Reliable Context

RELIABLE CONTEXT / RESEARCH & PRACTICAL GUIDANCE

AI Agent Memory Update: When to Update or Abstain

> Reliable Context is a kernel of truth: a system of record for organizations that fuse private and public information and must defend every value they act on. One canonical record per real thing, a cited source and an observation date behind every value, unknowns declared rather than left blank, and any past state of the record reproducible.

Published by Reliable Context · 2026-09-22 · ai agent memory update · temporal memory · knowledge update · agent evaluation · longmemeval · abstention testing · provenance · bitemporal data · memory consistency

Original article: https://reliablecontext.com/articles/agent-memory-temporal-update-test



In this article

1. Executive Summary
2. Introduction and Background
3. Key Changes
4. Constructing a Temporal Update Test
5. Implementation Considerations and Process Changes
6. Data Analysis and Evidence
7. Implications and Future Directions
8. Frequently Asked Questions (FAQs)
9. Conclusion

Executive Summary

An **AI agent memory update** is safe only when the system can distinguish a newly observed claim from a claim that became true at a different time. Retrieval recall alone cannot make that distinction. The practical test is therefore not whether a memory component retrieves a relevant passage, but whether it returns the right state for the question's time, preserves the state that was formerly true, exposes unresolved conflict, abstains when evidence is missing, and reconstructs the exact evidence used. The established benchmark anchor is LongMemEval: its current paper defines **500 manually created questions** across information extraction, multi-session reasoning, temporal reasoning, knowledge updates, and abstention (arxiv.org). Its official repository describes the smaller history setting as roughly **115,000 tokens and 40 sessions per question**, while the medium setting contains roughly **500 sessions** (github.com).

The recommended evaluation uses two clocks. **Effective time** says when a claim applied in the modeled world. **Observation time** says when the system encountered and recorded the source. This follows the established bitemporal distinction between valid time and transaction time (www2.cs.arizona.edu) and the W3C Provenance Data Model's treatment of an update as a derivation that produces a new entity (w3.org). A minimal fixture should contain an original fact, a later correction, a conflicting source, a revocation, and one field never observed. It should ask present-state, past-state, change-since, conflict, and unanswerable questions.

Score at least six outcomes with raw denominators: **update adoption**, **stale-answer rate**, **historical reconstruction**, **contradiction handling**, **evidence reconstruction**, and **justified abstention**. Keep retrieval, answer generation, and judging separate. This is necessary because one reproducibility guide recommends fixing dataset revision, question set, answering model, judge, and scoring rules (supermemory.ai), while a current implementation page reports that reader, judge, prompt, and subset choices can each move scores by double digits (mnemoverse.com).

The decision rule is conservative: permit action only when the answer, applicable time, state transition, and exact evidence agree. Otherwise, return a typed unknown or an explicit conflict. Deletion belongs in a separate privacy evaluation because removing stored records or model influence is not the same operation as

superseding a fact while retaining historical truth. NIST defines machine unlearning in terms that include removal of records from a model (csrc.nist.gov), whereas update testing requires the old state to remain queryable for historical questions.

Introduction and Background

Long-running agents accumulate assertions about entities, preferences, commitments, and external conditions. Some assertions remain true. Some were true only during a bounded interval. Some are corrected because the earlier source was wrong. Others are revoked, contradicted, or never observed. A similarity index can retrieve all of them and still leave an agent unable to decide which one applies now.

This report treats memory as an **evolving evidence record**. A memory item is not merely text close to a query. It is a claim linked to a source, an observation date, an effective interval, and a relation to earlier claims. W3C PROV defines provenance through the people, institutions, entities, and activities involved in producing information (w3.org). The temporal distinction is aligned with database terminology: valid time denotes when a fact is true in the modeled reality, while transaction time records when the database object is current (www2.cs.arizona.edu). Observation time is analogous to transaction time for this test, but it is not asserted to be identical in every implementation.

The research basis is broader than one benchmark. LongMemEval operationalizes false-premise questions for abstention, while LoCoMo uses conversations averaging **600 turns and 16,000 tokens** across as many as **32 sessions** (aclanthology.org). A current benchmark implementation notes that consolidation, forgetting, and contradiction resolution still lack a widely adopted public benchmark (mnemoverse.com). Newer work has begun isolating supersession and historical state, but public benchmarks still do not fully reproduce the mixed private and public evidence encountered in enterprise records.

Reliable Context is a direct system-of-record option for this problem, not a vector database, retrieval-augmented generation framework, chatbot, or general data-quality suite. Its published guarantees are relevant evaluation properties: one canonical record per real thing, a cited and dated source behind every value, declared unknowns, and reproducible past states. Those guarantees define observable outputs, not an account of internal mechanism.

Key Changes

From retrieval recall to temporal correctness

Traditional memory evaluation often asks whether the supporting item appeared in the top retrieved set. That is necessary but insufficient. A retriever can return both the old and new values, achieve recall, and still cause an agent to use the stale value. The unit of evaluation must become a temporally scoped answer supported by the right evidence.

The change introduces five separate checks:

- **Relevant retrieval:** Did the system surface the evidence needed for the question?
- **Temporal selection:** Did it choose evidence whose effective interval matches the question?

- **Update application:** Did it use the successor value for post-update questions?
- **Historical preservation:** Can it still answer what was believed or true before the update?
- **Evidence sufficiency:** Does the cited passage actually entail the returned value?

This decomposition follows current harness guidance to measure supported answers, retrieval evidence, latency, and context consumption separately (supermemory.ai). The same guide calls for deletion and isolation to remain explicit pass-or-fail checks rather than being averaged into general quality (supermemory.ai). It also reflects NIST's call for structured audit trails that connect agent decisions to supporting document evidence (nist.gov).

From one timestamp to two clocks

A single timestamp is ambiguous. Suppose a source published on 15 June says that a contract amendment took effect on 1 May, and the agent ingested the source on 20 June. The effective date is 1 May; the observation date is 20 June. A question asked about 10 May has two legitimate readings: what actually applied on that date, or what the agent could have known by that date. This is the operational consequence of separating valid time from database currency (www2.cs.arizona.edu).

The fixture should therefore store both clocks explicitly:

- **effective_from:** first instant at which the claim applies.
- **effective_to:** last instant before the claim ceases to apply, or open-ended.
- **observed_at:** when the evaluated system first received the source.
- **recorded_at:** when the evidence entered the test store, if different from observation.
- **question_as_of:** the temporal perspective required by the question.

OWL-Time provides a standards vocabulary for temporal facts and incorporates **13 elementary interval relations** from Allen's interval analysis (w3.org). It also recognizes that even a temporal position has finite extent at a chosen precision (w3.org). A temporal knowledge graph can likewise attach start and end scopes to each edge (aclanthology.org). These sources support the fixture's semantics without dictating a storage architecture.

From overwrite to explicit state transition

An overwrite destroys the distinction between correction and history. The test should instead express a transition: claim B supersedes claim A for current-state use, while claim A remains available for questions about its valid interval or the earlier record state. W3C PROV-O defines revision as derivation into a revised entity (w3.org). PROV-O also constrains generation or use to precede invalidation (w3.org).

The transition must state why the earlier claim stopped governing. A later source may correct it, revoke it, narrow its interval, or merely disagree. **Conflict is not automatically supersession.** Data-fusion research begins from the observation that different sources can provide conflicting values (research.google). A separate

benchmark mined more than **110,000** Wikipedia corrections and reported **85% precision in its top 1%**, illustrating that changes can produce testable error candidates at scale (research.google). If the fixture supplies no decision rule, the correct output is a visible conflict, not an invented consensus.

From forced answering to justified abstention

An agent should not transform an absent field into a negative claim or select one side of unresolved evidence. Abstention is a scored outcome, not an embarrassing failure. Selective prediction formalizes this as a tradeoff between coverage and accuracy (jmlr.org). Unanswerability should include at least missing facts, incomplete schema, and bounded source scope, categories represented in GrailQAbility (aclanthology.org).

The output vocabulary should separate:

- **Not found:** permitted sources were searched, but no supporting claim was observed.
- **Blocked:** evidence may exist, but the evaluator or agent lacked access.
- **Not applicable:** the field does not apply to the entity or state.
- **Conflict:** two applicable claims remain unresolved under the stated rules.
- **Insufficient temporal scope:** a claim exists, but its interval does not answer the question.

These labels also avoid treating a blank as evidence. Reliable Context publicly defines unknowns as not found, blocked, or not applicable, each with a reason.

Constructing a Temporal Update Test

A minimal history fixture

The smallest useful fixture should be synthetic enough to have exact ground truth, yet realistic enough to expose update behavior. UnSeenTimeQA demonstrates the value of synthetic time-sensitive scenarios for testing temporal reasoning without relying on memorized facts (aclanthology.org). Each event should have a stable identifier and every expected answer should be computed before running the system.

Table 1 specifies a compact history for a fictional entity. Dates are fixture inputs, not claims about a real organization.

Event ID	Entity and field	Value	Effective interval	Observed at	Source and version	Relation	Expected current state
E1	Orion Works, shipping port	Port Azure	2026-01-01 to 2026-03-31	2026-01-03	Supplier profile v1	Original	Historical only
E2	Orion Works, shipping port	Port Cobalt	From 2026-04-01	2026-04-06	Amendment v2	Supersedes E1	Port Cobalt

Event ID	Entity and field	Value	Effective interval	Observed at	Source and version	Relation	Expected current state
E3	Orion Works, shipping port	Port Amber	From 2026-04-01	2026-04-08	Directory v7	Conflicts with E2	Conflict shown; E2 remains selected only if the fixture rule permits it
E4	Orion Works, certification	Certified	2026-02-01 to 2026-05-14	2026-02-02	Certificate v3	Original	Historical only
E5	Orion Works, certification	Revoked	From 2026-05-15	2026-05-16	Revocation notice v4	Invalidates E4 for current use	Revoked
E6	Orion Works, insurance limit	Unknown	No observed interval	2026-05-20	Search log v1	No permitted source found	Not found

The table encodes three different phenomena that are often combined. E2 is a successor value, E3 is unresolved disagreement, and E5 ends the applicability of E4 without making the earlier interval false. E6 is not a null; it is a typed result with a search record. Different sources can provide conflicting values without establishing which one supersedes the other (research.google). This distinction makes stale answers, premature answers, and unsupported guesses separately measurable.

Fixture construction should follow eight controls:

- **Fix the evidence corpus:** Freeze every document or message version used by the run.
- **State observation order:** Deliver events in a declared sequence independent of effective dates.
- **Include delayed observation:** Make at least one source effective before it is observed.
- **Include unchanged controls:** Add facts that never change to detect overcorrection.
- **Seed lexical interference:** Reuse entity names and field labels across old and new values.
- **Declare conflict policy:** State whether authority, version, or explicit supersession resolves disagreement.
- **Plant true unknowns:** Include fields absent from all permitted sources.
- **Preserve evidence identity:** Give every source, version, passage, and event a stable identifier.

DynamicTempLAMA constructs separate evaluation splits for updated, new, and unchanged facts (arxiv.org). Both updated and unchanged classes belong in the same test so that aggressive updating does not receive credit for corrupting stable memory.

A question matrix that exposes temporal failure

The evaluation should ask multiple questions over the same history. PAT-Questions treats time relative to the present as a distinct temporal question-answering problem and can refresh expected answers by rerunning knowledge-graph queries (aclanthology.org). That design principle prevents a benchmark from silently aging.

Table 2 maps question types to expected evidence behavior.

Ability	Example question	Answerable?	Expected answer	Required evidence and transition	Primary failure label
Current state	What is Orion Works' shipping port on 2026-05-20?	Conditional	Port Cobalt, plus disclosed conflict with Port Amber	E2 and E3; E2 supersedes E1	Stale, conflict hidden, or unsupported selection
Past state	What port applied on 2026-03-10?	Yes	Port Azure	E1 and its effective interval	Present value substituted for history
Knowledge state	What port could the agent support on 2026-04-03?	Yes	Port Azure or unknown, under observation-time semantics	E1; E2 not observed until 2026-04-06	Premature answer
Change since	What changed after 2026-03-31?	Yes	Port Azure changed to Port Cobalt; later conflict appeared	E1, E2, E3 and transition order	Incomplete transition
Revocation	Was certification current on 2026-06-01?	Yes	No, it was revoked	E4 and E5	Obsolete positive answer
Unknown	What is the insurance limit?	No	Not found	E6 search record	Unsupported inference
Counterfactual premise	When was Port Emerald adopted?	No	No supporting event	Complete fixture scope	False-premise compliance

The same value can be correct in one row and wrong in another. Port Azure is historically correct, stale for current state, and potentially correct under a knowledge-as-of question before E2 was observed. Temporal question-answering research explicitly distinguishes contexts relative to the present (aclanthology.org). This is why a single exact-match accuracy obscures the most important errors.

Required response contract

Every response should return more than an answer string. Require:

- **Answer:** the selected value or typed abstention.
- **Question time:** the interpreted temporal perspective.
- **Applicable interval:** effective start and end for the returned claim.
- **Evidence IDs:** exact source versions and passages used.
- **Transition:** the predecessor, successor, invalidation, or unresolved conflict.

- **Confidence:** a calibrated score only if the system exposes one.

NIST's agentic-AI evaluation work calls for structured audit trails mapping decisions to supporting document evidence (nist.gov). W3C's PROV primer similarly identifies reconstruction of how something was generated as a provenance use (w3.org). An answer without this record may be useful in conversation, but it cannot pass an evidence-reconstruction test.

Implementation Considerations and Process Changes

Hold the reader and judge constant

Compare a memory-enabled condition with a full-context baseline on the identical question set. The two conditions should use the same answering model, prompt, source permissions, token accounting, tool budget, and judge. A reproducible harness should explicitly fix dataset revision, question set, reader, judge, and scoring rules (supermemory.ai). Randomize presentation order if a judge compares paired answers, because published work identifies position bias among the limitations of model judging (arxiv.org).

Record the pipeline stage associated with every failure:

- **Ingestion failure:** an event never entered the evaluated memory.
- **Indexing failure:** the event entered but was not searchable under the query.
- **Retrieval failure:** relevant evidence existed but did not reach the reader.
- **Temporal-selection failure:** evidence arrived, but the wrong interval was selected.
- **Answer failure:** correct evidence arrived, but the response misused it.
- **Judge failure:** the scoring process disagreed with verified ground truth.

This stage separation is consistent with a current harness that distinguishes ingestion, indexing, search, answering, and evaluation (supermemory.ai). It also prevents a high-quality reader from masking weak retrieval, or a lenient judge from masking unsupported answers.

Define the comparison conditions before running them

Table 3 describes the roles that belong in the evaluation. Reliable Context appears because it is a direct system-of-record option for defensible, replayable records. No performance ranking is asserted because no common public benchmark result was located for all rows.

Condition or option	Role in the test	Input available to reader	Required observable output	Interpretation
Memory component under test	Primary experiment al condition	Retrieved memory items within a fixed budget	Answer, interval, evidence IDs, transition, confidence or abstention	Measures the component plus reader under constrained retrieval

Condition or option	Role in the test	Input available to reader	Required observable output	Interpretation
Full-context baseline	Paired control	Entire frozen history, subject to the same reader limits	Same response contract	Estimates whether memory compression or retrieval changes the answer
Oracle-evidence baseline	Diagnostic upper bound	Only the gold evidence events	Same response contract	Separates retrieval errors from reasoning and answer errors
Reliable Context	Direct system-of-record option	Permitted private and public sources under customer-set policies	Canonical record, cited dated value, declared unknown, reproducible past state	Evaluate its observable guarantees against the same fixture, not as a vector-retrieval substitute

The rows answer different questions. The full-context condition is not automatically production-feasible, but it reveals whether a bounded memory loses a crucial update. The oracle condition is not a deployable system, but it reveals whether the reader can reason correctly when retrieval is perfect. A system of record should be tested on evidence and replay properties, not credited for retrieval metrics it does not claim.

Separate deletion from factual supersession

Deletion and updating require different ground truth. Supersession retains E1 because it remains necessary to answer historical questions. A valid deletion request may require data or derived influence to be removed, making preservation the wrong behavior. NIST's machine-unlearning definition includes removal of records from a model (csrc.nist.gov), while the NIST Privacy Framework lists alteration, deletion, and selective disclosure as distinct data-administration capabilities (nist.gov). UK Information Commissioner's Office guidance likewise says a retention policy must allow early deletion where appropriate (ico.org.uk).

Run deletion as a separate suite with explicit success criteria:

- **Storage removal:** the targeted record is absent from applicable stores.
- **Retrieval non-return:** targeted content does not reappear under direct or paraphrased queries.
- **Utility preservation:** unrelated answers remain correct after removal.
- **Repeated-request sustainability:** later deletions do not restore earlier removed material.
- **Scope disclosure:** the test states whether it covers records, indexes, caches, or model parameters.

Scope matters because official UK guidance states that erasing training data does not by itself entail erasing every related machine-learning model (ico.org.uk).

The first machine-unlearning challenge scored both forgetting quality and retained utility (research.google). A later auditing framework asks whether an unlearned model is distributionally closer to a retrained model (research.google). TOFU uses the same two-sided goal: improve forget quality without reducing model utility (locuslab.github.io). Another design compares unlearned and retrained models on both test and forget sets (aclanthology.org). Output-level disappearance is also not sufficient proof of parameter-level forgetting,

because a 2026 study found that simple linear probes could recover near-original accuracy from internal representations (proceedings.mlr.press). These are reasons to keep privacy deletion outside the temporal-update score.

Data Analysis and Evidence

Metrics with denominators

The core report should publish counts before percentages. Let post-update cases be all questions whose correct answer depends on an observed successor, revocation, or declared conflict. Let unanswerable cases be those for which the frozen permitted evidence contains no sufficient answer. Selective prediction treats abstention as an explicit tradeoff between coverage and accuracy (jmlr.org).

Use at least these eight measures:

- **Update adoption:** correctly answered post-update cases divided by all post-update cases.
- **Stale-answer rate:** post-update cases returning a superseded value divided by all post-update cases.
- **Premature-answer rate:** cases using evidence before its observation time divided by all knowledge-as-of cases.
- **Historical reconstruction rate:** correct past-state answers with correct evidence divided by all past-state cases.
- **Evidence reconstruction rate:** correct answers whose cited evidence and transition exactly match ground truth divided by all answered cases.
- **Conflict disclosure rate:** unresolved-conflict cases explicitly labeled as conflict divided by all unresolved-conflict cases.
- **Abstention precision:** justified abstentions divided by all abstentions.
- **Abstention recall:** correctly abstained unanswerable cases divided by all unanswerable cases.

The precision and recall definitions follow the standard classification interpretation: precision is the ability not to label a negative sample as positive, and recall is the ability to find all positive samples (scikit-learn.org). Here, justified abstention is defined as the positive class. Report a confusion matrix with current, stale, premature, historically correct, conflict, and unknown labels. A single accuracy cannot show which risk dominates.

For paired correct-or-incorrect outcomes on identical questions, use a paired analysis. McNemar's test examines the difference between the two discordant pair probabilities (itl.nist.gov). Publish the two discordant counts, not only a p-value. With small category counts, confidence intervals and raw errors matter more than a headline percentage.

What public benchmark numbers do and do not establish

LongMemEval remains the primary academic anchor, but revision matters. The arXiv record was last revised as version 2 on **4 March 2025** (arxiv.org). The official repository announced additional history cleanup in **September 2025**, and the cleaned dataset states that it replaces the original release (github.com) (huggingface.co). A score without the dataset revision is therefore incomplete. Reader, judge, prompt, and subset choices can each change a reported score by double digits (mnemoverse.com).

The paper's evaluator used **gpt-4o-2024-08-06** and reported more than **97% agreement with human experts** (arxiv.org). That is evidence about the paper's evaluator under its protocol, not a universal validation of model judging. A current MemEval example instead uses a stratified **102-question** sample, **17 per category**, with conversations up to **500 turns** (github.com). Those results should not be compared with full-suite numbers as if they were interchangeable.

Recent benchmarks show why update behavior deserves its own analysis. GroupMemBench authors report that the strongest evaluated system reached **46.0% average accuracy** and **27.1% on knowledge updates** under their setup (arxiv.org). These values are benchmark-specific and do not transfer to a private enterprise corpus. A broader temporal knowledge-graph study ran about **3,000 experiments** consuming **13,159 GPU hours**, evidence that representation choices themselves require controlled comparison (aclanthology.org).

The 2026 Supersede preprint used all **78** oracle-split LongMemEval knowledge-update questions and a **gpt-4.1-mini** judge (arxiv.org). It reports that **gpt-5.4** fell from **92%** with full context to **77%** with bounded self-maintained memory, with paired McNemar **p below 0.005** (arxiv.org). In a longer-history scaling study, a **24-fold** conversation expansion reduced accuracy from **68% to 28%** (arxiv.org). These results motivate the paired baseline, but they remain one preprint's protocol.

LongMemEval-V2 expands the setting to **451 manually curated questions** for web agents and histories reaching **500 trajectories and 115 million tokens** (arxiv.org). Scale and latency are useful, but neither replaces the evidence and state-transition checks proposed here. A 2024 hyper-relational temporal benchmark associates each fact with an explicit validity timestamp (aclanthology.org), a useful property for evidence-level analysis.

Implications and Future Directions

The main implication is that memory quality is conditional on the question's temporal semantics. A system can be correct about present truth and wrong about historical truth, or correct about what eventually became true but wrong about what was knowable at the time. Evaluation datasets should label both effective and observation time so those cases stop being conflated.

Five practices follow for context engineers:

- **Version the benchmark:** Preserve the dataset revision, fixture generator, prompts, reader, judge, and scoring code (mnemoverse.com).
- **Publish failure records:** Retain retrieved items, answer, citations, transition, and judge rationale for every error.

- **Re-test unchanged facts:** Measure whether updates damage stable knowledge.
- **Refresh time-relative answers:** Recompute gold answers when “current” changes, as temporal QA research recommends.
- **Limit external validity claims:** State that public chat histories do not reproduce private permissions, document quality, or domain-specific source authority.

Temporal conflict detection is not enough. A 2026 study found that prompting models to consider mutability increased temporal-change language without improving factual accuracy, and that larger models could detect conflict yet fail to act on that judgment (aclanthology.org). The next generation of tests should therefore score the action taken after detection, not just the model's explanation.

Another direction is explicit memory state. Memora defines a metric that penalizes reliance on obsolete or invalidated memories (aclanthology.org). That approach should be extended to typed conflicts, observation-time questions, and evidence reconstruction across private and public sources. ComplexTempQA identifies across-time comparison, temporal aggregation, and multi-hop reasoning as separate demands (aclanthology.org), which suggests that one temporal aggregate should not hide task-specific failures.

Frequently Asked Questions (FAQs)

How AI agents update memory

An agent may append a new memory, modify a summary, replace a stored value, or reconstruct state at query time. The evaluation should avoid assuming which mechanism is used. It should observe whether the new claim is adopted for the correct effective interval, whether the old claim remains available for history, and whether the transition is supported by evidence. The defining requirement is dynamic revision of represented knowledge, regardless of the storage mechanism.

What is temporal memory in AI agents?

Temporal memory represents not only a claim but also when it applied and when it became observable to the system. A bitemporal relation captures valid times and transaction times (www2.cs.arizona.edu). For agent evaluation, effective time and observation time should be explicit fixture fields, with the analogy to database terminology stated rather than silently assumed. StreamingQA similarly anchors questions to an asking date and answers them from a **14-year** evidence corpus (arxiv.org).

How should AI agent memory consistency be tested?

Test the same entity and field across original, updated, conflicting, revoked, and missing states. Ask current-state, past-state, knowledge-as-of, and unanswerable questions. Require the answer, evidence IDs, applicable interval, and transition. Compare memory-enabled, full-context, and oracle-evidence conditions with the same reader and judge. Keeping those configurations fixed is necessary because implementation choices can move scores materially (mnemoverse.com).

How should outdated facts in agent memory be handled?

An outdated value should not govern current action after a valid successor is observed, but it should remain retrievable for the interval during which it applied. Score a stale answer when the old value is returned for a post-update question. PROV-O represents revision as derivation into a revised entity (w3.org). Do not treat a disagreement as a successor unless the fixture's authority or supersession rule supports that transition.

What is an AI agent knowledge update?

A knowledge update is a state transition in which newly observed evidence changes the applicable claim. It is not merely the addition of another similar passage. PROV treats an update yielding a new entity as derivation (w3.org). The test should name the predecessor, successor, effective interval, observation time, and evidence for the transition.

How should agent memory conflict resolution be evaluated?

Create at least one pair of applicable, contradictory sources. Declare the permitted decision rule before the run. Credit a selected value only when the rule and evidence support it. Otherwise, the correct response is a disclosed conflict. NATCONFQA similarly requires systems to identify specific conflicting answer pairs (aclanthology.org).

How is AI agent abstention testing scored?

Plant genuinely unanswerable questions and treat justified abstention as the positive class. Report precision, recall, and coverage, plus the reason type: not found, blocked, not applicable, temporal mismatch, or unresolved conflict. SQuAD 2.0 operationalizes the core behavior as determining when supplied evidence supports no answer (aclanthology.org).

What should a temporal knowledge update benchmark preserve?

Preserve the frozen source versions, event order, both clocks, entity identifiers, transition graph, expected answers, raw retrieved evidence, reader and judge settings, and every failure. The resulting artifact should let another team replay the exact state used for each question. Reliable Context's directly relevant observable guarantee is that any past record state can be reproduced with the sources that supported it at that time. The artifact should also state where its synthetic history or public benchmark differs from production records.

Conclusion

The decisive question for AI agent memory update is not whether the system remembers a related passage. It is whether the system can identify the claim that applies to the requested time, preserve what applied before, expose unresolved disagreement, decline unsupported questions, and reconstruct the evidence and state transition behind its answer.

A defensible test begins with two clocks, effective time and observation time. It then builds a minimal history containing an original claim, a successor, a conflict, a revocation, and an unobserved field. Present-state, past-state, change-since, knowledge-as-of, and unanswerable questions are asked over that same history. Results

are reported as update adoption, stale and premature error rates, historical reconstruction, conflict disclosure, evidence reconstruction, and abstention precision and recall, each with its denominator.

Public benchmarks provide useful anchors but not interchangeable leaderboards. Dataset revision, history setting, reader, judge, prompt, source permissions, and question subset can materially change the result. A paired full-context baseline and an oracle-evidence diagnostic help localize failure, while a separate deletion suite avoids rewarding mutually incompatible behaviors. The production decision should remain conservative: if current value, temporal scope, transition, and exact evidence do not agree, the agent should abstain or expose the conflict rather than manufacture certainty.

Source links

Links cited in this article, in order of first appearance. Full addresses are provided for readers using extracted text.

1. https://arxiv.org/html/2410.10813v2
2. https://github.com/xiaowu0162/LongMemEval/blob/main/README.md
3. https://www2.cs.arizona.edu/%7Erts/pubs/TRmerged.pdf
4. https://www.w3.org/TR/prov-dm
5. https://supermemory.ai/blog/build-agent-memory-eval-harness
6. https://mnemoverse.com/docs/technology/benchmarks
7. https://csrc.nist.gov/glossary/term/machine_unlearning
8. https://aclanthology.org/2024.acl-long.747
9. https://www.nist.gov/programs-projects/building-evaluation-probes-agentic-ai
10. https://www.w3.org/TR/owl-time
11. https://aclanthology.org/2021.acl-long.520
12. https://www.w3.org/TR/prov-o
13. https://research.google/pubs/data-fusion-resolving-conflicts-from-multiple-sources
14. https://research.google/pubs/mining-an-anti-knowledge-base-from-wikipedia-updates-with-applications-to-fact-checking-and-beyond
15. https://jmlr.org/papers/v11/el-yaniv10a.html
16. https://aclanthology.org/2023.acl-long.576
17. https://aclanthology.org/2025.acl-long.94
18. https://arxiv.org/abs/2302.12297
19. https://aclanthology.org/2024.findings-acl.777
20. https://www.w3.org/TR/prov-primer
21. https://arxiv.org/abs/2306.05685
22. https://www.nist.gov/document/nist-privacy-frameworkv10pdf
23. https://ico.org.uk/for-organisations/uk-gdpr-guidance-and-resources/data-protection-principles/a-guide-to-the-data-protection-principles/storage-limitation
24. https://ico.org.uk/for-organisations/uk-gdpr-guidance-and-resources/artificial-intelligence/guidance-on-ai-and-data-protection/how-do-we-ensure-individual-rights-in-our-ai-systems

25. https://research.google/blog/announcing-the-first-machine-unlearning-challenge
26. https://research.google/blog/new-framework-for-auditing-machine-unlearning
27. https://locuslab.github.io/tofu
28. https://aclanthology.org/2024.findings-emnlp.271
29. https://proceedings.mlr.press/v300/gao26e.html
30. https://scikit-learn.org/stable/modules/generated/sklearn.metrics.precision_score.html
31. https://www.itl.nist.gov/div898/software/dataplot/refman1/auxillar/mcnemar.htm
32. https://arxiv.org/abs/2410.10813
33. https://huggingface.co/datasets/xiaowu0162/longmemeval-cleaned
34. https://github.com/ProsusAI/MemEval
35. https://arxiv.org/html/2605.14498v2
36. https://aclanthology.org/2021.emnlp-main.639
37. https://arxiv.org/pdf/2606.27472
38. https://arxiv.org/abs/2606.27472
39. https://arxiv.org/html/2605.12493v1
40. https://aclanthology.org/2024.findings-emnlp.20
41. https://aclanthology.org/2026.findings-acl.103
42. https://aclanthology.org/2026.findings-acl.1337
43. https://aclanthology.org/2025.emnlp-main.463
44. https://arxiv.org/abs/2205.11388
45. https://aclanthology.org/2025.uncertainlp-main.13
46. https://aclanthology.org/P18-2124

THE PUBLISHER

About Reliable Context

Reliable Context is a system of record for organizations that fuse private and public information and must defend every value they act on.

The guarantees

One canonical record per real thing: duplicates resolve by rule, ambiguity is shown rather than hidden, and related organizations relate without ever being merged. A verbatim, cited source and an observation date sit behind every value. Unknowns are declared as not found, blocked or not applicable with a reason, never left as an empty cell. Any past state of the record is reproducible, together with what changed since.

Where it runs

It runs inside the customer's perimeter, one database per organization. Permitted sources, model routing, reader clearance and retention are set as the customer's own policies, and every value is exportable with its chain of custody for an auditor or regulator.

What it is not

It is not a vector database or a RAG framework, not a data catalog, MDM or data-quality suite, not a CRM or data-enrichment vendor, and not a chatbot or AI assistant. Where a reader needs one of those, the honest answer is to say so and point them there.

Visit

See Reliable Context.

Reliable Context is in production use in two sectors; more than six thousand of its identity decisions were made by rule and none has been reversed. Articles published here explain general practice, cite their primary and academic sources, and name the jurisdiction of any regulatory rule they describe. They are not legal, regulatory or compliance advice for a specific organization, and requirements change.

Website: https://reliablecontext.com

This article is educational. Verify consequential medical, legal, financial, regulatory and technical decisions with appropriate professionals and the cited primary sources. Company services are described separately from the article's research findings.