When the Latest State Does Not Stick: State-Flow for Reliable Decisions over Evolving Records

Anonymous authors
Paper under double-blind review

ABSTRACT

The latest record is not always the current state. In evolving organizational records, the value that should govern a decision may need to be reconstructed from a sequence of updates involving source authority, hierarchical inheritance, entity migration, persistent overrides, and derived rules. Meanwhile, outdated but plausible values remain visible in the history and can still influence an LLM's answer. We study this failure mode, which we call the derived-state persistence gap. Using 1,200 mechanism-isolating benchmark instances, we apply matched interventions that remove stale semantics, explicitly provide the resolved state, or hide history at decision time. Across seven model configurations, these interventions improve macro exact-set accuracy from 62.7% to 89.4-91.1%, showing that failures arise not only from stale information, but also from reconstructing and reasoning over the current state. We introduce STATEFLOW, which verifies whether each update is authoritative and applicable, executes valid state transitions externally, and presents the decision model with a canonical current state instead of the full history. STATEFLOW reaches 94.8% accuracy, improving over Full History by 32.1 points and over a deployable LLM State Rewrite baseline by 12.1 points. A format-matched reference-state ablation reaches the same 94.8% accuracy with identical final decision inputs, localizing the remaining system-level gap to state construction. A natural-history evaluation on public Jira issues shows the same benefit from explicit current-state representation and history isolation. Our results suggest that reliable reasoning over evolving records requires reconstructing the state that is valid now, rather than simply attending to what was written most recently.

1 Introduction

Long-context language models are increasingly evaluated through retrieval, multi-document question answering, aggregation, and realistic long-input tasks (Hsieh et al., 2024; Bai et al., 2024; Yen et al., 2025; Modarressi et al., 2025). In decisions involving policies, configurations, and evolving organizational records, finding relevant information is only the first step. The answer may require composing hierarchical assignments, inherited defaults, entity-specific overrides, membership changes, and conditional rules into one current state. Meanwhile, stale or non-authoritative records remain explicit in the input even though they should no longer determine the answer. Evidence can thus be available yet used unreliably because of its location or the overall context burden (Liu et al., 2024; Du et al., 2025).

This setting distinguishes directly stated values from derived states. A final value may require composing several operations, while a superseded value can remain directly visible and locally plausible. We call the resulting behavioral failure pattern the *derived-state persistence gap*. We operationalize it through matched interventions that selectively neutralize stale semantics, materialize the resolved current state, or remove historical competition at decision time.

We construct four matched diagnostic conditions. Full History requires state reconstruction from the complete sequence. Neutral Replacement preserves approximate length and position while replacing decision-relevant stale records with irrelevant text. State Write-Back retains the history but appends

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.

the correct canonical current state. State-Only supplies the same *semantic* state and decision rules without the history. Across seven model configurations and 1,000 held-out instances, the three interventions improve macro-average exact-set accuracy from 62.7% to 89.4–91.1%. Their similarly large gains reveal a multi-factor failure pattern involving stale-semantic competition, distributed state reconstruction, and decision-time history visibility.

Motivated by this result, we introduce STATEFLOW, a framework that maps structured event traces to executable operations, verifies authority and applicability, executes updates in order, serializes a canonical state, and invokes the decision model in a fresh context. STATEFLOW reconstructs the state directly from visible records, metadata, and policies and reaches 94.8% macro-average accuracy. A format-matched State-Only ablation uses the reference state with the same serializer and decision template; prompt-identity verification confirms token-identical final decision inputs on 1,000/1,000 held-out instances and the same 94.8% accuracy under deterministic decoding. A deployable LLM State Rewrite baseline reaches 82.6%, leaving a 12.1-point gap to STATEFLOW under the same final-answer interface.

Our contributions are: (1) a mechanism-isolating formulation of authoritative current-state reconstruction that combines supersession with source authority, hierarchy, migration, and derived-rule closure; (2) matched interventions separating stale-semantic interference, explicit state representation, and history isolation; (3) an authority-aware external state-execution framework; (4) a 1,200-instance benchmark suite plus project-disjoint natural-history validation of state representation and history isolation on public Jira histories; and (5) a decomposition of state construction, decision representation, and downstream decision error, together with a deployable state-rewrite comparison.

2 RELATED WORK

Long context and memory. Long-context benchmarks test evidence localization, aggregation, multi-hop tracing, and latent association (Liu et al., 2024; Hsieh et al., 2024; Bai et al., 2024; Yen et al., 2025; Modarressi et al., 2025). Long-term memory benchmarks extend this setting to cross-session retention and updating (Maharana et al., 2024; Wu et al., 2025; Xu et al., 2025). These lines establish the context and memory burden within which current-state reconstruction must operate.

Evolving-state and authority benchmarks. Recent benchmarks directly target update-sensitive memory. MemoryAgentBench includes FactConsolidation for conflict resolution and selective forgetting (Hu et al., 2026). STALE probes state resolution, resistance to stale premises, and adaptation to implicit state changes (Chao et al., 2026). StateMemBench scores whether answers follow current or superseded state and includes standing rules and derived-value recomputation (Fan et al., 2026). AuthMem-Bench isolates whether memory consolidation preserves source authority into downstream action (Zhan et al., 2026). These benchmarks establish supersession, dependency propagation, and authority as distinct failure axes. Our benchmark targets their intersection in an executable organizational state: source authority and applicability interact with hierarchical inheritance, entity migration, persistent overrides, fixed-point derived rules, and set-valued downstream decisions. Appendix A summarizes the design axes.

State tracking and external execution. Dialogue state tracking represents state through explicit updates (Lesci et al., 2023), while temporal QA and world-state tracking study evolving knowledge and validity over time (Meem et al., 2024; Lyu et al., 2025; Zhu et al., 2025). Program- and tool-aided methods delegate computation or verification to external runtimes (Gao et al., 2023; Yao et al., 2023; Schick et al., 2023; Gou et al., 2024). Recent memory systems likewise externalize update resolution through deterministic version-aware aggregation (Reddy & Challaram, 2026) and source-supported transaction boundaries (Cui et al., 2026). STATEFLOW executes the full current-state transition system and then removes the original history from the final decision context; Section 5.1 compares this design with LLM State Rewrite under the same final-answer interface.

3 TASK AND DIAGNOSIS

3.1 TASK SETTING

Intuitively, each instance represents an organization in which Products belong to Suites, Suites belong to Divisions, and lower-level entities inherit field values unless explicitly overridden. Chronological records may change field values, move Products between Suites, or trigger conditional updates. Only records that are both authoritative and applicable modify the current state.

Formally, an instance contains governing rules R, an initial hierarchical state S_0, an ordered record sequence $H = \langle h_1, \ldots, h_T \rangle$, source metadata $M = \langle m_1, \ldots, m_T \rangle$, an authority policy P, and a final query q. The hierarchy contains Divisions, Suites, and Products with fields such as SLA, Retention, ResponseWindow, SecurityTier, and AuditInterval. These field names are task attributes rather than additional reasoning primitives; the core mechanisms are inheritance, override, migration, authority, applicability, and conditional derivation.

A valid parent update propagates to descendants that currently inherit the affected field. A Product assignment creates an explicit override; migration changes active Suite membership, preserves Product overrides, and refreshes non-overridden fields from the target Suite. Non-authoritative or inapplicable records remain visible but do not change the state. A deterministic reference executor applies

$$S_t = U_{R,P}(S_{t-1}, h_t, m_t), \quad t = 1, \ldots, T, \tag{1}$$

and the reference decision is $Y^{\mathrm{ref}} = D_R(S_T, q)$. Both Y^{ref} and the model prediction $\hat{Y}$ are sets of Products. Exact-set correctness requires $\hat{Y} = Y^{\mathrm{ref}}$; over-inclusion and missing output are $\hat{Y} \setminus Y^{\mathrm{ref}} \neq \emptyset$ and $Y^{\mathrm{ref}} \setminus \hat{Y} \neq \emptyset$, respectively.

A current value is *derived* when it depends on inheritance, migration, conditional propagation, or interacting operations. A record is *stale* with respect to S_T when it was once applicable but has been superseded. Staleness differs from non-authority: a stale record may have validly modified an earlier state, whereas a non-authoritative record was never permitted to do so.

3.2 MATCHED DIAGNOSTIC CONDITIONS

Full History contains all rules, hierarchy, records, and the query. Neutral Replacement replaces each decision-relevant stale authoritative record with task-irrelevant text of similar length at the same position, without entity, field, value, update, or query cues. State Write-Back appends the reference executor's canonical state while retaining the full history. State-Only provides the same semantic state, decision-relevant rules, and query, but omits the record sequence.

The Neutral–Full contrast isolates stale-semantic interference at approximately matched context length and position. Write-Back–Full measures the effect of explicitly representing the resolved state while retaining history. State-Only–Full adds history isolation at decision time. The similarly large gains across these interventions support a multi-factor account spanning stale-semantic competition, state reconstruction, and history visibility; Section 6 reports the paired significance pattern.

4 STATEFLOW

STATEFLOW separates record interpretation and state execution from final decision making (Figure 2). A mapping function produces a candidate operation

$$o_t = I(h_t, m_t) = (\tau_t, e_t, \theta_t, c_t), \tag{2}$$

where τ_t is the operation type, e_t the target, θ_t the payload, and c_t any applicability condition. In our schema-constrained event language, I is deterministic; the framework itself is parser-agnostic.

Let $A_P(o_t, m_t)$ test source authority under policy P and $V_R(o_t, \tilde{S}_{t-1})$ test applicability against the evolving state. With $\tilde{S}_0 = S_0$, the execution gate is

$$g_t = A_P(o_t, m_t) \wedge V_R(o_t, \tilde{S}_{t-1}), \tag{3}$$

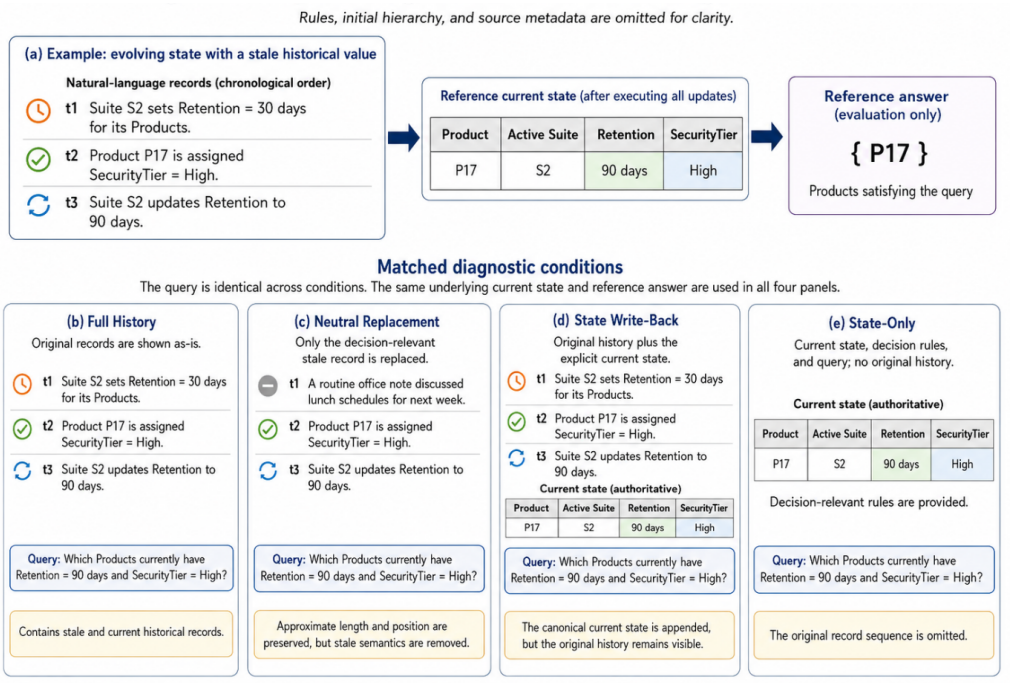

Figure 1: Four matched diagnostic conditions share the same underlying current state, query, and reference answer. Neutral Replacement removes stale semantics while approximately preserving length and position; Write-Back and State-Only supply the same canonical semantic state but differ in history visibility.

and the state transition is

$$\tilde{S}_t = \begin{cases} \mathrm{Cl}_R\Big(\mathrm{Apply}(\tilde{S}_{t-1}, o_t)\Big), & g_t = 1, \\ \tilde{S}_{t-1}, & g_t = 0. \end{cases} \tag{4}$$

Here Cl_R repeatedly applies enabled derived rules to their unique fixed point. Generation rejects rule sets that do not terminate or admit a unique resolved state. Derived values remain part of the evolving state for later operations; migration updates active membership, preserves explicit Product overrides, and refreshes inherited values.

After all records are processed,

$$C = \mathrm{Serialize}(\tilde{S}_T), \qquad \hat{Y} = f_\phi(C, R_q, q), \qquad R_q \subseteq R. \tag{5}$$

C contains active memberships and one resolved value per relevant entity–field pair, excluding superseded and rejected alternatives. The final decision call receives C, the query, and only the rules required to interpret that query. At inference time, the pipeline uses visible records, metadata, policy, rules, and query; reference operations, labels, states, traces, and answers are used only for evaluation. Appendix G provides a complete worked trace through authority rejection, migration, inheritance, override preservation, derived-rule execution, and serialization.

5 Experimental Setup

Mechanism-isolating benchmark. The benchmark contains 1,200 underlying instances: 200 development and 1,000 held-out test instances, balanced across four complexity levels. Each instance produces four diagnostic inputs plus one StateFlow input, totaling 6,000 condition-specific inputs. Every instance contains at least one decision-relevant stale authoritative record; about half also contain a conflicting non-authoritative record. Queries contain three or four conjunctive conditions with three to eight correct Products.

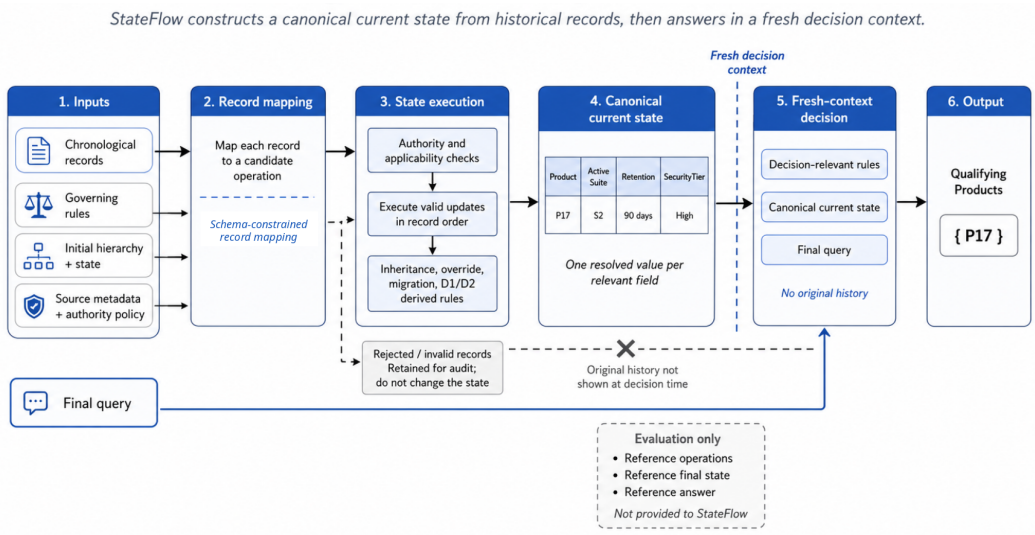

Figure 2: StateFlow maps records to candidate operations, verifies authority and applicability, executes valid updates, and exposes the canonical state, relevant rules, and query in a fresh decision context. Reference artifacts remain outside the inference pipeline and support evaluation.

Complexity increases jointly. L1 averages about 16 Products, 79 records, and 1,248 Full-History words; L4 averages 93 Products, 442 records, and 5,831 words, with longer update chains, more migrations and overrides, more conflicts, and more conditional dependencies. The reference executor removes instances with inconsistent rules, non-unique semantics, non-terminating derivations, or degenerate answer sets. Structured operations and reference artifacts are retained only for evaluation. Appendix B gives the complete generator configuration used in the reported experiments.

Models and implementation. We evaluate seven configurations from six models: o3-mini with medium and high reasoning (OpenAI, 2025); Gemini 2.5 Pro (Comanici et al., 2025); Qwen2.5-7B-Instruct (Qwen Team, 2025); DeepSeek-R1-Distill-Qwen-7B and DeepSeek-R1-Distill-Llama-8B (DeepSeek-AI, 2025); and Llama-3.1-8B-Instruct (Grattafiori et al., 2024). Each request starts from a fresh context with matched deterministic decoding. The StateFlow pipeline reproduces the reference operations, authority decisions, and final states on all 1,200 benchmark instances. This verifies implementation consistency under the schema-constrained event language, where deterministic parsing isolates state management from open-ended extraction.

Metrics and tests. The primary metric is exact-set accuracy. We also measure precision, recall, F1, over-inclusion, missing output, and parse success. All comparisons are paired by underlying instance. Confidence intervals use 10,000 paired bootstrap resamples; exact-set comparisons additionally use two-sided exact McNemar tests with Benjamini–Hochberg correction. Bootstrap resampling is performed over underlying instances; all model outputs associated with a sampled instance are kept in the same resampling block when computing macro-averages across configurations.

Public Jira natural-history validation. We sample 300 issues from the anonymized Public Jira Dataset, using 50 for development and 250 for a project-disjoint test. Each issue provides a naturally occurring chronological field-change history, with its latest snapshot serving as the reference state. This evaluation targets the two mechanisms directly represented by the Jira changelog: explicit current-state representation and decision-time history isolation. We evaluate Full History, Write-Back, and State-Only on Jira. The factorized benchmark suite continues to evaluate source authority, hierarchical inheritance, migration, and derived-rule execution.

Model configuration	Full	Neutral	Write-Back	State-Only	LLM State Rewrite	StateFlow
o3-mini high	72.8	91.9	92.0	92.3	88.6	97.0
Gemini 2.5 Pro	70.9	91.3	93.1	91.8	87.9	96.8
o3-mini medium	64.5	89.1	91.2	90.0	84.1	95.1
R1-Distill-Qwen-7B	66.7	90.8	92.8	91.5	85.0	96.3
R1-Distill-Llama-8B	59.8	88.7	90.0	90.1	80.4	93.8
Qwen2.5-7B-Instruct	56.8	88.9	90.7	91.0	79.7	93.6
Llama-3.1-8B-Instruct	47.4	85.1	87.9	84.7	72.8	90.9
Macro average	62.7	89.4	91.1	90.2	82.6	94.8

Table 1: Exact-set accuracy (%) on the 1,000 held-out mechanism-isolating benchmark instances.

Comparison	Exact-set Δ	95% CI	Over-inclusion Δ	Missing Δ
Neutral – Full	+26.7	[+25.3,+28.1]	-22.9	-8.1
Write-Back – Full	+28.4	[+27.0,+29.8]	-24.4	-8.8
State-Only – Full	+27.5	[+26.1,+28.9]	-23.9	-8.5
StateFlow – Full	+32.1	[+30.5,+33.7]	-28.2	-11.1
StateFlow – State-Only	+4.6	[+3.9,+5.3]	-4.3	-2.6
StateFlow – LLM State Rewrite	+12.1	[+11.0,+13.3]	-10.9	-5.7

Table 2: Macro-average paired effects in percentage points. The original StateFlow–State-Only row uses different decision representations; Section 6.2 isolates that factor.

5.1 Deployable State-Management Baseline

For a deployable comparison, we add an *LLM State Rewrite* baseline. It processes records chronologically. At step t, the model receives the previous serialized state, the new record and source metadata, and the governing rules, then emits a complete updated JSON-like state under a fixed schema. The final query is answered from the rewritten state in a fresh context using the same deterministic decoding policy as the corresponding model configuration. The baseline receives no stale-record labels, reference operations, reference state, or reference answer. We run this protocol for all model configurations; Appendix C records the prompt and serializer interfaces used in the experiments.

6 Results

6.1 Main Results and Matched Effects

Table 1 shows that Full History is weakest for every measured configuration. Its macro-average accuracy is 62.7%. Neutral Replacement, State Write-Back, and State-Only reach 89.4%, 91.1%, and 90.2%, respectively, with similar performance among the three diagnostic interventions. StateFlow reaches 94.8%. The LLM State Rewrite baseline reaches 82.6%; in the paired evaluation, StateFlow exceeds it by 12.1 points with a [11.0,13.3] 95% bootstrap interval. LLM State Rewrite recovers a substantial fraction of the gain, while deterministic state execution retains a 12.1-point advantage.

Neutral Replacement improves accuracy by 26.7 points, Write-Back by 28.4 points, and State-Only by 27.5 points. The comparable effect sizes support a multi-factor account in which stale-semantic competition, explicit state representation, and history visibility all contribute to the Full-History deficit.

The intermediate diagnostic conditions have small macro differences: none of the 21 model-level McNemar comparisons among them remains significant after correction, while all 28 intervention-versus-Full comparisons remain significant. Appendix D summarizes these corrected test counts. The +4.6-point StateFlow–State-Only contrast combines state source with decision representation; the format-matched ablation below separates those factors.

Complexity	Full	Neutral	Write-Back	State-Only	StateFlow
L1	85.9	96.4	95.9	96.7	98.6
L2	72.3	92.1	93.8	92.8	96.5
L3	54.7	89.4	88.9	89.2	93.7
L4	37.9	79.7	85.8	82.1	90.4
Macro average	62.7	89.4	91.1	90.2	94.8

Table 3: Exact-set accuracy (%) by joint complexity level.

6.2 Format-Matched State-Only Ablation

The original State-Only condition uses the reference canonical state in its diagnostic template, whereas StateFlow uses the production serializer and decision template. *Format-Matched State-Only* serializes the reference state with the same field order, rules block, system/user instructions, and deterministic decoding used by StateFlow. Prompt-identity verification confirms token-identical final decision inputs on 1,000/1,000 held-out instances. Deterministic decoding therefore produces the same per-model predictions as StateFlow, yielding 94.8% macro exact-set accuracy and a 0.0-point difference. This removes decision-stage representation from the comparison and localizes the substantive system difference upstream to state construction: State-Only receives the reference state, while StateFlow reconstructs the same state from the record history.

6.3 Error Patterns, Complexity, and Decision Decomposition

StateFlow reduces over-inclusion by 28.2 points and missing output by 11.1 points relative to Full History. Under Full History, rollback-compatible predictions—incorrect selections that would satisfy the query under a superseded authoritative value—occur in 4.9–14.2% of instances and account for 17.9–27.0% of exact-set errors. Write-Back lowers rollback rates to 0.5–2.1%. Neutral Replacement has structural zero stale rollback because the relevant claim is removed; rollback is not applicable to State-Only or StateFlow because their decision contexts omit the history.

Full History declines from 85.9% at L1 to 37.9% at L4 (Table 3). All interventions reduce this degradation, but their ordering again varies by level. StateFlow remains above 90% at L4. Its remaining failures occur downstream of state reconstruction: the reference-executable pipeline exactly reproduces the reference final state, so applying the deterministic reference decision function D_R to the reconstructed state yields 100% exact-set accuracy by construction. Because state reconstruction is exact, the remaining gap to LLM decision accuracy arises in query-predicate interpretation, set emission, or output formatting.

As a descriptive cross-configuration analysis, reasoning-oriented configurations average 66.9% under Full History versus 52.1% for direct-instruction models, a 14.8-point gap; the gap contracts to 2.5–3.5 points under Neutral Replacement, Write-Back, and State-Only. The same contraction appears in three family/effort-aligned contrasts: o3-mini high versus medium is +8.3 points under Full History and +0.8 to +2.8 under the three interventions; R1-Distill-Qwen-7B versus Qwen2.5-7B-Instruct is +9.9 and +0.5 to +2.1; R1-Distill-Llama-8B versus Llama-3.1-8B-Instruct is +12.4 and +2.1 to +5.4. These configuration contrasts show the largest separation when the model must reconstruct state from history; once state is explicit, the gap is much smaller. Appendix E reports the per-condition differences.

Appendix F reports the expanded metric table and residual-error taxonomy. Among the 365 non-exact StateFlow outputs across the seven model-by-instance evaluations, query-predicate misapplication is the largest category (188, 51.5%), followed by omitted/extra Products (137, 37.5%) and output-format/parse violations (40, 11.0%).

6.4 Natural-History Results on Public Jira

On the 250 held-out Jira issues, the measured Write-Back and State-Only conditions improve exact-set accuracy over Full History by 13.6 and 18.4 points, from 70.3% to 83.9% and 88.7%, respectively. Over-inclusion falls from 24.0% to 10.8% and 5.2%. The direction matches the mechanism-isolating

Condition	State Exact	Field Acc.	Decision Exact	F1	Over	Missing
Full History	-	-	70.3	93.0	24.0	13.6
Write-Back	-	-	83.9	96.6	10.8	9.2
State-Only	100.0	100.0	88.7	97.8	5.2	7.6

Table 4: Public Jira results (%) on the 250 held-out issues. State-Only receives the reference state, so its state reconstruction columns are 100% by construction.

benchmark and supplies natural-history evidence for explicit current-state representation and decision-time history isolation.

7 DISCUSSION AND LIMITATIONS

Our results suggest that failures over evolving records go beyond retrieval. Full History reaches 62.7% macro exact-set accuracy, while the three matched interventions reach 89.4-91.1%; at L4, Full History falls to 37.9% while STATEFLOW remains at 90.4%. This extends long-context findings that available evidence may still be used unreliably (Liu et al., 2024; Du et al., 2025). Recent memory studies likewise show that models can retain updated information yet still act on outdated state, and that supersession, dependencies, and source authority matter for reliable memory (Chao et al., 2026; Fan et al., 2026; Zhan et al., 2026). The problem is therefore not only finding evidence, but determining what remains valid now.

For real systems, this suggests separating historical records from the state used for decisions. Old records remain useful for audit, explanation, and recovery, but should not automatically retain decision-time influence. The Jira results provide initial natural-history support: explicit state representation raises accuracy from 70.3% to 83.9%, and history isolation raises it to 88.7%. This distinction is relevant to long-running agents, configuration systems, access control, and workflows where rules and facts change over time.

The main limitation is that our benchmark assumes explicit source roles, deterministic order, and fully specified rules, while Jira does not test all mechanisms together. Real records may be ambiguous, incomplete, conflicting, or delayed. Future work should therefore study state construction under uncertainty: retaining provenance and confidence for inferred updates, preserving unresolved conflicts, allowing abstention or clarification, and supporting versioned rollback and recovery. Recent authority-aware and transactional memory work points toward these directions (Zhan et al., 2026; Cui et al., 2026). Evaluation should also test downstream actions, uncertainty recognition, and safe correction.

8 CONCLUSION

This work identifies the derived-state persistence gap: having relevant history available does not guarantee that an LLM will act on the state that is valid now. STATEFLOW reaches 94.8% exact-set accuracy, 32.1 points above Full History, and the format-matched ablation shows that when the same current state is presented in the same form, the remaining difference disappears. This localizes the main system-level bottleneck to state construction before the final decision.

The broader contribution is a distinction between what a system remembers and what should govern its next action. History remains useful for provenance, audit, and recovery, but reliable decisions require an explicit account of what is currently valid. STATEFLOW provides one concrete framework for making this separation and for studying reliable state maintenance in more realistic settings.

AI USE STATEMENT

In this work, we have not used generative AI tools for any tasks with required disclosure. Additionally, we used generative AI tools solely to improve the readability of the manuscript, specifically for grammar correction, sentence-level restructuring, and wording refinement. We have reviewed all AI-assisted edits and verified that they did not alter the scientific content, claims, or conclusions of the paper. We take responsibility for the final content of this work.

REFERENCES

Bai, Y.; Lv, X.; Zhang, J.; Lyu, H.; Tang, J.; Huang, Z.; Du, Z.; Liu, X.; Zeng, A.; Hou, L.; Dong, Y.; Tang, J.; and Li, J. 2024. LongBench: A Bilingual, Multitask Benchmark for Long Context Understanding. ACL 2024.

Benjamini, Y.; and Hochberg, Y. 1995. Controlling the False Discovery Rate: A Practical and Powerful Approach to Multiple Testing. *JRSS B*, 57(1):289–300.

Buneman, P.; Chapman, A.; and Cheney, J. 2006. Provenance Management in Curated Databases. SIGMOD 2006.

Chao, H.; Bai, Y.; Sheng, R.; Li, T.; and Sun, Y. 2026. STALE: Can LLM Agents Know When Their Memories Are No Longer Valid? arXiv:2605.06527.

Cheney, J.; Chiticariu, L.; and Tan, W.-C. 2009. Provenance in Databases: Why, How, and Where. *Foundations and Trends in Databases*, 1(4):379–474.

Comanici, G.; et al. 2025. Gemini 2.5: Pushing the Frontier with Advanced Reasoning, Multimodality, Long Context, and Next Generation Agentic Capabilities. arXiv:2507.06261.

Cui, H.; Tang, Z.; Yao, Z.; Meng, F.; Ma, Q.; and Jia, W. 2026. MemTxn: A Transaction Boundary for Source-Supported Updates and Complete-State Recovery in Agent Memory. arXiv:2607.27834.

DeepSeek-AI. 2025. DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning. arXiv:2501.12948.

Du, Y.; Tian, M.; Ronanki, S.; Rongali, S.; Bodapati, S. B.; Galstyan, A.; Wells, A.; Schwartz, R.; Huerta, E. A.; and Peng, H. 2025. Context Length Alone Hurts LLM Performance Despite Perfect Retrieval. Findings of EMNLP 2025.

Efron, B. 1979. Bootstrap Methods: Another Look at the Jackknife. *Annals of Statistics*, 7(1):1–26.

Fan, X.; Liu, M.; Yang, R.; Ouyang, S.; and Han, J. 2026. Can Agent Memory Systems Track Evolving State? arXiv:2608.19652.

Gao, L.; Madaan, A.; Zhou, S.; Alon, U.; Liu, P.; Yang, Y.; Callan, J.; and Neubig, G. 2023. PAL: Program-Aided Language Models. ICML 2023.

Gou, Z.; Shao, Z.; Gong, Y.; Shen, Y.; Yang, Y.; Duan, N.; and Chen, W. 2024. CRITIC: Large Language Models Can Self-Correct with Tool-Interactive Critiquing. ICLR 2024.

Grattafiori, A.; et al. 2024. The Llama 3 Herd of Models. arXiv:2407.21783.

Hsieh, C.-P.; Sun, S.; Kriman, S.; Acharya, S.; Rekesh, D.; Jia, F.; Zhang, Y.; and Ginsburg, N. 2024. RULER: What's the Real Context Size of Your Long-Context Language Models? COLM 2024.

Hu, Y.; Wang, Y.; and McAuley, J. 2026. Evaluating Memory in LLM Agents via Incremental Multi-Turn Interactions. ICLR 2026.

Lesci, P.; Fujinuma, Y.; Hardalov, M.; Shang, C.; Benajiba, Y.; and Marquez, L. 2023. Diable: Efficient Dialogue State Tracking as Operations on Tables. Findings of ACL 2023.

Liu, N. F.; Lin, K.; Hewitt, J.; Paranjape, A.; Bevilacqua, M.; Petroni, F.; and Liang, P. 2024. Lost in the Middle: How Language Models Use Long Contexts. *TACL*, 12:157–173.

Lyu, Z.; Yang, K.; Kong, L.; and Klein, D. 2025. FactTrack: Time-Aware World State Tracking in Story Outlines. NAACL 2025, 2825–2848.

Maharana, A.; Lee, D.-H.; Tulyakov, S.; Bansal, M.; Barbieri, F.; and Fang, Y. 2024. Evaluating Very Long-Term Conversational Memory of LLM Agents. ACL 2024.

McNemar, Q. 1947. Note on the Sampling Error of the Difference Between Correlated Proportions or Percentages. *Psychometrika*, 12(2):153–157.

Meem, J.; Rashid, M.; Dong, Y.; and Hristidis, V. 2024. PAT-Questions: A Self-Updating Benchmark for Present-Anchored Temporal Question-Answering. Findings of ACL 2024.

Modarressi, A.; Deilamsalehy, H.; Dernoncourt, F.; Bui, T.; Rossi, R. A.; Yoon, S.; and Schuetze, H. 2025. NoLiMa: Long-Context Evaluation Beyond Literal Matching. ICML 2025.

Montgomery, L.; Lueders, C.; and Maalej, W. 2022. An Alternative Issue Tracking Dataset of Public Jira Repositories. MSR 2022.
Montgomery, L.; Lueders, C.; and Maalej, W. 2025. The Public Jira Dataset, Version 7.
OpenAI. 2025. OpenAI o3-mini System Card. Technical report.

Reddy, V.; and Challaram, S. 2026. Reliable Post-Retrieval Assembly for Agent Memory: Separating Evidence Extraction from Policy Execution. arXiv:2606.01435.
Schick, T.; Dwivedi-Yu, J.; Dessi, R.; Raileanu, R.; Lomeli, M.; Hambro, E.; Zettlemoyer, L.; Cancedda, N.; and Scialom, T. 2023. Toolformer: Language Models Can Teach Themselves to Use Tools. NeurIPS 2023.
Wu, D.; Wang, H.; Yu, W.; Zhang, Y.; Chang, K.-W.; and Yu, D. 2025. LongMemEval: Benchmarking Chat Assistants on Long-Term Interactive Memory. ICLR 2025.
Xu, W.; Liang, Z.; Mei, K.; Gao, H.; Tan, J.; and Zhang, Y. 2025. A-MEM: Agentic Memory for LLM Agents. NeurIPS 2025.
Yao, S.; Zhao, J.; Yu, D.; Du, N.; Shafran, I.; Narasimhan, K.; and Cao, Y. 2023. ReAct: Synergizing Reasoning and Acting in Language Models. ICLR 2023.
Yen, H.; Gao, T.; Hou, M.; Ding, K.; Fleischer, D.; Izsak, P.; Wasserblat, M.; and Chen, D. 2025. HELMET: How to Evaluate Long-Context Language Models Effectively and Thoroughly. ICLR 2025.
Zhan, Q.; Zhang, R.; Guo, S.; Zhao, L.; and Liu, Z. 2026. When Memory Becomes Authority: Benchmarking Authority Collapse at the Memory Consolidation Boundary. arXiv:2608.01679.
Zhu, Z.; Liao, Y.; Chen, Z.; Wang, Y.; Guan, Y.; Wang, Y.; and Wang, Y. 2025. EvolveBench: A Comprehensive Benchmark for Assessing Temporal Awareness in LLMs on Evolving Knowledge. ACL 2025.

A Comparison with Recent Evolving-State Benchmarks

Table 5 positions the benchmark against recent work that directly evaluates update-sensitive memory. The comparison summarizes the task-design axes most relevant to current-state reconstruction.

Benchmark	Primary state signal	Source authority	Executable structure	History-isolation probe	Scenario source
MemoryAgentBench / FactCon. (Hu et al., 2026)	newer conflicting facts	–	multi-hop fact consolidation	–	newly built task
STALE (Chao et al., 2026)	implicit invalidation	–	propagated stale dependencies	–	expert-validated scenarios
StateMemBench (Fan et al., 2026)	current vs. superseded state	–	standing rules + derived recomputation	–	symbolic event programs
AuthMemBench (Zhan et al., 2026)	source-authority preservation	explicit	–	–	paired authority variants
This work	authoritative current state	explicit	hierarchy + migration + fixed-point derived rules	matched StateOnly	factorized generator + public Jira

Table 5: Design-axis comparison with recent update-sensitive memory benchmarks. Dashes denote axes that are outside the cited benchmark's primary evaluation design.

B Benchmark Generation and Reproducibility

This appendix records the complete generator configuration used for the reported benchmark and is included to support reproducibility.

Generation pipeline.

1. Sample a Division–Suite–Product hierarchy at one of four joint-complexity levels.
2. Initialize SLA, Retention, ResponseWindow, SecurityTier, and AuditInterval values and record inherited versus explicit Product fields.
3. Generate source metadata and an authority policy, then construct chronological assignments, parent updates, Product overrides, migrations, and conditional-rule activations.
4. Ensure every retained instance contains at least one decision-relevant stale authoritative record; approximately half additionally contain a conflicting non-authoritative record.
5. Execute the sequence with the deterministic reference executor after every record, including derived-rule closure.
6. Generate a final query with three or four conjunctive predicates and retain instances with three to eight correct Products.
7. Reject instances with inconsistent rules, non-unique semantics, non-terminating derivations, or degenerate answer sets.

Generator parameter	L1	L2	L3	L4
Divisions	2–3	3–4	4–6	6–8
Suites	4–7	8–12	13–18	19–26
Products	12–20	28–44	48–72	80–106
Records	60–98	135–225	230–350	370–514
Conditional rules	1–2	2–3	3–5	5–8

Table 6: Generator ranges used for the four joint-complexity levels.

The operation mixture is direct assignment 0.28, parent update 0.22, Product override 0.20, migration 0.15, and conditional-rule-activating update 0.15. The generator injects a conflicting non-authoritative record with probability 0.50; query length is sampled uniformly between three and four predicates, followed by rejection sampling to preserve the three-to-eight answer-set range. Reproducibility seeds are 20270117 for development and 20270123 for held-out generation. Field vocabularies are SLA $\in$ {Gold, Silver, Bronze}, Retention $\in \{30, 45, 60, 90, 120, 180\}$ days, ResponseWindow $\in \{1, 2, 4, 8, 12, 24\}$ hours, SecurityTier $\in$ {Low, Medium, High, Critical}, and AuditInterval $\in \{7, 14, 30, 60, 90\}$ days.

C Prompt and Serialization Interfaces

This appendix makes the representation variables explicit so that the State-Only/StateFlow comparison is reproducible. The templates below reproduce the interfaces used in the reported runs.

StateFlow / format-matched decision interface.

```
SYSTEM: Answer only from the authoritative current state below.
RULES: <decision-relevant rules>
CURRENT STATE: <canonical serialized state>
QUERY: <final conjunctive query>
OUTPUT: <set of qualifying Products>
```

The serializer order is `Product`, `ActiveSuite`, `SLA`, `Retention`, `ResponseWindow`, `SecurityTier`, `AuditInterval`, with one resolved row per Product and no superseded/rejected alternatives.

LLM State Rewrite update interface.

```
SYSTEM: Maintain the complete authoritative current state.
RULES/POLICY: <governing rules and authority policy>
PREVIOUS STATE: <serialized state after record t-1>
NEW RECORD + SOURCE METADATA: <record t>
OUTPUT: <complete updated state in the fixed schema>
```

The final query call then uses the same decision interface as StateFlow. This design isolates the state-maintenance mechanism while holding the final-answer interface fixed.

D Corrected Pairwise-Test Summary

The exact-set comparisons use two-sided exact McNemar tests with Benjamini–Hochberg correction, as described in Section 5. The corrected model-level test counts are summarized below.

Contrast family	Significant / total	Interpretation
Among Neutral, Write-Back, and State-Only	0 / 21	diagnostic interventions statistically indistinguishable
Each intervention versus Full History	28 / 28	consistent improvement over Full History

Table 7: Summary of corrected model-level exact-set comparisons.

E Reasoning-Oriented Configuration Contrasts

Table 8 reports point differences for three family/effort-aligned configuration contrasts. They are used to characterize configuration-level patterns across the evaluated systems.

Contrast (first minus second)	Full	Neutral	Write-Back	State-Only	LLM State Rewrite	StateFlow
o3-mini high − medium	+8.3	+2.8	+0.8	+2.3	+4.5	+1.9
R1-Distill-Qwen-7B − Qwen2.5-7B-Instruct	+9.9	+1.9	+2.1	+0.5	+5.3	+2.7
R1-Distill-Llama-8B − Llama-3.1-8B-Instruct	+12.4	+3.6	+2.1	+5.4	+7.6	+2.9

Table 8: Exact-set accuracy differences (percentage points) for family/effort-aligned configuration contrasts. The separation is largest when models must reconstruct state from Full History.

F EXPANDED METRICS AND MATCHED ABLATIONS

Condition	Exact	Precision	Recall	F1	Over	Missing	Parse success
Full History	62.7	90.8	94.7	92.7	31.6	16.2	98.9
Neutral Replacement	89.4	97.0	97.8	97.4	8.7	8.1	99.5
Write-Back	91.1	97.5	98.0	97.7	7.2	7.4	99.6
State-Only	90.2	97.4	97.9	97.6	7.7	7.7	99.7
LLM State Rewrite	82.6	95.4	96.3	95.8	14.3	10.8	99.3
Format-Matched State-Only	94.8	98.9	99.2	99.0	3.4	5.1	99.8
StateFlow	94.8	98.9	99.2	99.0	3.4	5.1	99.8

Table 9: Expanded mechanism-isolating benchmark metrics (%) across all evaluated conditions.

Residual StateFlow error type	Count	Share
Query-predicate misapplication	188	51.5%
Omitted/extra Products	137	37.5%
Output-format/parse violation	40	11.0%
Total	365	100.0%

Table 10: Taxonomy of residual non-exact StateFlow cases across 7,000 model–instance evaluations.

Model configuration	Format-Matched State-Only	StateFlow
o3-mini high	97.0	97.0
Gemini 2.5 Pro	96.8	96.8
o3-mini medium	95.1	95.1
R1-Distill-Qwen-7B	96.3	96.3
R1-Distill-Llama-8B	93.8	93.8
Qwen2.5-7B-Instruct	93.6	93.6
Llama-3.1-8B-Instruct	90.9	90.9
Macro average	94.8	94.8

Table 11: Format-matched State-Only ablation. Equality follows from verified prompt identity under deterministic decoding.

G WORKED STATEFLOW TRACE

Table 12 gives an illustrative trace of the mechanisms that distinguish STATEFLOW from a simple last-write-wins store.

Step	Record / parsed operation	Authority	Applicable	State effect
Initial	P17 in S1; P17 has explicit SecurityTier=High; Retention is inherited	–	–	P17: Suite=S1, Retention=30, SecurityTier=High
t_1	Authorized Division update: Retention $\leftarrow$ 45	yes	yes	P17 inherits Retention=45
t_2	Unauthorized Product claim: P17 Retention $\leftarrow$ 7	no	–	Rejected; visible in history but state unchanged
t_3	Authorized migration: P17 S1 $\rightarrow$ S2	yes	yes	Membership becomes S2; explicit SecurityTier=High persists; non-overridden Retention refreshes from S2 to 90
t_4	Authorized Suite S2 update: Retention $\leftarrow$ 120	yes	yes	P17 inherits Retention=120
t_5	Derived rule: if Retention$\geq$90 and SecurityTier=High, AuditInterval$\leftarrow$14d	–	enabled	Fixed-point execution derives AuditInterval=14d
Final	Serialize canonical state	–	–	P17: Suite=S2, Retention=120, SecurityTier=High, AuditInterval=14d; stale/rejected alternatives omitted

Table 12: Illustrative end-to-end STATEFLOW trace covering non-authority, migration, inheritance, persistent override, derived-rule execution, and canonical serialization.