SLIDETRACE: GOVERNED MEMORY FOR CROSS-PROJECT SLIDE PERSONALIZATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Cross-project slide personalization requires an agent to carry reusable user preferences across projects while preserving their scope and evolution over time. Raw feedback histories leave these properties implicit, making it difficult to determine which preferences remain applicable as interactions accumulate. We introduce SLIDETRACE, a governed memory framework that represents reusable feedback as atomic, scoped preference claims and maintains them across project working memory and canonical long-term memory. Semantic relation proposals and deterministic validation govern how claims are updated, related, and consolidated across projects. We evaluate SlideTrace against four memory systems and a no-memory reference under a shared generator. After 20 sequential training projects, frozen memory states are evaluated on 15 held-out tasks. SlideTrace achieves the highest overall Score under both GPT-5.4 and Kimi-K3, exceeding the strongest baseline by 0.38 and 0.51 points, with the clearest gains in requirement realization. Ablations further support relation-governed canonicalization and structured memory packages. These findings show that cross-project personalization benefits from governing reusable preference state rather than merely retaining interaction history. Code and data are available in our anonymous repository at `https://anonymous.4open.science/r/slidetrace-anonymous-artifact-1315`.

1 INTRODUCTION

Presentation agents increasingly combine content planning, editable slide generation, rendering, and visual revision (Zheng et al., 2025; 2026). Yet successful revision within one project does not ensure personalization across projects. An agent may correct generic titles, competing numerical highlights, or weak evidence in one deck, then repeat the same pattern on the next task, requiring the user to restate preferences that were already expressed and successfully realized. At the same time, interaction histories contain local edits, project facts, rendering diagnostics, and abandoned attempts that should not automatically shape future work. Cross-project personalization therefore requires deciding which feedback is reusable and how it should persist beyond the project in which it was observed.

Two decisions are easily conflated: *what should affect the current project*, and *what should persist across projects*. "Use conclusion-first titles" may carry across projects, while "keep only conclusions in this board version" is specific to an audience or project. Later feedback may refine, qualify, replace, or withdraw an earlier preference. Raw interaction history preserves these statements but leaves their applicability, relationships, and current status implicit. Semantic retrieval can identify relevant records (Lewis et al., 2020; Karpukhin et al., 2020), but relevance alone does not establish whether a preference still applies, is valid for the current scope, or has been realized in the generated artifact. Longer contexts do not resolve these distinctions (Liu et al., 2024). Our central premise is that cross-project personalization requires governing reusable preference state rather than retrieving an undifferentiated record of prior interactions.

We introduce SLIDETRACE (Figure 1), a preference-claim memory layer for cross-project slide personalization that connects to an existing generator. SlideTrace represents reusable feedback as atomic, scoped preference claims and maintains them at two timescales. Long-term claims initialize a project working memory, where new feedback can reinforce, refine, replace, or qualify existing

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.

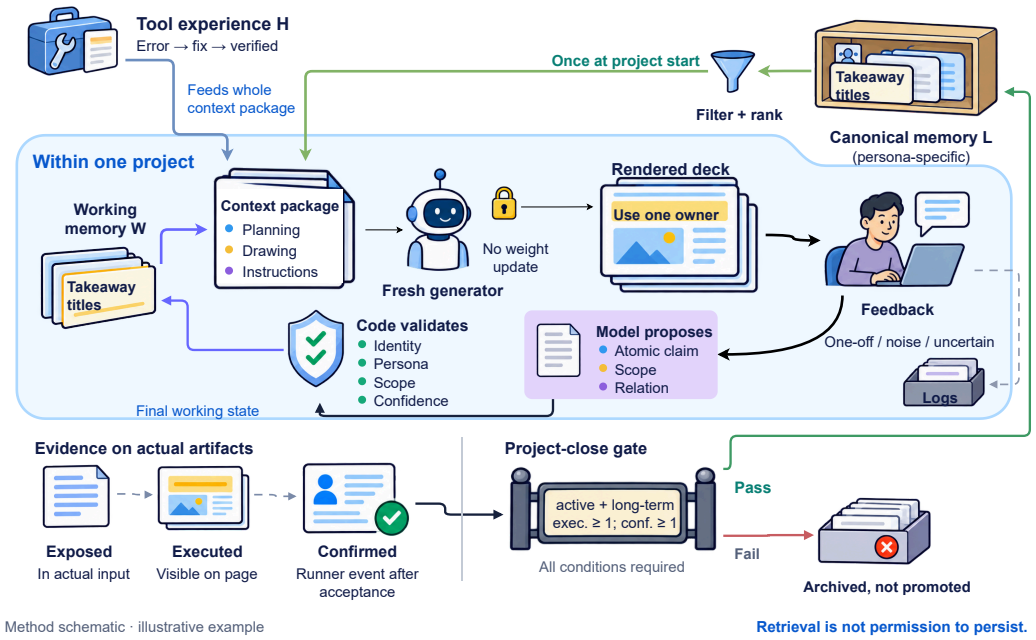

Figure 1: **SlideTrace overview.** Reusable feedback is represented as atomic, scoped preference claims with explicit relations and lifecycle state. Project working memory supports within-project adaptation, while validated updates are consolidated into canonical long-term memory for subsequent projects. Execution and confirmation provide evidence for long-term admission.

preferences through explicit relations. At the project boundary, supported updates are consolidated into a canonical long-term store, while task-relevant claims are retrieved to guide subsequent generation. Semantic analysis proposes claim content and relations, and deterministic validators govern scope, lifecycle transitions, and canonical updates. Execution and confirmation provide the final evidence for long-term admission.

Hierarchical memory has been explored in general-purpose agents such as MemGPT and in presentation-specific systems such as MemSlides (Packer et al., 2023; Jin et al., 2026), while Reflexion and ExpeL reuse interaction experience to improve subsequent behavior (Shinn et al., 2023; Zhao et al., 2023). SlideTrace focuses on the state of reusable preferences across projects: where a preference applies, how it relates to prior feedback, how its lifecycle changes, and what evidence supports its persistence.

We evaluate SlideTrace on sequential training projects followed by frozen-memory held-out tasks, using a shared slide generator across systems. On 15 Persona-B held-out tasks, SlideTrace reaches Score values of 8.36 under GPT-5.4 and 8.55 under Kimi-K3, exceeding the strongest baseline by 0.38 and 0.51 points, respectively. The largest separation appears in requirement realization, where the gains over MemSlides are 1.05 and 1.01 points. In the five-task ablation, Append-Only lowers Score by 0.23/0.47 points and Flat by 0.24/0.43 under GPT-5.4/Kimi-K3; Raw Feedback gives smaller reductions of 0.04/0.30. SlideTrace also reaches training acceptance in 1.65 rounds on average, compared with 1.85–2.20 for the baselines, while exposing 556 memory tokens per non-empty generation round rather than 711–1,544.

In summary, our main contributions are as follows:

- We formulate cross-project slide personalization as continual governance of preference state, with atomic, scoped claims—rather than raw feedback, facts, or interaction summaries—as the unit of persistence. Each claim retains explicit applicability, lifecycle, and provenance.

- We develop a two-timescale memory framework that combines project-local adaptation with relation-governed canonicalization and evidence-constrained consolidation across projects.
- We evaluate SlideTrace against four memory systems and a no-memory reference under a shared-generator, frozen-memory protocol. The results show consistent gains in held-out requirement realization, while ablations support relation-governed canonicalization and structured memory packages.

2 Related Work

Agent memory and experience reuse. Retrieval-augmented methods provide language models with external information (Lewis et al., 2020; Karpukhin et al., 2020; Wang et al., 2023b), while persistent-memory systems organize and update information across interactions. Generative Agents, MemGPT, and MemoryBank retain observations or user information over time; Mem0 extracts and consolidates facts, A-MEM maintains evolving linked notes, and MemOS manages memory with provenance and versioning (Park et al., 2023; Packer et al., 2023; Zhong et al., 2023; Chhikara et al., 2025; Xu et al., 2025; Li et al., 2025). Feedback-driven agents further reuse reflections, experience, and executable skills to improve subsequent behavior (Yao et al., 2023; Madaan et al., 2023; Shinn et al., 2023; Zhao et al., 2023; Wang et al., 2023a). The relevant distinction for our setting is not persistence alone, but the unit and update semantics of persistence. Unlike systems that primarily retain facts, linked notes, or interaction summaries, SlideTrace maintains reusable feedback as scoped preference claims whose relations, lifecycle, and evidence state are governed explicitly across project-local adaptation and cross-project canonicalization. Appendix A summarizes these representation and update differences across the compared systems.

Personalization and longitudinal evaluation. Personalization benchmarks study profile-conditioned generation (Salemi et al., 2024; Kumar et al., 2024), while longitudinal memory benchmarks examine recall, updates, temporal reasoning, and evolving user profiles (Maharana et al., 2024; Wu et al., 2024; Jiang et al., 2025). Our setting evaluates whether preferences acquired from prior projects are appropriately realized in new rendered artifacts.

Presentation generation and artifact evaluation. Presentation agents increasingly combine generation, revision, and artifact-level evaluation (Zheng et al., 2025; 2026; Zeng et al., 2025). Mem-Slides is the closest presentation-memory antecedent, using hierarchical memory to support personalization and local revision (Jin et al., 2026). Our comparison uses its memory architecture with the shared generator, thereby isolating memory behavior rather than comparing complete editing pipelines. SlideTrace focuses on cross-project preference-state governance, with held-out evaluation based on visible realization in rendered decks. This artifact-level protocol follows the broader motivation for visual and structured evaluation of generated outputs (Si et al., 2024; Liu et al., 2023; Zheng et al., 2023; Chen et al., 2024).

3 SlideTrace

SlideTrace formulates cross-project personalization as continual governance of preference state. An external slide generator remains parameter-frozen, while SlideTrace maintains project-local preference state and controls which preferences become eligible for persistence across projects. We first define governed preference state, then describe its project-local evolution, governed feedback updates, evidence-constrained consolidation, and interface with the generator.

3.1 System Boundary and Governed Preference State

For persona p, project j, and revision round r, a fresh generator session produces

$$D_{p,j}^{(r)} = G\Big(T_j, C_j, M_{p,j}^{(r)}\Big), \qquad M_{p,j}^{(r)} = \mathrm{Package}\Big(W_{p,j}^{(r)}, H\Big), \tag{1}$$

where T_j is the project task, C_j contains fixed generation constraints, $W_{p,j}^{(r)}$ is project working memory, and H is a separate tool-experience channel. The canonical store L_p persists across projects

for persona p. The generator creates slides from the task, constraints, and memory package, while SlideTrace constructs and updates the preference state supplied through that package. Sequential project interactions form the persistent memory state later used for cross-project personalization.

A governed preference claim is the unit of persistent preference state in SlideTrace. We represent a claim as

$$c = (x, d, \pi, s, \kappa, \rho, \gamma, \ell, E), \tag{2}$$

where x is normalized preference content, d is direction, π is priority, s is scope, κ contains conditions and exceptions, ρ is persistence eligibility, γ is semantic confidence, ℓ is lifecycle state, and E stores provenance and evidence events. Each claim additionally carries a stable identifier and persona metadata. Direction belongs to $\{\texttt{require}, \texttt{prefer}, \texttt{avoid}, \texttt{prohibit}\}$, priority belongs to $\{\texttt{must}, \texttt{should}, \texttt{soft}\}$, and persistence eligibility distinguishes `session_only` from `candidate_long_term`. Scope may span all projects or remain domain-specific, with audience restrictions added when supported by the feedback. Working copies retain links to their canonical counterparts, preserving provenance during project-local revision.

3.2 Two-Timescale Memory

The canonical store and project working memory define two adaptation timescales. Their evolution is summarized as

$$\begin{aligned} W_{p,j}^{(1)} &= \mathrm{Init}\Big(L_p^{(j)}, T_j\Big), \\ W_{p,j}^{(r+1)} &= \mathrm{Govern}\Big(W_{p,j}^{(r)}, F_{p,j}^{(r)}\Big), \\ L_p^{(j+1)} &= \mathrm{Consolidate}\Big(L_p^{(j)}, W_{p,j}^{(R_j)}\Big), \end{aligned} \tag{3}$$

where $L_p^{(j)}$ denotes the canonical store before project j, $F_{p,j}^{(r)}$ denotes the feedback received at revision round r, and R_j denotes the final revision round of project j. Working memory supports rapid within-project adaptation, while canonical-state updates occur through consolidation at the project boundary.

At project start, deterministic filtering checks persona identity, lifecycle state, scope compatibility, and activation conditions. Candidates that pass these checks are ranked by

$$q(c, T_j) = \lambda_s \,\mathrm{similarity}(c, T_j) + \lambda_c \,\mathrm{confidence}(c) + \lambda_p \,\mathrm{priority}(c). \tag{4}$$

Candidates with $q(c, T_j) \geq \tau_r$ are eligible for working memory, and at most K_r compatible preferences initialize $W_{p,j}^{(1)}$. The fixed values of $\boldsymbol{\lambda} = (\lambda_s, \lambda_c, \lambda_p)$, τ_r, and K_r, together with the numerical encoding of priority, are reported in Appendix A. These filtering and ranking operations determine the inherited preference state available at project initialization.

Stable feedback admitted by the governance layer updates project working memory directly. Within $W_{p,j}^{(r)}$, current-project preferences precede task-specific constraint records, which precede inherited preferences. Records within the same source tier are ordered by recency, priority, confirmation evidence, execution evidence, and stable identifier. The fixed generator constraints in C_j remain separate from these memory records. This design allows accepted project-local feedback to affect the next generation round while concentrating persistent state changes at project close.

3.3 Feedback Interpretation and State Governance

Long feedback is segmented into short, independent items. Given the current task scope and relevant working claims, a semantic analyzer classifies each item as a stable preference, one-off incident, noise, or uncertain content. Stable preferences proceed to claim construction and may update working memory. One-off incidents, noise, and uncertain items remain outside the preference state and are retained with their supporting context in the decision and evidence logs.

For each stable item, one semantic call jointly proposes normalized preference content, direction, priority, scope, conditions, and a relation to an existing candidate, together with its generator-package partition. The relation set covers duplication, reinforcement, refinement, subsumption, exception, contradiction, temporal update, independence, and uncertainty. Joint prediction keeps claim construction and relation interpretation within a shared semantic context.

The semantic analyzer proposes an interpretation, and the deterministic governance layer authorizes state transitions. It validates field completeness, legal values, candidate identifiers, persona isolation, scope boundaries, and lifecycle transitions. Scope is assigned at the widest reusable range supported by the feedback. General habits may span projects, domain-dependent requirements retain their domain, and audience restrictions require explicit support. Project-specific values and one-off names remain project-local.

Paraphrase merging requires relation confidence of at least τ_{dup}, while contradiction and exception require at least τ_{rel}. Relations below the corresponding threshold are recorded as uncertain and authorize no state transition. Malformed outputs receive at most R_{repair} repair attempts and are then recorded as validation failures. The fixed operating values are reported in Appendix A.

Validated updates are non-destructive and preserve lineage. Duplication and reinforcement attach new source evidence to an existing claim. Refinement and temporal update create lineage between preference versions. Conditional exceptions preserve their applicability ranges, while contradictions move the relevant claims to an unresolved state. Superseded, retracted, and unresolved claims leave the active set while remaining available for audit. Together, these relation-governed updates define canonicalization of reusable preference state across projects.

3.4 Evidence-Constrained Consolidation

Each preference claim records four event types: retrieval, exposure, execution, and confirmation. These events form an evidence chain with increasing proximity to the final artifact. Retrieval places a claim in the task-specific candidate set, exposure records its inclusion in the actual generator input, execution records visible realization identified by a verifier on rendered slides, and confirmation records project-level acceptance evidence written by the runner after persona acceptance. Their separation preserves the distinction between memory access, generator exposure, artifact realization, and accepted project outcome.

After each generation round, the runner performs a read-only execution check for every preference included in the generator input and returns `hit` or `miss` with slide-level evidence. A preference without visible supporting evidence is scored as `miss`. The check is read-only for every experimental system. SlideTrace records verified hits as execution events and runner-issued acceptance signals as confirmation events.

At project close, a project-local preference claim c enters the canonical store when

$$\mathrm{Promote}(c) \iff \mathrm{active}(c) \wedge \mathrm{candidate_long_term}(c) \wedge n_{\mathrm{exec}}(c) \geq 1 \wedge n_{\mathrm{conf}}(c) \geq 1. \tag{5}$$

Here, n_{exec} and n_{conf} count verified execution and confirmation events. Canonical admission requires an active lifecycle state, long-term eligibility, at least one verified execution event, and at least one project-level confirmation event. Session-only claims, claims lacking the required evidence, and superseded or retracted claims remain in the project archive. Confirmation is interpreted as project-level acceptance evidence.

3.5 Generator Interface

Active preference claims are serialized into Planning, Drawing, and Instructions sections before generation. Planning contains content structure, narrative order, and evidence requirements. Drawing contains layout, color, and visual hierarchy requirements. Instructions contains conditions, exceptions, and remaining operational constraints. These partitions organize generator exposure while preserving the underlying claim content, scope, evidence state, and lifecycle state. Together, they form the structured memory package used by the generator.

A deployment budget may omit complete preference records, with each omission recorded explicitly. Only claims included in the memory package receive exposure events, aligning exposure evidence with the actual generator input.

Tool experience occupies a separate auxiliary state channel H because tool-repair experience has different semantics from user preference. A valid tool-experience record contains an observed error, a recorded repair, and successful re-rendering or inspection. Repeated records are merged by tool, operation, error pattern, and repair, while later successes add supporting evidence. Retrieval first

filters by tool and operation and then ranks candidates by confidence and textual overlap. Retrieved tool guidance uses a separate input budget and retains its tool-experience identity. Keeping tool experience separate prevents rendering-specific repairs from being interpreted as user preferences.

4 EXPERIMENTS

4.1 EXPERIMENTAL SETUP

Benchmark and protocol. The benchmark contains two isolated persona tracks with identical task structures and distinct preference content. Each persona bank contains 216 private preferences, with a fixed applicability map assigning 12 preferences to each task according to domain, audience, and conditions. Each system completes 20 sequential training projects covering 168 distinct preferences, followed by 15 held-out tasks, B01–B15. We use Persona B for the main evaluation and report the complementary Persona-A analysis in Appendix B. A shared project order is deliberate in this stateful comparison: different orders would produce different memory states. Holding the curriculum fixed gives every system the same task sequence, while its native feedback and update policy determine the resulting interaction history and memory state.

Each system learns only from feedback induced by its own artifacts, with stores, histories, and outputs isolated across systems. Every model call uses a fresh session. During training, a persona evaluator inspects the rendered pages and returns at most eight visibly grounded issues when revision is required. A project is accepted when every must requirement is rated as a hit and the 3/2/1 priority-weighted requirement score reaches at least 8.5/10. The runner recomputes this decision from the 12 item-level verdicts rather than using the evaluator's self-reported stopping decision, and acceptance emits confirmation events for the same-round verified claims. The memory system processes unresolved feedback before the next generation round, with at most five rounds per project. After training, the memory state is frozen, and each held-out task uses an independent copy for read-only generation to prevent test-order contamination.

Baselines. We compare SlideTrace with Mem0, A-MEM, MemOS, a memory-decoupled adaptation of MemSlides, and a no-memory reference (Chhikara et al., 2025; Xu et al., 2025; Li et al., 2025; Jin et al., 2026). Mem0 uses fact extraction and updating, A-MEM uses linked notes and memory evolution, and MemOS uses a persistent-note adapter. The MemSlides adaptation retains its persona profile, working memory, preference stores, tool-experience channel, and consolidation mechanism under the shared generator. Each memory method otherwise follows its native storage, update, and retrieval policies.

Implementation. All methods use GLM-5.3-Flash through the shared ZCode agent harness, with common task inputs, page constraints, and rendering procedures. Methods requiring semantic embeddings use Qwen/Qwen3-Embedding-8B (Zhang et al., 2025). Method-specific memory operations follow their respective procedures, while the slide generator and rendering pipeline remain shared across systems. The generator is treated as a controlled executor rather than an experimental variable. Holding it fixed across all conditions removes generation-capability differences from the comparison, so observed differences can be attributed to the memory pipeline.

Evaluation. GPT-5.4 and Kimi-K3 independently evaluate anonymized held-out decks using only the task, fixed rubric, and rendered pages. For each of the 12 task requirements, the evaluator returns `hit`, `partial`, or `miss`, together with page-level visible evidence. A requirement without visible supporting evidence is scored as `miss`. We assign scores $s(v_i) \in \{10, 5, 0\}$ to `hit`, `partial`, and `miss`, respectively, and priority weights $w_i \in \{3, 2, 1\}$ to `must`, `should`, and `soft`. The reported metrics are

$$P = \frac{\sum_{i=1}^{12} w_i s(v_i)}{\sum_{i=1}^{12} w_i}, \qquad \text{Score} = 0.40P + 0.15S + 0.15K + 0.20C + 0.10V, \tag{6}$$

Here, P uses the same 12 requirement identifiers for every system and denotes requirement realization; S denotes scope accuracy, K domain knowledge, C cross-page consistency, and V visual quality. Because the comparison concerns memory, requirement realization receives the largest weight: it measures whether requirements learned from earlier projects appear in a new deck. Cross-page

Table 1: **Persona-B held-out results.** Mean scores over 15 tasks on a 0–10 scale. Best and second-best results within each evaluator are shown in **bold** and underline, respectively. P, S, K, C, and V denote requirement realization, scope accuracy, domain knowledge, cross-page consistency, and visual quality; Score is their weighted aggregate.

System	P	S	K	C	V	**Score**
GPT-5.4						
A-MEM	6.68	8.90	7.81	8.37	8.35	7.69
Mem0	6.43	9.07	7.79	8.32	**8.68**	7.63
MemOS	6.07	8.77	8.43	8.68	8.47	7.59
MemSlides	6.80	**9.09**	**8.79**	**8.71**	8.33	7.98
No memory	5.27	9.02	7.53	8.25	8.66	7.11
SlideTrace(ours)	**7.86**	9.08	8.66	8.55	8.47	**8.36**
Kimi-K3						
A-MEM	6.61	8.83	8.63	**8.93**	8.43	7.89
Mem0	6.69	8.80	8.73	8.57	8.50	7.87
MemOS	6.02	8.57	8.30	**8.93**	8.57	7.58
MemSlides	7.00	8.83	8.73	8.87	8.23	8.03
No memory	5.17	8.50	8.13	7.73	**8.77**	6.99
SlideTrace(ours)	**8.01**	**9.13**	**8.96**	8.85	8.59	**8.55**

consistency receives the next-largest weight because the pages must form one coherent artifact. Scope accuracy and domain knowledge account for task boundaries and subject-matter correctness, while visual quality keeps readability in the evaluation without allowing presentation polish to dominate the memory comparison. Appendix D reports sensitivity to this weighting. All dimensions use a 0–10 scale. Every evaluator output contains a verdict for all 12 requirement identifiers. The two evaluators provide complete coverage of every anonymized deck under the same page-grounded rubric. Reporting them separately preserves evaluator-specific variation rather than averaging it away. This design targets scalable, uniform, and page-grounded evaluation rather than replacing human judgment.

4.2 Main Results

Overall performance. Table 1 reports requirement realization (P), scope accuracy (S), domain knowledge (K), cross-page consistency (C), visual quality (V), and their weighted aggregate, Score, on the 15 Persona-B held-out tasks. SlideTrace achieves the highest Score under both evaluators, reaching 8.36 with GPT-5.4 and 8.55 with Kimi-K3. The strongest baseline, MemSlides, reaches 7.98 and 8.03, yielding margins of 0.38 and 0.51 points, respectively.

Requirement realization. The clearest separation appears in Requirement Realization P, the dimension most directly tied to memory. Relative to MemSlides, SlideTrace improves P by 1.05 points under GPT-5.4 and 1.01 points under Kimi-K3. The main gain therefore comes from carrying learned requirements into new tasks rather than from a uniform increase across all slide-quality dimensions. The closer S, K, C, and V results show that this gain does not come at the expense of scope control, domain correctness, cross-page consistency, or visual quality.

Task-level consistency. Paired task-level checks compare SlideTrace with MemSlides, the highest-scoring baseline. SlideTrace obtains a higher Score on 10 of 15 tasks under GPT-5.4 and on 13 of 15 tasks under Kimi-K3. The positive mean differences under both evaluators are therefore supported across multiple held-out tasks rather than arising from a single outlier. Complete task-level scores and pairwise descriptive checks are reported in Appendix B. Representative rendered pages from three cross-domain tasks on which SlideTrace ranks first under both evaluators are shown in Appendix B.1.

Trace-level transfer. We link the Persona-B applicability map, SlideTrace frozen store, held-out generator packages, and final item-level verdicts at the requirement level. Among 180 held-out

Table 2: **Persona-B requirement-level transfer trace.** Coverage rows link the 180 held-out requirement slots to the recorded training and memory states. Realization rows report hit counts separately for each evaluator and their strict intersection.

Trace subset	n	GPT-5.4 hit	Kimi-K3 hit	Both hit
Training-applicable	160	–	–	–
Matched to frozen-store claim	33	–	–	–
Retrieved and exposed	23	22 (95.7%)	22 (95.7%)	22 (95.7%)
Not exposed	157	91 (58.0%)	101 (64.3%)	89 (56.7%)

Table 3: **Ablation results on five Persona-B tasks.** Mean Score is reported for each evaluator and across all ten evaluator–task scores. Parenthesized values denote changes from Full.

Variant	GPT-5.4	Kimi-K3	Mean Score
Full	8.24	8.77	8.50
No Evidence Gate	8.32 (↑0.08)	8.63 (↓0.14)	8.47 (↓0.03)
Append-Only	8.01 (↓0.23)	8.30 (↓0.47)	8.15 (↓0.35)
Flat	8.00 (↓0.24)	8.34 (↓0.43)	8.17 (↓0.33)
Raw Feedback	8.20 (↓0.04)	8.47 (↓0.30)	8.33 (↓0.17)
No-Scope	8.26 (↑0.02)	8.65 (↓0.12)	8.46 (↓0.04)

requirement slots, 160 were applicable in at least one training project, 33 matched a claim in the frozen long-term store, and 23 matched a claim retrieved and exposed during held-out generation (Table 2). Both evaluators rated 22 of the 23 exposed requirements as hits (95.7%). For requirements not exposed from memory, the corresponding hit rates were 58.0% under GPT-5.4 and 64.3% under Kimi-K3. This separation connects the Score advantage to the retrieval–exposure–realization pathway: requirements selected from the frozen store are usually visible in the final deck. Because retrieval is selective, the association is descriptive rather than a randomized causal effect of exposure.

4.3 Ablation Study

We examine two stages of the SlideTrace memory pipeline on the same five Persona-B tasks. All three generation-time variants are constructed from the original Full store frozen after the 20-project Persona-B training sequence and modify only the package read by the generator, while the fixed-event replay modifies write-time governance. All six conditions use the same prespecified cross-domain tasks, generator, and evaluation criteria and are compared with the same Full condition. Flat preserves the retrieved claims, their content, metadata, and claim count from Full, removing only the Planning, Drawing, and Instructions partitioning before generation.

Append-Only and Flat produce the largest consistent reductions. Append-Only lowers Score by 0.23 under GPT-5.4 and 0.47 under Kimi-K3, while Flat lowers it by 0.24 and 0.43, respectively. Raw Feedback gives smaller reductions of 0.04 and 0.30. Append-Only removes consolidation before storage, whereas Flat removes structure before generation; their consistent losses locate the main contribution at both stages of the memory pipeline. The lower Raw Feedback results further support converting feedback into reusable claims. No Evidence Gate and No-Scope remain close to Full under both evaluators; Full, however, reaches comparable performance to No-Scope with a more compact injected context. The two stable effects in this probe are therefore claim canonicalization and package structure.

4.4 Training-Stage Diagnostics

All five memory systems reach persona acceptance on all 20 training projects. SlideTrace reaches first acceptance in 1.65 rounds per project, compared with 1.85–2.20 for the baselines, while injecting 556 rather than 711–1,544 memory tokens per non-empty generation round (Figure 2).

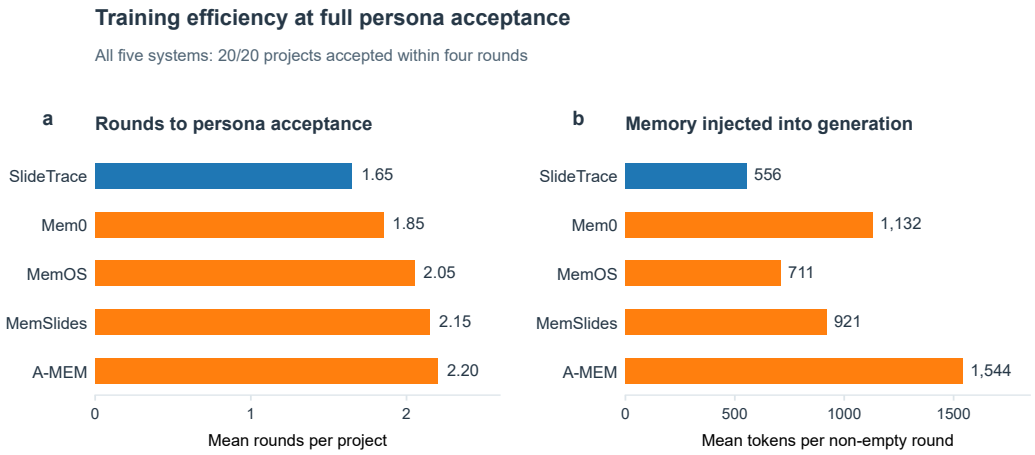

Figure 2: **Training-stage diagnostics on Persona B.** All memory systems reach persona acceptance on all 20 training projects. (a) Mean revision rounds to first acceptance per project. (b) Mean memory tokens injected per non-empty generation round.

5 Discussion and Limitations

Governance and evidence boundaries. The write-time replay provides more specific evidence for canonical-state governance. Removing relation-based canonicalization through Append-Only reduces Score under both evaluators, supporting governed updates over unrestricted accumulation under the matched event history. Full and No Evidence Gate remain close in the fixed-event replay. The measurable write-time gain is therefore localized to relation-governed canonicalization, while the evidence boundary functions as a conservative admission policy for long-term claims rather than as a primary source of score improvement. Together with the frozen-store diagnostic, these effects locate the observed gain in canonicalization and structured exposure rather than a larger long-term store.

Evaluation scope. Our evaluation uses two isolated synthetic personas under a fixed sequential curriculum. A shared project order holds the task sequence constant across systems, while each memory method carries its own state forward. The shared generator provides a common execution environment without replacing each system's native feedback loop. Persona B provides the main held-out comparison, and Persona A supplies a complementary transfer setting under the same protocol. GPT-5.4 and Kimi-K3 independently score the same anonymized rendered pages under one frozen rubric. Both evaluators identify SlideTrace as the leader by Score, providing two independent page-grounded assessments of the same artifacts. Under the shared generator, each method retains its native storage, update, retrieval, and computation, so the comparison measures end-to-end memory behavior rather than a compute-normalized proxy.

6 Conclusion

We presented SlideTrace, a cross-project personalization framework that maintains user feedback as governed preference state across sequential presentation tasks. SlideTrace organizes feedback into scoped preference claims, adapts them within project working memory, consolidates supported updates into a canonical long-term store, and retrieves task-relevant state for subsequent generation. Across 15 held-out tasks, SlideTrace reaches Score values of 8.36 under GPT-5.4 and 8.55 under Kimi-K3, exceeding MemSlides by 0.38 and 0.51 points, with the clearest gains in requirement realization. The generation-time and write-time ablations support relation-governed canonicalization and structured memory packages, while the training diagnostics show acceptance after 1.65 rounds on average with 556 injected memory tokens per non-empty generation round. These results support explicit preference-state governance as a practical foundation for persistent personalization across projects.

AI USE STATEMENT

Generative AI is integral to the experimental protocol: models generate slides, interpret preference feedback, provide persona-level responses, and judge rendered artifacts as described in Section 4. Generative AI assistance was also used to restructure and edit this manuscript, identify and check related literature, prepare LaTeX and figure source, and write scripts for score transcription and deterministic numerical analysis. The additional plots and tables are computed from supplied scores; they are not newly generated evaluation observations. Method schematics are explanatory diagrams.

ETHICS STATEMENT

The reported evaluation uses model-based personas rather than a longitudinal human-subject study. Persistent preference memory can nevertheless expose sensitive information, propagate misinterpretations, and retain outdated instructions. Persona isolation and scope filtering reduce particular failure modes but are not a privacy guarantee. Deployment should provide informed control over retention, inspection, correction, and deletion, and should not treat deck acceptance as consent to infer unrelated personal traits. Any release of tasks, logs, or artifacts should be checked for sensitive content and licensing constraints. No human-study approval or consent record is claimed for this synthetic evaluation.

REPRODUCIBILITY STATEMENT

The companion repository contains the versioned SlideTrace implementation and comparison adapters, frozen persona banks and benchmark definitions, final main and ablation score tables, figure-reconstruction scripts, and the numerical audit described in Appendix G. Together, these artifacts support deterministic reconstruction of the reported score analyses and diagnostic figures. Appendix A summarizes the implementation contract, and Appendix F reports the write-time claim-governance replay.

REFERENCES

Dongping Chen, Ruoxi Chen, Shilin Zhang, Yinuo Liu, Yaochen Wang, Huichi Zhou, Qihui Zhang, Yao Wan, Pan Zhou, and Lichao Sun. MLLM-as-a-judge: Assessing multimodal LLM-as-a-judge with vision-language benchmark. *arXiv preprint arXiv:2402.04788*, 2024.

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.

Bowen Jiang, Zhuoqun Hao, Young-Min Cho, Bryan Li, Yuan Yuan, Sihao Chen, Lyle Ungar, Camillo J. Taylor, and Dan Roth. Know me, respond to me: Benchmarking LLMs for dynamic user profiling and personalized responses at scale. *arXiv preprint arXiv:2504.14225*, 2025.

Ye Jin, Yangyang Xu, Jun Zhu, and Yibo Yang. MemSlides: A hierarchical memory driven agent framework for personalized slide generation with multi-turn local revision. *arXiv preprint arXiv:2606.17162*, 2026.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, 2020.

Ishita Kumar, Snigdha Viswanathan, Sushrita Yerra, Alireza Salemi, Ryan A. Rossi, Franck Dernoncourt, Hanieh Deilamsalehy, Xiang Chen, Ruiyi Zhang, Shubham Agarwal, Nedim Lipka, Chien Van Nguyen, Thien Huu Nguyen, and Hamed Zamani. LongLaMP: A benchmark for personalized long-form text generation. *arXiv preprint arXiv:2407.11016*, 2024.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems*, 2020.

Zhiyu Li et al. MemOS: A memory os for AI system. *arXiv preprint arXiv:2507.03724*, 2025.

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

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. G-Eval: NLG evaluation using GPT-4 with better human alignment. *arXiv preprint arXiv:2303.16634*, 2023.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Sean Welleck, Bodhisattwa Prasad Majumder, Shashank Gupta, Amir Yazdanbakhsh, and Peter Clark. Self-refine: Iterative refinement with self-feedback. In *Advances in Neural Information Processing Systems*, 2023.

Adyasha Maharana, Dong-Ho Lee, Sergey Tulyakov, Mohit Bansal, Francesco Barbieri, and Yuwei Fang. Evaluating very long-term conversational memory of LLM agents. *arXiv preprint arXiv:2402.17753*, 2024.

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.

Joon Sung Park, Joseph C. O'Brien, Carrie J. Cai, Meredith Ringel Morris, Percy Liang, and Michael S. Bernstein. Generative agents: Interactive simulacra of human behavior. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*, 2023.

Alireza Salemi, Sheshera Mysore, Michael Bendersky, and Hamed Zamani. LaMP: When large language models meet personalization. *arXiv preprint arXiv:2304.11406*, 2024.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: Language agents with verbal reinforcement learning. In *Advances in Neural Information Processing Systems*, 2023.

Chenglei Si, Yanzhe Zhang, Zhengyuan Yang, Ruibo Liu, and Diyi Yang. Design2Code: How far are we from automating front-end engineering? *arXiv preprint arXiv:2403.03163*, 2024.

Guanzhi Wang, Yuqi Xie, Yunfan Jiang, Ajay Mandlekar, Chaowei Xiao, Yuke Zhu, Linxi Fan, and Anima Anandkumar. Voyager: An open-ended embodied agent with large language models. *arXiv preprint arXiv:2305.16291*, 2023a.

Weizhi Wang, Li Dong, Hao Cheng, Xiaodong Liu, Xifeng Yan, Jianfeng Gao, and Furu Wei. Augmenting language models with long-term memory. *arXiv preprint arXiv:2306.07174*, 2023b.

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

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. ReAct: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations*, 2023.

Wenzheng Zeng, Mingyu Ouyang, Langyuan Cui, and Hwee Tou Ng. SlideTailor: Personalized presentation slide generation for scientific papers. *arXiv preprint arXiv:2512.20292*, 2025.

Yanzhao Zhang, Mingxin Li, Dingkun Long, Xin Zhang, Huan Lin, Baosong Yang, Pengjun Xie, An Yang, Dayiheng Liu, Junyang Lin, Fei Huang, and Jingren Zhou. Qwen3 Embedding: Advancing text embedding and reranking through foundation models. *arXiv preprint arXiv:2506.05176*, 2025.

Andrew Zhao, Daniel Huang, Quentin Xu, Matthieu Lin, Yong-Jin Liu, and Gao Huang. ExpeL: LLM agents are experiential learners. *arXiv preprint arXiv:2308.10144*, 2023.

Hao Zheng, Xinyan Guan, Hao Kong, Jia Zheng, Weixiang Zhou, Hongyu Lin, Yaojie Lu, Ben He, Xianpei Han, and Le Sun. PPTAgent: Generating and evaluating presentations beyond text-to-slides. *arXiv preprint arXiv:2501.03936*, 2025.

Hao Zheng, Guozhao Mo, Xinru Yan, Qianhao Yuan, Wenkai Zhang, Xuanang Chen, Yaojie Lu, Hongyu Lin, Xianpei Han, and Le Sun. DeepPresenter: Environment-grounded reflection for agentic presentation generation. *arXiv preprint arXiv:2602.22839*, 2026.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging LLM-as-a-judge with MT-Bench and chatbot arena. *arXiv preprint arXiv:2306.05685*, 2023.

Wanjun Zhong, Lianghong Guo, Qiqi Gao, He Ye, and Yanlin Wang. MemoryBank: Enhancing large language models with long-term memory. *arXiv preprint arXiv:2305.10250*, 2023.

A PREFERENCE-CLAIM LIFECYCLE AND IMPLEMENTATION CONTRACT

Algorithm 1 summarizes the ordering implemented by the versioned preference-claim lifecycle. The semantic analyzer proposes claim content and relations but does not directly mutate either store, and the generator session is not a persistence mechanism.

Algorithm 1 SlideTrace preference-claim lifecycle

Require: Persona p, project task T_j, constraints C_j, canonical store L_p, tool store H, maximum rounds $R = 5$
Ensure: Final deck, archived project claims, and eligible canonical claim updates

$Q \leftarrow \mathrm{HardFilter}(L_p, p, T_j)$
$Q \leftarrow \mathrm{Top}_{K_r}(\{c \in Q : q(c, T_j) \geq \tau_r\})$ ▷ Equation 4
Initialize project claim state W from Q with lineage and task constraints; record retrieval events
for $r = 1, \ldots, R$ **do**
    $M \leftarrow \mathrm{Package}(\mathrm{RankActive}(W), T_j, H)$
    Record exposure only for claims actually included in M
    $D \leftarrow G(T_j, C_j, M)$ in a fresh independent generator session
    $X \leftarrow \mathrm{VerifyVisibleExecution}(D, \mathrm{Claims}(M))$
    Record verified hits as execution evidence
    Record eligible tool error–fix–verification events separately
    $(a, F) \leftarrow \mathrm{PersonaEvaluate}(D, T_j, \mathcal{A}_{p,j})$
    if $a = 1$ **then**
        Runner writes confirmation events under its acceptance rule
        break
    end if
    for each atomic feedback item $f \in F$, with $|F| \leq 8$ **do**
        $z \leftarrow \mathrm{SemanticPropose}(f, T_j, W)$ in a fresh session
        Validate fields, identity, persona, scope, relation, and confidence
        Repair malformed output at most R_{repair} times; otherwise reject closed
        if z is a valid stable preference **then**
            $W \leftarrow \mathrm{ApplyRelationGovernedUpdate}(W, z)$; preserve lineage
        else
            Log the decision without adding a preference to W
        end if
    end for
end for
Archive the project and final satisfaction state
Promote active reusable claims satisfying the evidence boundary in Equation 5

Preference-claim state and evidence. An atomic claim records its subject, attribute, value or action, direction, priority, scope, conditions, exceptions, package partition, and source evidence. Working copies retain their canonical links. The four evidence signals have different meanings: retrieval records candidate selection, exposure records actual prompt inclusion, execution records a visually supported hit, and confirmation records the runner's acceptance-related event. An exposure counter is not a success counter. The implementation serializes event identifiers and the provenance of lifecycle transitions in addition to aggregate counts.

Versioned configuration. Table 4 summarizes the principal experimental settings used in the reported runs.

Table 4: Implementation contract and versioned experimental configuration.

Item	Documented setting and repository record
Generator and harness	GLM-5.3-Flash through ZCode under the shared generator interface and task protocol.
Embedding model	Qwen/Qwen3-Embedding-8B with normalized vectors and fixed indexing settings.
Retrieval	One canonical query per project; hard filters precede ranking; $\boldsymbol{\lambda} = (0.75, 0.15, 0.10)$; $\tau_r = 0.35$; $K_r = 15$.
Priority and confidence	Ordered priority levels, numerical encoding, and confidence normalization are fixed in the implementation.
Semantic relation thresholds	$\tau_{\mathrm{dup}} = 0.90$; $\tau_{\mathrm{rel}} = 0.85$ for contradiction and exception; the relation configuration is versioned with the implementation.
Feedback and iteration	At most eight feedback items and five rounds per project; $R_{\mathrm{repair}} = 3$.
Memory package	Planning, Drawing, Instructions; no fixed token cap; an optional token budget omits whole claims. The injected-token series is retained in the training diagnostics.
Evaluation	GPT-5.4 and Kimi-K3 independently apply the frozen five-dimensional rubric to anonymized rendered pages.
Rendering and tasks	Three rendered pages per task; task-specific content, audience, and goal definitions are fixed in the benchmark files.
Baselines	Native memory mechanisms operate through the comparison adapters under the shared generator and task protocol.

Persistent units and update semantics. Table 5 summarizes the persistent units and update semantics instantiated by the comparison adapters. It characterizes each system as used in this study rather than its full native capability set.

Table 5: Persistent units and update semantics in the compared memory systems.

System	Persistent unit	Update semantics represented in the comparison
Mem0	Extracted user facts	Fact extraction, update, and consolidation.
A-MEM	Evolving linked notes	Note linking and memory evolution.
MemOS	Persistent notes	Provenance-aware persistence and versioning.
MemSlides	Hierarchical interaction memory	Persona profile, working memory, preference and tool-experience stores, and consolidation.
SLIDETRACE	Scoped preference claims	Explicit relation, lifecycle, and evidence state across project-local adaptation and cross-project canonicalization.

Parameter choices. The ranking gives most of its weight to semantic match (0.75), because a memory should first be relevant to the current task. Confidence (0.15) lowers the rank of weakly supported claims, while priority (0.10) orders relevant claims without allowing a generic high-priority claim to override task relevance. After persona and scope filtering, the 0.35 cutoff removes weak candidates, and the 15-claim retrieval limit prevents inherited memory from crowding the task prompt. We use 0.90 for duplicate merging because an incorrect merge would alter the canonical record; contradiction and exception relations use 0.85 because both claims remain traceable in the versioned store. Malformed outputs receive at most three repair attempts before the update is rejected. These settings remain fixed across all SlideTrace runs.

Artifact-local corrections. The classifier excludes `one_off_incident` from reusable preference memory while retaining the event in the project trace. This keeps transient artifact corrections auditable without allowing them to alter cross-project preference state.

Evidence binding. Claim-level event identifiers link each execution verdict to the rendered artifact, accepted project round, and promoted claim version. These links make the promotion path explicit in the implementation rather than inferring it from aggregate counters.

B Complete Held-Out Results and Task-Level Analysis

Table 6 contains every Persona-B main-benchmark Score used in the analysis. All six systems have scores for B01–B15. A baseline name denotes the adapter described in Section 4; in particular, MemSlides is the memory-decoupled condition, not its original complete presentation agent.

Table 6: Complete Persona-B held-out Scores under the 3/2/1 priority-weighted requirement metric. Each judge is shown separately; every system has all 15 task Scores.

Task	SlideTrace	A-MEM	Mem0	MemOS	MemSlides	No memory
GPT-5.4						
B01	8.28	8.05	8.08	8.71	8.47	8.28
B02	8.34	7.52	7.35	7.36	7.88	6.99
B03	8.66	8.36	8.88	8.36	7.65	7.42
B04	8.67	8.92	7.42	7.57	7.60	7.52
B05	9.08	8.56	9.10	8.26	8.97	8.33
B06	8.43	8.34	6.65	8.63	8.66	6.53
B07	8.25	6.90	7.39	7.63	8.03	5.78
B08	7.59	6.81	7.02	7.44	7.94	7.98
B09	8.32	8.28	8.12	7.64	8.03	7.05
B10	7.94	7.35	6.84	7.29	6.70	6.91
B11	7.81	4.66	7.46	5.98	6.71	6.36
B12	8.63	8.78	8.32	7.70	8.68	7.11
B13	8.45	8.03	7.54	7.44	8.97	7.33
B14	9.12	7.80	7.07	6.99	7.98	6.48
B15	7.85	6.92	7.26	6.86	7.41	6.55
Kimi-K3						
B01	8.68	8.59	8.82	8.78	9.17	6.62
B02	8.64	7.75	7.61	7.07	7.77	7.54
B03	8.06	8.46	8.24	8.96	7.34	6.80
B04	8.20	8.93	7.09	7.55	8.16	7.54
B05	8.93	8.32	9.18	8.38	9.09	7.11
B06	8.90	7.81	7.74	7.92	8.72	7.06
B07	8.30	7.01	7.63	7.56	7.81	6.20
B08	8.51	7.52	7.61	7.43	8.30	8.04
B09	8.54	8.74	8.04	7.52	7.36	7.30
B10	7.42	7.57	7.49	7.07	7.29	6.81
B11	8.54	6.78	7.12	5.86	6.09	6.70
B12	9.01	8.52	8.60	7.47	8.86	6.42
B13	9.13	7.57	7.50	8.01	8.91	6.69
B14	9.14	7.50	8.17	7.56	8.00	6.63
B15	8.20	7.32	7.22	6.61	7.62	7.33

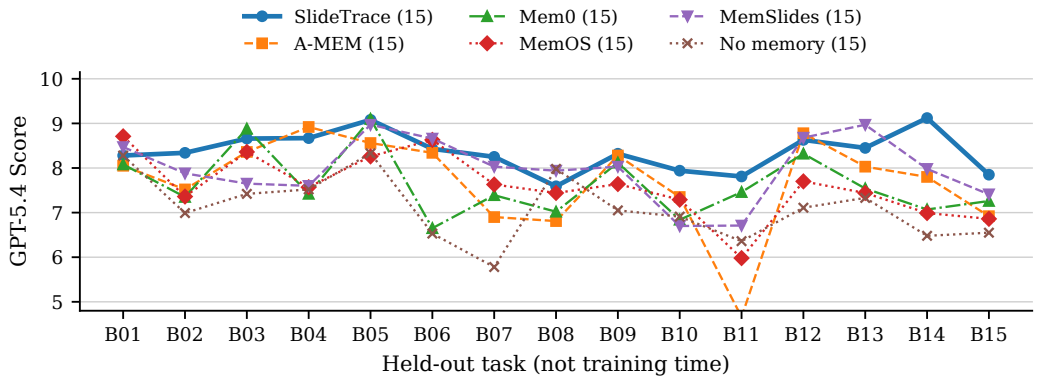

Figure 3: Full GPT-5.4 Persona-B held-out Score profile. All decks use frozen post-training stores, and all six systems cover B01–B15. Connecting lines organize task identifiers and do not indicate temporal progression.

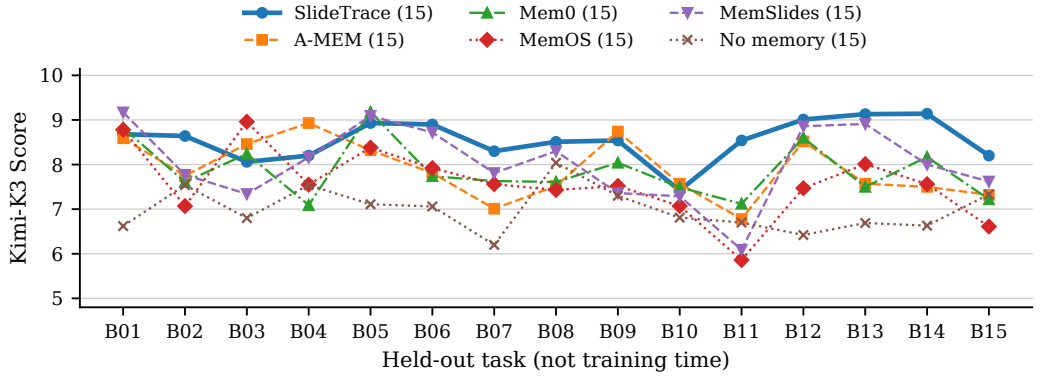

Figure 4: Full Kimi-K3 Persona-B held-out Score profile over B01–B15. The numerical scale is shared with Figure 3; evaluator outputs remain separate.

Figures 3 and 4 expose task-level variation that a mean alone hides. Across the printed task Scores, the sample standard deviations for SLIDETRACE/MemSlides are 0.44/0.71 under GPT-5.4 and 0.46/0.84 under Kimi-K3. The lower spread for SLIDETRACE accompanies its higher mean under both judges.

Pairwise descriptive checks. For judge e and baseline b, let $\mathcal{J} = \{\mathrm{B01}, \ldots, \mathrm{B15}\}$ be the shared task set. Define

$$d_j = \mathrm{Score}_{\mathrm{ST},e,j} - \mathrm{Score}_{b,e,j}, \qquad \bar{d} = \frac{1}{|\mathcal{J}|} \sum_{j \in \mathcal{J}} d_j. \tag{7}$$

The win/tie/loss columns in Table 7 count the signs of these differences. The leave-one-task-out range is the minimum and maximum of the means obtained after removing one task at a time. These checks summarize sensitivity to task composition within the fixed sequential benchmark.

Transfer-trace construction. The trace in Table 2 uses the 180 requirement slots in the final Persona-B evaluation. A slot is *training-applicable* when its normalized claim identifier occurs in at least one of the 20 training-task applicability maps; this denotes an opportunity for the preference to apply, not necessarily an emitted feedback item. Because semantic extraction may rewrite preference text, held-out requirements are linked to frozen-store claims by a bigram-Dice match with threshold 0.30. A matched claim is marked as retrieved and exposed only when it also matches a serialized line in the actual held-out generator package. This procedure identifies 160 training-applicable slots

Table 7: Pairwise main-benchmark differences across all 15 tasks. Positive mean differences favor SlideTrace. Leave-one-out ranges are descriptive, not confidence intervals.

Judge	Baseline	n	Mean Score diff.	Win/tie/loss	Leave-one-out range
GPT-5.4	A-MEM	15	+0.68	13/0/2	[0.50, 0.74]
GPT-5.4	Mem0	15	+0.73	13/0/2	[0.63, 0.80]
GPT-5.4	MemOS	15	+0.77	13/0/2	[0.67, 0.86]
GPT-5.4	MemSlides	15	+0.38	10/0/5	[0.32, 0.45]
GPT-5.4	No memory	15	+1.25	13/1/1	[1.15, 1.37]
Kimi-K3	A-MEM	15	+0.65	11/0/4	[0.58, 0.75]
Kimi-K3	Mem0	15	+0.68	11/0/4	[0.61, 0.74]
Kimi-K3	MemOS	15	+0.96	13/0/2	[0.84, 1.10]
Kimi-K3	MemSlides	15	+0.51	13/0/2	[0.38, 0.59]
Kimi-K3	No memory	15	+1.56	15/0/0	[1.49, 1.64]

(100 unique claim identifiers), 33 slots matched to the frozen store (22 unique identifiers), and 23 exposed slots (16 unique identifiers). GPT-5.4 and Kimi-K3 verdicts are parsed separately from the final weighted-evaluation records; "both hit" is their strict intersection. The score archive retains the resulting linkage counts and evaluator hit rates.

B.1 Qualitative Showcases

Figures 5–10 pair one representative SlideTrace page with the same page index from a contrast system on B02, B11, and B14, the three held-out tasks on which SlideTrace ranks first under both evaluators. For each task, the contrast system is the one with the lowest mean Score across the two evaluators: MemOS for B02, A-MEM for B11, and No memory for B14. Under the priority-weighted main scoring rule, SlideTrace's margins over the strongest baseline are $+0.46/+0.87$, $+0.35/+1.42$, and $+1.14/+0.97$ under GPT-5.4/Kimi-K3, respectively. The tasks span campaign planning, financial review, and healthcare quality. Across these domains, the common advantage is requirement realization: requested preferences appear as inspectable decision structures rather than generic prose. B02 links phases to owners, dependencies, acceptance gates, and explicit approval requests; B11 connects scenario assumptions to cash-flow and coverage outcomes and then to a trigger-based risk register; B14 states the cohort denominator, missingness, confidence intervals, and risk-adjusted comparisons. The paired contrast pages remain visually structured, but omit several of these task-specific requirements, matching the item-level verdicts from both evaluators.

Table 8 makes this comparison explicit at the Persona-B requirement level. Every listed requirement is a hit for SlideTrace under both evaluators. The contrast columns show where an otherwise coherent deck leaves the same preference incomplete or absent.

Table 8: **Representative Persona-B requirements realized in the showcase decks.** Verdict pairs are GPT-5.4/Kimi-K3; H, P, and M denote hit, partial, and miss, respectively.

Task	ID	Persona requirement	SlideTrace	Contrast
B02	R3	Separate items requiring approval, awareness, or further validation.	H/H	H/P
B02	R5	For each channel, specify the target role, conversion task, and content format.	H/H	P/P
B02	R12	End with only the budget, creative direction, and stop-loss decisions requiring management action.	H/H	P/P
B11	R2	Explain operations on the management basis and reconcile disclosure-basis differences item by item.	H/H	M/P
B11	R4	Report receivable, inventory, and payable days, aging structure, and accountable owners.	H/H	M/M
B11	R6	Connect management and statutory bases with an auditable reconciliation table.	H/H	M/M
B11	R10	Bridge major variances into volume, price, mix, foreign exchange, and one-off effects.	H/H	M/P
B11	R12	State assumptions and correlations for base, stress, and reverse-stress scenarios.	H/H	M/P
B14	R2	Give effect size, confidence interval, and missingness explanation for group comparisons.	H/H	M/P
B14	R5	Report confidence intervals for small-sample indicators.	H/H	M/M
B14	R7	State the escalation deadline, reporting recipient, and protective measures for severe events.	H/H	M/M
B14	R8	Provide a department-level primary view that supports drill-down.	H/H	M/M

Additional Persona-A comparison. The earlier Persona-A benchmark provides a second fixed-persona comparison under the same six-system protocol. Table 9 retains its reported aggregates; SlideTrace also ranks first under both judges.



Figure 5: **B02 campaign-planning showcase.** The closing page makes the delivery sequence operational through owners, dependencies, leading and lagging targets, acceptance gates, stop conditions, and separate approval, awareness, and continued-validation requests.



Figure 6: **B02 contrast (MemOS, same page index).** The page provides channel economics and stopping conditions, but retains one budget case and month-level phases. Both evaluators identify the missing reduced/base/expanded budget comparison, week-level owner–dependency–deliverable plan, and explicit separation of approval, awareness, and continued-validation requests.



Figure 7: **B11 financial-review showcase.** Base, stress, and reverse-stress assumptions are tied to operating cash flow and coverage ratios, then carried into a risk register with probability, impact, trigger, mitigation, and residual risk.



Figure 8: **B11 contrast (A-MEM, same page index).** The page emphasizes trust boundaries, data-flow contracts, and solution alternatives. Across the full deck, both evaluators identify missing or incomplete three-way financial stress scenarios, working-capital aging and ownership, management-to-statutory reconciliation, and the requested variance bridge.



Figure 9: **B14 healthcare-quality showcase.** The page exposes the cohort definition, numerator and denominator, missingness mechanism, confidence intervals, and department-level risk-adjusted comparison, making the evidence chain directly auditable.



Figure 10: **B14 contrast (No memory, same page index).** The page gives a clear cohort overview and point estimates, but the full deck omits confidence intervals, an explicit missingness mechanism, the severe-event escalation path, and a department-level drill-down. Both evaluators mark these requirements as missing or only partially realized.

Table 9: Additional Persona-A held-out results. Reported means on a 0–10 scale; larger is better.

System	n	P	S	K	C	V	Score
GPT-5.4							
SlideTrace	15	**7.91**	**8.87**	**7.87**	**8.80**	**7.73**	**8.21**
A-MEM	15	6.92	8.40	7.53	8.27	7.60	7.57
Mem0	15	6.68	8.27	7.27	8.20	7.47	7.39
MemOS	15	6.33	8.20	7.27	7.87	7.27	7.15
MemSlides	15	6.62	8.14	7.34	7.87	7.60	7.30
No memory	15	5.94	8.40	7.34	7.87	7.13	7.02
Kimi-K3							
SlideTrace	15	**8.08**	**9.00**	**7.00**	**8.13**	**6.93**	**7.95**
A-MEM	15	7.36	8.80	6.73	7.73	6.67	7.49
Mem0	15	6.45	8.87	6.33	7.80	6.13	7.03
MemOS	15	6.41	8.87	6.20	7.13	5.80	6.83
MemSlides	15	6.41	8.80	6.46	7.66	6.40	7.03
No memory	15	6.79	8.87	6.33	7.73	6.53	7.20

C Complete Ablation Results

The ablation uses the same five Persona-B tasks for the Full condition, two write-time variants, and three generation-time variants. The generation-time variants are derived from the original Full store frozen after the 20-project Persona-B training sequence. Flat is claim-matched to Full and changes only generation-time partitioning. Table 10 reports every task-level Score. Each cell is ordered as GPT-5.4/Kimi-K3. Append-Only and Flat show the largest consistent reductions from Full. Raw Feedback is also lower under both evaluators, while No Evidence Gate and No-Scope remain closer to Full. The corresponding ten-score means are 8.50 for Full, 8.47 for No Evidence Gate, 8.15 for Append-Only, 8.17 for Flat, 8.33 for Raw Feedback, and 8.46 for No-Scope.

Table 10: Complete five-task Persona-B ablation Scores. Each cell reports GPT-5.4/Kimi-K3. The mean row is computed over the five tasks for each evaluator.

Task	Full	No Evidence Gate	Append-Only	Flat	Raw Feedback	No-Scope
B01	8.25/8.39	8.23/8.12	7.99/7.36	8.07/8.22	8.40/8.14	7.83/8.10
B02	8.44/9.07	8.60/8.90	7.56/8.65	8.15/8.87	8.22/8.70	8.67/9.12
B10	8.51/8.95	8.46/8.97	8.58/8.67	8.64/8.87	8.27/8.56	8.20/8.85
B14	8.23/8.55	7.96/7.97	8.16/8.57	7.73/7.96	8.18/9.03	8.26/8.54
B15	7.75/8.87	8.35/9.17	7.74/8.26	7.42/7.80	7.93/7.91	8.35/8.64
Mean	8.24/8.77	8.32/8.63	8.01/8.30	8.00/8.34	8.20/8.47	8.26/8.65

D SENSITIVITY TO SCORING WEIGHTS

The main Score assigns 70% to the $P/S/K$ group and 30% to the C/V group. To examine dependence on this weighting, we hold each group's internal weights fixed and define

$$M = \frac{0.40P + 0.15S + 0.15K}{0.70}, \qquad Q = \frac{0.20C + 0.10V}{0.30}, \tag{8}$$
$$\text{Score}_\alpha = \alpha M + (1-\alpha)Q, \qquad 0 \le \alpha \le 1.$$

At $\alpha = 0.70$, this recovers the specified Score up to rounding of the component means. Figures 11 and 12 include all six systems over the complete 15-task Persona-B benchmark. They are arithmetic projections of reported component means; changing α does not invoke the generator or rerun an evaluator. At the specified $\alpha = 0.70$, SLIDETRACE ranks first under both judges. Its separation

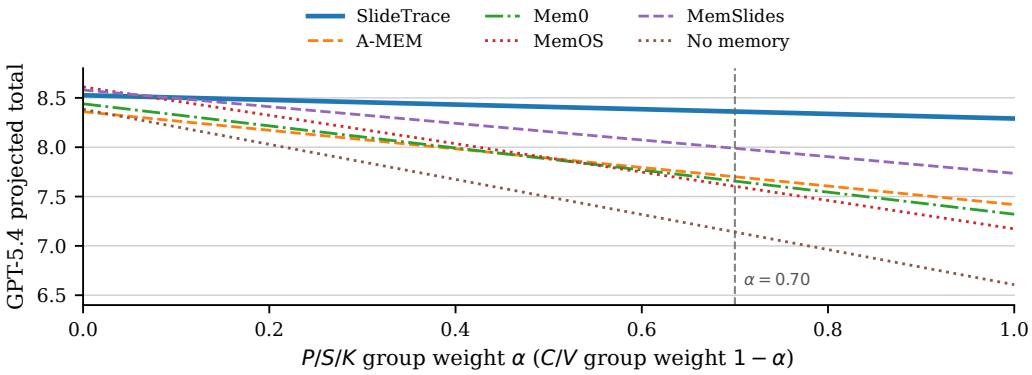

Figure 11: GPT-5.4 arithmetic weight sensitivity for all six systems. The vertical reference marks the specified $P/S/K$ group weight of 0.70.

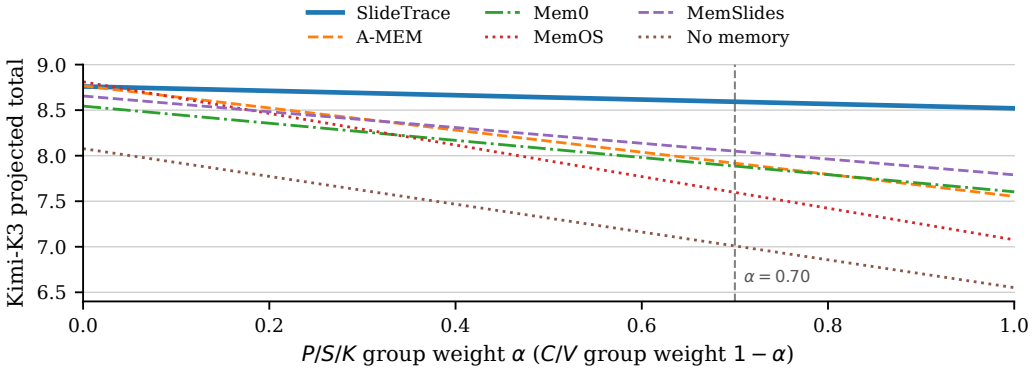

Figure 12: Kimi-K3 arithmetic weight sensitivity under the same $P/S/K$–C/V transformation. Curves are based on aggregate dimension means, not newly judged decks.

grows as more weight is assigned to the $P/S/K$ group, while the systems are closer near the C/V endpoint. The projection therefore shows that the main ordering is robust across a broad range of relative weights and is strongest toward the requirement-realization, scope-accuracy, and domain-knowledge side of the rubric.

E FROZEN-STORE DIAGNOSTICS

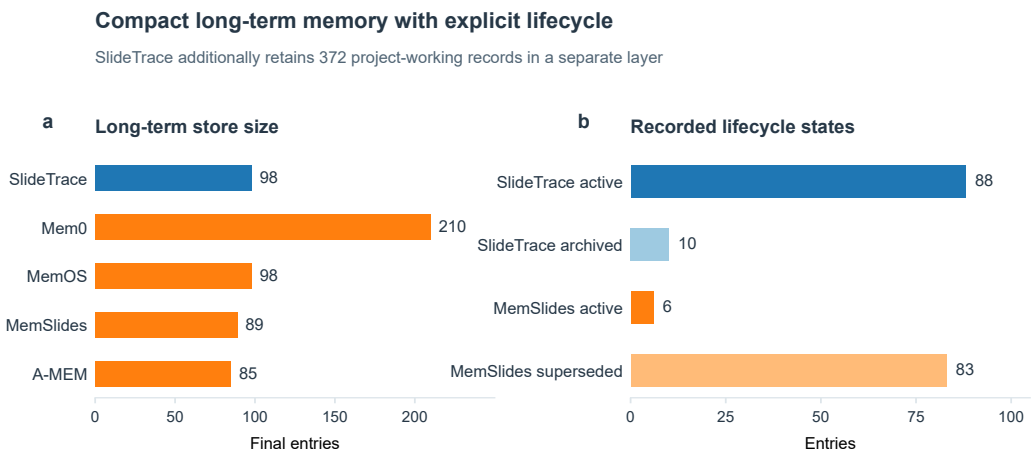

Figure 13: **Persona-B long-term memory size and lifecycle state after training.** (a) Final long-term entry counts. SlideTrace's 372 project-working records are held in a separate layer and excluded from its long-term count. (b) Native lifecycle states for SLIDETRACE, separated into active and archived entries, and MemSlides, separated into active and superseded entries; the other stores have no comparable status field.

SLIDETRACE ended training with 98 long-term claims, including 88 active and 10 archived, while 372 project-working records remained separate (Figure 13). Its long-term count matched MemOS and was less than half of Mem0's 210 entries. At the same 98-entry store size as MemOS, SLIDETRACE injected less context (556 versus 711 tokens) and reached acceptance sooner (1.65 versus 2.05 rounds). Store size alone therefore does not explain the observed difference: SLIDETRACE maintains explicit claim states while exposing a smaller, effective memory package to the generator. Its 4.1% near-duplicate share mainly reflects retained archived–active versions.

F WRITE-TIME CLAIM-GOVERNANCE REPLAY

We replay the Persona-B 20-project event chain while holding project order, feedback, atomic claim proposals, scopes, evidence events, generator, retrieval, and held-out tasks fixed. Only the rule for incorporating eligible claims into the canonical store changes. The three resulting stores are evaluated on the same five held-out tasks (B01, B02, B10, B14, and B15) by GPT-5.4 and Kimi-K3.

Let $B(c)$ indicate that a claim is active and reusable, and let $G(c)$ indicate that the claim satisfies the execution-and-acceptance evidence boundary in Equation 5. Let $\mathcal{R}$ denote the validated relation-governance operation that may merge, refine, qualify, replace, or retain a claim while preserving lineage. The replay conditions are:

Replay condition	Canonical-store update
Full Claim Governance	Apply $\mathcal{R}$ to claims satisfying $B(c) \wedge G(c)$.
Append-Only Canonicalization	For every claim satisfying $B(c) \wedge G(c)$, append a distinct canonical record; disable duplicate and reinforcement merges, refinement and subsumption updates, exceptions, temporal replacement, and contradiction-driven state transitions.
No Evidence Gate	Apply $\mathcal{R}$ to every claim satisfying $B(c)$, without requiring $G(c)$.

Full Claim Governance is the documented SlideTrace update path. Append-Only Canonicalization retains claim extraction, scope, evidence eligibility, and retrieval, but removes relation-driven curation so that repeated or revised claims remain as separate records. No Evidence Gate retains relation governance but removes the final execution-and-acceptance requirement; active reusable status remains a basic claim-validity condition rather than part of the ablated evidence boundary.

The task-level replay scores are reported together with the generation-time variants in Table 10. Append-Only Canonicalization is lower than Full by 0.23 under GPT-5.4 and 0.47 under Kimi-K3. No Evidence Gate changes the mean by $+0.08$ under GPT-5.4 and -0.14 under Kimi-K3. Because the upstream event chain remains fixed, the replay estimates write-time effects conditional on the observed history rather than changes to future feedback trajectories.

G SCORE-LEVEL REPRODUCIBILITY AND ARTIFACT AVAILABILITY

Score records. The score archive distinguishes reported aggregate dimensions, printed per-task totals, and additional calculations from those totals. Main-benchmark and ablation observations are separate datasets, even when their task identifiers coincide. The author documentation records the table-level provenance of the transcribed data. No observations from other evaluator configurations or scoring rules enter these analyses.

Machine-readable reconstruction. The companion archive separates main-benchmark aggregates, per-task totals, ablation aggregates, and ablation task scores. There are 180 main judge scores with complete coverage of six systems, 15 tasks, and two judges, plus 60 ablation scores. Rounded published aggregates are retained as published; derived statistics are independently computed from the printed inputs.

The score audit checks coverage, score ranges, reported means, and weighted totals. `scripts/plot_heldout_profiles.py` and `scripts/plot_weight_sensitivity.py` regenerate the four held-out profile and sensitivity figures. The training and frozen-store figures are generated from the reported aggregate diagnostics by `scripts/plot_diagnostics.py`. The overview schematic and all quantitative figure files reside in `figure/`; the included vector PDFs permit ordinary pdfLaTeX compilation without Python, shell escape, or external network requests.

Rounding convention. A rounded mean of rounded task scores need not exactly equal a mean computed before rounding, and weighting rounded component means can introduce another small discrepancy. The audit checks consistency within the precision of the supplied tables rather than overwriting source values to force exact equality. Derived paired means and leave-one-out ranges are labeled as reconstructions from printed scores. No confidence intervals, additional seeds, or statistical significance claims are inferred from these rounded observations.

Artifact scope. The write-time replay reports five tasks for each of three conditions under both evaluators. Training-stage and frozen-store figures use reported Persona-B aggregate diagnostics, not synthetic learning curves. The held-out result curves show observed score profiles or explicitly defined arithmetic sensitivity.

The companion repository versions the core implementation and baseline adapters, frozen persona and benchmark assets, final score tables, plotting scripts, and paper figures. The numerical audit connects the released task-level records to the reported tables and figures, providing deterministic reconstruction of the published score analyses.