large language model assistants were used to generate synthetic data and to assist with language polishing of the manuscript.

REFERENCES

Hanxiang Chao, Yihan Bai, Rui Sheng, Tianle Li, and Yushi Sun. STALE: Can LLM agents know when their memories are no longer valid? *arXiv preprint arXiv:2605.06527*, 2026.

Wenhu Chen, Xinyi Wang, and William Yang Wang. A dataset for answering time-sensitive questions. In *NeurIPS Datasets and Benchmarks Track*, 2021.

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.

Yifei Li, Weidong Guo, Lingling Zhang, Rongman Xu, Muye Huang, Hui Liu, Lijiao Xu, Yu Xu, and Jun Liu. LoCoMo-Plus: Beyond-factual cognitive memory evaluation framework for LLM agents. *arXiv preprint arXiv:2602.10715*, 2026.

Yuhang Li and Yuchen Li. What should an agent forget? separating what is stored from what is used. *arXiv preprint arXiv:2609.10263*, 2026.

Zhiyu Li, Chenyang Xi, Chunyu Li, Ding Chen, Boyu Chen, Shichao Song, et al. MemOS: A memory OS for ai system. *arXiv preprint arXiv:2507.03724*, 2025.

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 ACL*, 2024.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in GPT. In *Advances in Neural Information Processing Systems*, 2022.

Kevin Meng, Arnab Sen Sharma, Alex Andonian, Yonatan Belinkov, and David Bau. Mass-editing memory in a transformer. In *International Conference on Learning Representations*, 2023.

Kazuki Nakayashiki. When stale constraints go unchecked: Budgeted verification failures in inherited agent memory. *arXiv preprint arXiv:2608.25553*, 2026.

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.

Preston Rasmussen, Pavlo Paliychuk, Travis Beauvais, Jack Ryan, and Daniel Chalef. Zep: A temporal knowledge graph architecture for agent memory. *arXiv preprint arXiv:2501.13956*, 2025.

Miao Su, Yucan Guo, Zhongni Hou, Long Bai, Zixuan Li, Yufei Zhang, Guojun Yin, Wei Lin, Xiaolong Jin, Jiafeng Guo, and Xueqi Cheng. Beyond dialogue time: Temporal semantic memory for personalized LLM agents. *arXiv preprint arXiv:2601.07468*, 2026.

Theodore R. Sumers, Shunyu Yao, Karthik Narasimhan, and Thomas L. Griffiths. Cognitive architectures for language agents. *Transactions on Machine Learning Research*, 2024.

Changzhi Sun, Xiangyu Chen, Jixiang Luo, Dell Zhang, and Xuelong Li. Beyond heuristics: A decision-theoretic framework for agent memory management. *arXiv preprint arXiv:2512.21567*, 2025.

Haofei Sun and Lin He. When memory updates but behavior does not: Repairing implicit stale dependencies in personalized agent responses. *arXiv preprint arXiv:2608.01619*, 2026.

Yiheng Sun, Huifei Wang, Yancheng Zhu, Zhenyu Li, Zebin Zhao, and Yifan Yuan. When "must" becomes "maybe": Constraint weakening in LLM agent workflows. *arXiv preprint arXiv:2608.24569*, 2026.

Zhen Tao, Jinxiang Zhao, Peng Liu, Dinghao Xi, Yanfang Chen, Wei Xu, and Zhiyu Li. MemConflict: Evaluating long-term memory systems under memory conflicts. *arXiv preprint arXiv:2605.20926*, 2026.

Md Nayem Uddin, Kumar Shubham, Eduardo Blanco, Chitta Baral, and Gengyu Wang. From recall to forgetting: Benchmarking long-term memory for personalized agents. *arXiv preprint arXiv:2604.20006*, 2026.

Heng Wang, Yifei Li, Lingling Zhang, Pengyu Li, Xinyu Che, Xinyu Zhang, and Zesheng Yang. QUMem: Personalized memory for query-conditioned user-state inference in LLM agents. *arXiv preprint arXiv:2608.16168*, 2026.

Yu Wang and Xi Chen. MIRIX: Multi-agent memory system for LLM-based agents. *arXiv preprint arXiv:2507.07957*, 2025.

Ziming Wang. TOKI: A bitemporal operator algebra for contradiction resolution in LLM-agent persistent memory. *arXiv preprint arXiv:2606.06240*, 2026.

Di Wu, Hongwei Wang, Wenhao Yu, Yuwei Zhang, Kai-Wei Chang, and Dong Yu. LongMemEval: Benchmarking chat assistants on long-term interactive memory. In *Proceedings of ICLR*, 2025. arXiv preprint arXiv:2410.10813, 2024.

Zhishang Xiang, Zerui Chen, Yunbo Tang, Zhimin Wei, Ruqin Ning, Yujie Lin, Qinggang Zhang, and Jinsong Su. MemSyco-Bench: Benchmarking sycophancy in agent memory. *arXiv preprint arXiv:2607.01071*, 2026.

Rongwu Xu, Zehan Qi, Zhijiang Guo, Cunxiang Wang, Hongru Wang, Yue Zhang, and Wei Xu. Knowledge conflicts for LLMs: A survey. *arXiv preprint arXiv:2403.08319*, 2024.

Wujiang Xu, Zujie Liang, Kai Mei, Hang Gao, Juntao Tan, and Yongfeng Zhang. A-MEM: Agentic memory for llm agents. *arXiv preprint arXiv:2502.12110*, 2025.

Meriem Yacoubi, Pia Schmidt, Nenad Petrovic, Ahmed Frikha, Martin Kirchhoff, and Alois Knoll. MemoryLACE: Memory lifecycle-aware consolidation and evidence retrieval. *arXiv preprint arXiv:2609.03201*, 2026.

Dongxu Yang. Control-plane placement shapes forgetting: An architectural study of agent memory across thirteen system configurations. *arXiv preprint arXiv:2606.15903*, 2026.

Michael J. Q. Zhang and Eunsol Choi. SituatedQA: Incorporating extra-linguistic contexts into QA. In *Proceedings of EMNLP*, 2021.

Zeyu Zhang, Ziqing Wang, and Kaize Ding. MedLoCoMo: A long-context multi-session medical dialogue benchmark for large language models. *arXiv preprint arXiv:2607.22566*, 2026.

Yan Zhou, Yue Ouyang, Kaiyang Zheng, and Suncheng Xiang. TEPA: Revoking stale memories for conflict-robust language agents. *arXiv preprint arXiv:2608.07429*, 2026.

Mingxi Zou, Zhihan Guo, Langzhang Liang, Zhuo Wang, Qifan Wang, Qingsong Wen, Irwin King, Lizhen Qu, and Zenglin Xu. Remember the decision, not the description: A rate-distortion framework for agent memory. *arXiv preprint arXiv:2605.10870*, 2026.

A Limitations in full

CCD uses synthetic cases with one constraint per event. Its design cannot disentangle errors in patient ownership, event identity, and action scope.

Answer-layer scoring is deterministic term matching on the answer's conclusion. Typed exemption uses oracle type labels. The rank analysis applies to the fixed, query-independent suppression policies defined in this paper. The MedLoCoMo redundancy correlation is observational.

Protecting a binding constraint does not determine when it expires. We test how answers change when lifting evidence is supplied, but do not implement or validate an automatic exit rule.

B Denominators and evaluation subsets

Each experiment reports the largest set on which its measurement is defined.

n	Definition
53	all clinical cases; also the per-variant corpus count.
50	resolution-evidence retrieval: 53 minus the three RESOLVED follow-ups that never state the index event resolved.
44	release decisions: 53 minus the nine cases whose lifting contradicts the relation's own stated permanent basis.
43	substitution (Table 3): the 44 minus one case lacking a donor lifting or a resolution-only unit.
38	suppression-strength and state-axis runs: the cases of the eight in-vocabulary relations; the six held-out relations (15 cases) are excluded.
37	the gate run (subsection D.3): the release-scorable set under the earlier gold audit, before seven missing lifting sentences were repaired; the run was not repeated on the current 44, so its cells pair within the run only.
26	the trade-off set (subsection 4.2): cases where, at s=0, the index-event unit is in Q_{bind}'s top-k and the resolution unit is in Q_{state}'s top-k (53 minus 13 with the event initially unreachable, minus 14 of the remaining 40 with the resolution initially unreachable).
116	corpora ingested in the native `mem0` run: 53 RESOLVED, 53 ONGOING, 10 NEVER.
159	replay-validation rows: 53 RESOLVED corpora × three queries.

C RUN INDEX

Each number in the paper comes from one of the following runs.

Run	Store / retriever	Labels	Answerer	Reported in
Native trace (A)	`mem0`, its own extraction & conflict resolution	none	—	subsection 4.1
Frozen-store fork (B)	`mem0` snapshots, stack's own embedder	oracle	—	subsection 4.1
State-axis test	same fork	forced-on	`gemini-2.5-flash`	subsection 4.1
Shift thresholds	BM25 skeleton (dense check)	oracle	—	subsection 4.2, Appendix E
Policy poles, λ	LangMem `InMemoryStore`; MedLoCoMo text	oracle	—	subsection 4.2
Four-cell retrieval	shared pipeline, k=20 ($3k$ for retrieve-more)	oracle	—	Table 1
Typed exemption	LangMem `InMemoryStore`; BM25 check	oracle types	`gemini-2.5-flash`	subsection 5.2
Decision table	shared candidate pool	oracle	`gemini-2.5-flash`	Table 2
Substitution	typed+fwd retrievals, fixed context	oracle	`gemini-2.5-flash`; `claude-sonnet-5`	Table 3
Gate run (pre-repair)	fixed candidates	oracle	`gemini-2.5-flash`	subsection D.3

D AUDIT PROTOCOLS AND THE PRE-REPAIR RUN

D.1 THE BENCHMARK AUDIT (0 OF 2,486)

The corpora are the released LoCoMo set (10 conversations, 1,986 QA items) and the LongMemEval oracle split (500 questions), both fixed into the artefact repository. An item counts iff its gold answer requires *applying* a resolved event as a constraint on a decision the question does not name (the C0B1 cell); an item that asks for the resolution itself, or uses it as a date anchor, does not. The predictions were registered before any item content was read. A lexical screen over question and answer text flagged candidates (85 on LongMemEval); one author then read every item, flagged or not, and the items nearest the criterion are logged with the reason each fails it. The 38 LoCoMo questions that refer to resolved events all ask for recall of the resolution or its surrounding content; LoCoMo's 446 adversarial items contain none that is unanswerable because a fact lapsed, and LongMemEval's 78 knowledge-update items all score the updated value or name the old value as the retrieval target. All items were labelled by a single author; the per-item verdicts and audit scripts ship with the artefacts, so the audit can be re-run and re-judged in full.

D.2 RELEASE-GOLD AUDIT AND REPAIR

Every RELEASED follow-up was audited by a tiered rule: T1, an explicit lifting of the restriction; T2, an implicit release ("no restriction / resume as usual"); T3, no lifting sentence. The rule reads only the corpus text and each relation's stated basis, never a model answer, and the same rule fires on $0/53$ RESOLVED variants (its false-positive check). It found seven T3 follow-ups, repaired by replacing only the final sentence with the explicit lifting phrasing of same-relation siblings (index-event text unchanged, memory units re-extracted with the same extractor), and nine cases, three cumulative-exposure relations $\times$ three variations, whose lifting contradicts the relation's stated permanent basis and cannot be legitimately rewritten; their RELEASED variant is excluded from release scoring and their other variants are retained. Post-repair the rule classifies 46 T1 and 7 T2, giving the 44 scorable release cases of section 2. All release-axis results in this paper were produced on the repaired set; the pre-repair corpus and the full audit table ship with the artefacts.

D.3 THE PRE-REPAIR GATE RUN

A query-conditioned gate was tested on the 37-case pre-repair scorable set (Appendix B), at fixed candidates, budget and answerer: an (m, q, t)-conditioned gate reaches joint release-and-retain accuracy $11/37$ against $9/37$ for relevance-only selection, and a ground-truth verdict reaches $14/37$. Neither paired comparison (gate against relevance-only, verdict against gate) shows a reliable gain in this run. Splitting the same 37 by whether a sentence recording the lifting is in the retrieved context separates the conditions sharply: with it, the ground-truth verdict releases and retains in $11/12$; without it no condition exceeds $3/25$. The run was not repeated on the repaired 44, so its cells pair within the run only.

E THE EVICTION CONDITION IN CLOSED FORM

A suppressor applies one shift to every query, so a single scalar decides for every query at once what stays retrievable.

Lemma 1. *Under Equation 4, let $r_{\min} = \min_{i \in R} \mathrm{rank}(i)$ be the pre-shift rank of the best-ranked suppressed unit. Its post-shift rank is*

$$\mathrm{rank}' \;=\; \big|\{\, i \notin R \;:\; \mathrm{rank}(i) < r_{\min} + s \,\}\big| \;+\; 1. \tag{5}$$

Proof. After re-sorting, the units preceding the best suppressed unit are exactly the unsuppressed units whose key, their unshifted rank, is below $r_{\min} + s$; no suppressed unit precedes it by definition of $r_{\min}$. Key ties (an unsuppressed unit at exactly $r_{\min} + s$) are broken by pre-shift rank under a stable re-sort; the strict inequality in Equation 5 encodes this. □

Equation 5 addresses the best-ranked suppressed unit. For an arbitrary suppressed target j with pre-shift rank r_j and a_j suppressed units ranked above it, the stable re-sort gives $\mathrm{rank}'(j) = a_j + 1 + |\{i \notin R : \mathrm{rank}(i) < r_j + s\}|$; Equation 5 is the $a_j{=}0$ case. The per-case thresholds reported in this paper are computed by exact re-sorting with binary search over integer s, not from Equation 5 directly.

Corollary 1 (Eviction threshold)**.** *Write $U(x) = |\{i \notin R : \mathrm{rank}(i) < x\}|$. Eviction from the top-$k$ requires $U(r_{\min} + s) \geq k$, so the minimum sufficient shift is $s_{\mathrm{evict}} = \min\{s \in \mathbb{Z}_{\geq 0} : U(r_{\min} + s) \geq k\}$ when this set is nonempty, which requires at least k unsuppressed units; otherwise $s_{\mathrm{evict}} = +\infty$. Approximating U by a constant density of unsuppressed units, $U(x) \approx (1 - \rho_s)\, x$ where ρ_s is the fraction of high-ranked units that are suppressed, yields*

$$s_{\mathrm{evict}} \;\approx\; \frac{k}{1 - \rho_s} \;-\; r_{\min}. \tag{6}$$

When at least k unsuppressed units exist, the exact threshold is itself closed form: writing u_k for the rank of the k-th unsuppressed unit, $s_{\mathrm{evict}} = \max(0,\, u_k - r_{\min} + 1)$. Against the exact condition, Equation 6 has a median absolute error of half a rank position on CCD, and it is stable under a dense retriever. For a target other than R's top-ranked unit the same form is a surrogate whose error is included in that figure.

Theorem 1 (Suppression window)**.** *Fix a case and let $s_{\mathrm{ev,bind}}$ be the smallest shift that evicts the event evidence under Q_{bind}, and $s_{\mathrm{res,state}}$ the smallest shift that evicts the resolution evidence under Q_{state}, each $+\infty$ where eviction is impossible. A single query-independent shift meets the suppressor's contract (event evidence evicted under Q_{bind}, resolution evidence retained under Q_{state}) iff it lies in $W = [\, s_{\mathrm{ev,bind}},\ s_{\mathrm{res,state}})$, so such a setting exists iff $s_{\mathrm{ev,bind}} < s_{\mathrm{res,state}}$.*

The retention trade-off of subsection 4.2 has the same structure with different thresholds: a shift clearing the event from Q_{state} while keeping it under Q_{bind} exists iff $s_{\mathrm{ev,state}} < s_{\mathrm{ev,bind}}$, and requiring the resolution record to survive adds $s < s_{\mathrm{res,state}}$. The $21/26$ and $41/53$ counts of subsection 4.2 test the retention trade-off; the $17/26$ and $38/53$ counts test Theorem 1. All counts use exact thresholds computed on the re-sorted ranking; substituting Equation 6 gives a two-statistic estimate whose boundary cases can flip, so it is used to read logs, not to produce the reported counts.

On CCD under the broad labelling W is empty for $38/53$ cases, but 13 have the event outside the top-k before any suppression, and 14 of the remaining 40 have the resolution already outside Q_{state}'s window; both are initial retrieval failures rather than suppression costs. On the 26 where both records start inside their windows, the set of subsection 4.2, W is empty for $17/26$, and in every one the resolution is evicted at or before the event.