Memory Beyond Recall: A Dual-Process Cognitive Memory System for Self-Evolving LLM Agents

Anonymous ACL submission

Abstract

Long-term memory for an LLM agent is more than retrieving the right passage at the right time. Current memory systems collapse belief revision, causal coupling, and cross-domain abstraction into a single retrieval surface tuned for surface recall, and consequently struggle on implicit personalisation that requires reasoning over how a user has evolved. We propose DPCM, which reorganises agent memory along a *cognitive capability hierarchy* ascending from raw inputs and atomic facts, through diachronic belief trajectories and identity, to domain schemas, latent intentions and cross-domain patterns. The hierarchy is driven by two processes inheriting the architectural split of dual-process theory: a synchronous daytime writer (SYSTEM 1) that records belief revisions as doubly linked supersedes chains, and an asynchronous nighttime engine (SYSTEM 2) that induces schemas and intentions and sweeps for cross-domain collisions abstracted into higher-level core schemas. On LongMemEval, PersonaMem and PersonaMem-v2, enabling SYSTEM 2 contributes most where the benchmark rewards implicit cross-session inference (up to $+5.20$ on PersonaMem-v2) and least on span recall, matching the architectural prediction.

1 Introduction

An LLM agent that has spoken with the same user for weeks accumulates a long, evolving record of what the user said, did, and preferred, yet still confidently misanswers basic personalisation queries that a careful human assistant would not. Turning that record into useful long-term memory is currently treated as a retrieval problem, with vector stores, temporal knowledge graphs and update-in-place layers all engineered to surface relevant text chunks at query time (Packer et al., 2023; Chhikara et al., 2025; Rasmussen et al., 2025; Zhang and Wang, 2026; Feng et al., 2026; Liu et al., 2026).

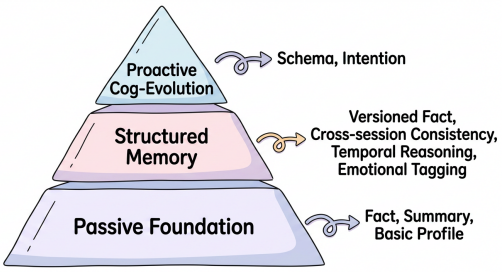

Figure 1: Capability hierarchy for LLM-agent memory, from passive foundation (raw inputs and facts), through structured memory (identity and supersedes chains), to proactive cognitive evolution (schemas, intentions and cross-domain core schemas). Existing systems mostly occupy the base.

An agent that knows its user well must model more than what the user has said: it must track *how* a stated belief was revised, *why* a preference shifted, and *what latent patterns* link behaviours across seemingly unrelated life domains. Current systems collapse all three into a single retrieval surface and lose the intermediate states a belief passed through, the causal links that make those states retrievable from a topically distant query, and the higher-order schemas that recur across life domains (Gärdenfors, 1988; Tulving and Thomson, 1973; Fiske and Taylor, 2013; Bartlett, 1932).

We trace these gaps to a single root: the conflation of memory-as-storage with memory-as-cognition. We organise agent memory along a *cognitive capability hierarchy* that ascends from raw inputs and atomic *facts*, through stable user *identity* items and diachronic *supersedes* chains over the fact store, to within-domain behavioural *schemas*, latent *intentions*, and cross-domain core schemas induced on top of them. Read at the level of capability rather than implementation (Figure 1), the hierarchy traces an arc from *passive foundation* through *structured memory* to *proactive cognitive evolution*, informed by Theory-of-Mind develop-

ment (Wellman, 2014) and dual-process accounts of cognition (Kahneman, 2011) that motivate separating fast online encoding from slow offline abstraction. Existing memory systems almost all sit in the passive foundation tier.

We instantiate the hierarchy as DPCM, with two processes that inherit only the architectural distinction of dual-process theory. A synchronous *daytime writer* (SYSTEM 1) reconciles each new turn against the vector store and writes bidirectional pointers when a belief is superseded, recording revisions as doubly linked chains. An asynchronous *nighttime engine* (SYSTEM 2) clusters fresh facts, induces schemas and intentions over a graph, and sweeps for cross-domain collisions whose behavioural similarity is high but semantic similarity is low, abstracting them into higher-level core schemas. The split makes a testable prediction: SYSTEM 2 should contribute almost nothing to single-turn fact recall but should account for most of the gains on implicit cross-domain inference, which we revisit in Section 3.4.

Contributions. We (i) organise long-term agent memory by a *cognitive capability hierarchy* grounded in Theory-of-Mind and dual-process theory; (ii) introduce a synchronous writer that records belief-evolution trajectories as doubly linked supersedes chains over a flat vector store, with a read path that requires only embedding lookup and pointer traversal; and (iii) design an asynchronous abstraction engine whose induction step and cross-domain sweeper jointly populate schema and intention layers and surface latent cross-domain patterns. We instantiate the design as $\text{DPCM}_{\text{lite}}$ (SYSTEM 1 only) and $\text{DPCM}_{\text{full}}$ (with SYSTEM 2), and evaluate them on LongMemEval, PersonaMem and PersonaMem-v2 with stage-wise ablations.

2 Method

Figure 2 summarises the architecture: DPCM runs two coupled processes over a shared store. The synchronous SYSTEM 1 daytime writer (left) is invoked on demand whenever the user or the application issues an explicit `add_memory` request, persisting raw input and writing reconciled facts and identity items with bidirectional supersedes pointers under a few-second budget. The asynchronous SYSTEM 2 nighttime engine (right) runs during idle or nightly windows: a no-LLM preprocess triages and clusters fresh facts, an induction agent M_a equipped with four tools writes schemas, intentions and structural edges into the graph $\mathcal{G}$, and a cross-domain sweeper induces higher-level core schemas from behaviour-similar but semantically-distant schema pairs. The read path traverses the vector store, supersedes pointers, and $\mathcal{G}$ deterministically without invoking any LLM. We next introduce the cognitive capability hierarchy (Section 2.1) and the two processes that instantiate it (Sections 2.2–2.3); read path and complexity follow in Sections 2.4–2.5. Full pseudocode is in Algorithms 1–2.

2.1 Cognitive Capability Hierarchy

DPCM organises memory along an ascending hierarchy of capabilities. The base persists raw inputs in a vector store and guarantees zero loss before any extraction is attempted. One level up, a fact store contains atomic facts extracted from the raw inputs, each potentially carrying SUPERSEDES/SUPERSEDEDBY pointers that link revisions into doubly linked chains. Alongside the fact store, an identity store holds stable user attributes and durable preferences. The upper tier lives in a separate graph $\mathcal{G}$ and contains two co-equal kinds of nodes: *schemas* summarise observed behavioural regularities within a single life domain, while *intentions* encode latent concerns about events the user is likely to face in the future and can be used by the agent for proactive care; intentions sit beside, not above, schemas. A cross-domain sweeper induces higher-level core schemas on top of within-domain schemas. Each graph node carries `evidence_vdb_ref` edges back to its supporting facts. In total the system maintains five kinds of memory content: facts, identity items, schemas, intentions, and core schemas.

2.2 System 1: Synchronous Daytime Writer

SYSTEM 1 is triggered *synchronously and on demand*, not on every dialogue turn: the short-term conversational context already sits in the model's window, so persisting a memory is something the user or the application chooses to do via an explicit `add_memory` request. When such a request arrives, the writer executes immediately under a tight latency budget in four steps.

(i) Raw persist. The raw input is written to the raw-input vector store as the immutable ground truth before any LLM call, so a downstream failure cannot lose user content.

(ii) Extraction. An extraction LLM M_e reads the current turn together with a short recent-history

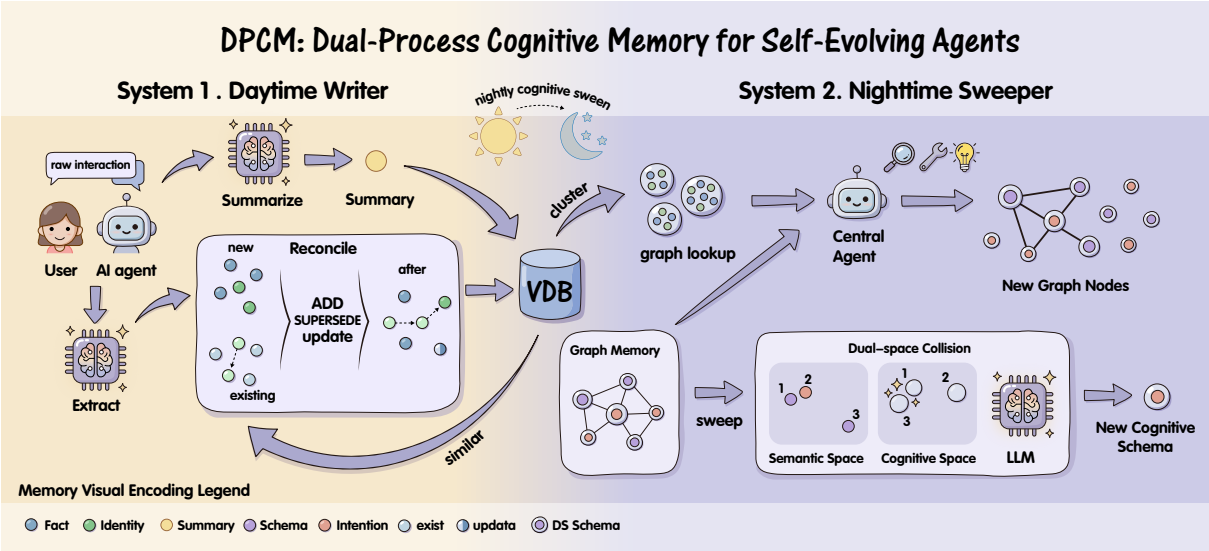

Figure 2: Overview of DPCM. A synchronous SYSTEM 1 daytime writer (left) handles on-demand `add_memory` requests, while an asynchronous SYSTEM 2 nighttime engine (right) induces schemas, intentions and cross-domain core schemas. The vector database (centre) holds raw inputs, facts and identity items. See Section 2 for details.

window and emits, in a single call, *identity items* (long-lived user attributes) and *atomic facts* (turn-level statements), each tagged with type, topic, salience and timestamp. We use `qwen-3.5-a17b`, `deepseek-v4-flash` or `kimi-2.5` as M_e, all in the no-think setting; temperature is 0 except for `kimi-2.5` (τ=0.6, its default).

(iii) Reconciliation. A reconcile LLM M_r takes each extracted item with its top-k nearest items in the vector store and emits one of ADD, SUPERSEDE, UPDATE, or NOOP in a single batched call sharing M_e's backbone and decoding setting.

(iv) Batch upsert and supersedes chain. Items kept by ADD/SUPERSEDE/UPDATE are batch-embedded and upserted. On SUPERSEDE the writer adds two pointers, n_{new}.SUPERSEDES=n_{old} and n_{old}.SUPERSEDEDBY=n_{new}, and does *not* delete the predecessor, so the chain forms a doubly linked list of revisions over a flat vector store. This single design choice distinguishes DPCM from update-in-place systems and is the architectural source of diachronic continuity: a single hit at read time triggers an $O(k)$ traversal that recovers the full evolution arc, including the causal middle that embedding similarity typically misses.

2.3 System 2: Asynchronous Nighttime Engine

SYSTEM 2 runs *asynchronously* on a schedule (idle periods or nightly), with a much looser latency budget (tens of seconds). It operates over the stores accumulated by SYSTEM 1 and writes back into the graph $\mathcal{G}$ in three phases.

Phase 1 (Preprocess, no LLM). A *fact triage* selects items not yet incorporated into any schema or intention node, then clusters them with two-stage DBSCAN (an initial pass over semantic embeddings, followed by a tighter pass that subdivides over-sized clusters). For each cluster, we compute a centroid embedding and retrieve *graph context* in both directions: the nearest existing schema and intention nodes, and any nodes whose evidence edges already point into the cluster's fact ids. Phase 1 makes no LLM calls.

Phase 2 (LLM-agent induction). An induction agent M_a is invoked once per cluster with four tools: CREATE_SCHEMA creates a within-domain behavioural-regularity node (e.g. work habits, food preferences); CREATE_INTENTION creates a latent-concern node about events the user is likely to face in the future, usable for proactive care; ADD_EVIDENCE attaches the cluster's fact ids to an existing schema or intention node via `evidence_vdb_ref`; and ADD_EDGE writes a structural edge between two graph nodes. All actions persist to $\mathcal{G}$ in emission order, so the graph evolves incrementally across nights rather than being rebuilt.

Phase 3 (Cross-domain sweeper). After per-cluster induction, the sweeper runs in three steps. (a) *Behavioural abstraction*: for each schema s, an LLM-based abstractor produces a short be-

havioural description (e.g. *"escalates commitment after sunk costs"*), embedded into a behavioural-cognitive space E_{beh}; a sentence encoder (Reimers and Gurevych, 2019) produces the semantic embedding E_{sem}. (b) *Collision scan*: the sweeper enumerates cross-domain schema pairs (s_i, s_j) and flags a collision when

$$\begin{aligned} &\cos(E_{\text{beh}}(s_i), E_{\text{beh}}(s_j)) > \theta_{\text{beh}}, \\ &\cos(E_{\text{sem}}(s_i), E_{\text{sem}}(s_j)) < \theta_{\text{sem}}. \end{aligned} \tag{1}$$

Collision pairs are aggregated with union-find, and trivially redundant within-domain pairs (high in both spaces) are filtered out. (c) *Core induction*: for each collision cluster, a final LLM call M_c takes the member schemas and induces a higher-level *core schema* written back into $\mathcal{G}$ with edges to all member schemas. Core schemas serve as the most abstract retrieval targets.

2.4 Read Path and Latency

At inference time, a query embedding hits the fact and identity stores by vector similarity, traverses any attached supersedes chain, and surfaces the schema, intention and core-schema nodes in $\mathcal{G}$ whose `evidence_vdb_ref` edges contain the matched fact ids. No LLM is invoked; total read latency is dominated by vector search.

2.5 Complexity

Let N be the number of fact items in the fact store, C the number of supersedes chains, $\bar{k}$ their average length, D the number of life domains, and $\bar{s}$ the average number of schemas per domain in the graph. The write-time cost of SYSTEM 1 is $O(N \log N)$ for nearest-neighbour lookup using an HNSW index, dominated by the extraction and reconcile LLM calls (each at most one batched call per turn). SYSTEM 2 performs Phase 1 preprocess at $O(N)$ similarity work with no LLM calls, Phase 2 induction at $O(|\mathcal{C}|)$ LLM calls where $|\mathcal{C}|$ is the number of clusters returned by two-stage DBSCAN, and Phase 3 sweeping at $O(D\bar{s})$ behavioural-abstraction LLM calls plus $O((D\bar{s})^2)$ vector comparisons and one core-schema LLM call per collision cluster. In practice $D \leq 20$ and $\bar{s} \leq 30$ for chat-assistant deployments we examined, keeping the quadratic term tractable. Read-time complexity is $O(\log N + \bar{k})$, namely a single approximate nearest-neighbour query plus chain traversal, and never invokes an LLM. Memory footprint scales as $O(N + C\bar{k} + D\bar{s})$, dominated by the fact store at typical chain depths $\bar{k} \leq 4$.

3 Experiments

3.1 Setup

We follow the no-think evaluation protocol throughout. Extraction and inference models are paired identically across systems for fairness, with $M_e, M_i \in \{$`qwen-3.5-a17b`, `deepseek-v4-flash`, `kimi-2.5`$\}$. The reconcile LLM M_r shares the same backbone as M_e. LongMemEval responses are judged by `gemini-3.0-flash` following the protocol of Wu et al. (2024).

Benchmarks. (1) **LongMemEval** (Wu et al., 2024), dataset S, evaluates recall, reasoning over time, knowledge update and abstention. (2) **PersonaMem (32k)** (Jiang et al., 2025a) evaluates persona-consistent response selection across multi-session histories. (3) **PersonaMem-v2 (32k)** (Jiang et al., 2025b) stresses *implicit* preference inference over 32k-token contexts, with frontier LLMs reaching only 37–48% accuracy in the original report (Jiang et al., 2025b), making this the most discriminative of the three.

Baselines. (B1) **Long-context agent** (Llama Team, 2024): the full conversation history is placed in the context window subject to model truncation. (B2) **Mem0** (Chhikara et al., 2025), evaluated with and without its graph backend. (B3) **Zep/Graphiti** (Rasmussen et al., 2025), the underlying temporal knowledge graph. We discuss concurrent self-evolving writers (Zhang and Wang, 2026; Feng et al., 2026; Liu et al., 2026) as related work in Section 4. We compare two configurations of our system. $\text{DPCM}_{\text{lite}}$ uses only the SYSTEM 1 daytime writer (raw, fact and identity stores with supersedes chains), and $\text{DPCM}_{\text{full}}$ additionally enables the SYSTEM 2 nighttime engine (schemas, intentions, and the cross-domain core-schema layer).

Diagnostic alignment. The three benchmarks above are not interchangeable; each isolates one of the gaps from Section 1. The *knowledge-update* subset of LongMemEval (Wu et al., 2024) probes **diachronic continuity**, since the ground-truth answer requires recovering an earlier belief state that was later revised, which update-in-place writers cannot supply. The same subset doubles as a **causal-coupling** probe via whether the revising turn is in the top-k retrieved chunks, since nearest-neighbour search frequently misses a turn whose embedding sits between topic and terminal belief. The *implicit-preference* split of PersonaMem-

Algorithm 1 SYSTEM 1 synchronous write pipeline

Require: `add_memory` request for user turn u, recent-history window w, extraction LLM M_e, reconcile LLM M_r, vector store $\mathcal{V}$ holding raw inputs, facts and identity items

$\textsc{InsertRaw}(\mathcal{V}, u)$ ▷ (i) raw persist, zero-loss
$(I, F) \leftarrow M_e(u, w)$ ▷ (ii) extract identity items I and atomic facts F
$A \leftarrow I \cup F$
$\mathcal{O} \leftarrow \emptyset$ ▷ batched reconcile decisions
for each item $a \in A$ **do**
  $\mathcal{N}_a \leftarrow \textsc{TopK}(\mathcal{V}, a)$ ▷ nearest existing items by hybrid embedding + topic key
  $\textsc{op}(a) \leftarrow M_r(a, \mathcal{N}_a) \in \{\text{ADD}, \text{SUPERSEDE}, \text{UPDATE}, \text{NOOP}\}$
  $\mathcal{O} \leftarrow \mathcal{O} \cup \{(a, \textsc{op}(a))\}$
end for
$A^+ \leftarrow \{a \mid (a, \textsc{op}) \in \mathcal{O}, \textsc{op} \neq \text{NOOP}\}$
$\textsc{BatchEmbedUpsert}(\mathcal{V}, A^+)$ ▷ (iv) batch embed and upsert into fact / identity stores
for all $(a, \text{SUPERSEDE}, n^*) \in \mathcal{O}$ **do** ▷ n^* is the targeted predecessor
  $a.\textsc{Supersedes} \leftarrow n^*$; $n^*.\textsc{SupersededBy} \leftarrow a$
end for

Algorithm 2 SYSTEM 2 asynchronous three-phase engine

Require: vector store $\mathcal{V}$, graph $\mathcal{G}$ (containing schema and intention nodes), induction agent M_a with four tools (CREATE_SCHEMA, CREATE_INTENTION, ADD_EVIDENCE, ADD_EDGE), behaviour-abstractor LLM M_b, core-schema LLM M_c, encoders $E_{\text{sem}}, E_{\text{beh}}$

Phase 1: Preprocess (no LLM)
$\mathcal{F}_{\text{fresh}} \leftarrow \{f \in \mathcal{V} \mid f \text{ is a fact not yet referenced by any schema or intention node}\}$ ▷ fact triage
$\mathcal{C} \leftarrow \textsc{TwoStageDBSCAN}(\mathcal{F}_{\text{fresh}})$
for all $c \in \mathcal{C}$ **do**
  $\mathcal{R}_c \leftarrow \textsc{ForwardNearest}(\mathcal{G}, \textsc{Centroid}(c)) \cup \textsc{BackwardByEvidence}(\mathcal{G}, c)$
end for

Phase 2: LLM Induction (single call per cluster)
for all $c \in \mathcal{C}$ **do**
  $O \leftarrow M_a(c, \mathcal{R}_c)$ ▷ actions $\in$ {CREATE_SCHEMA, CREATE_INTENTION, ADD_EVIDENCE, ADD_EDGE}
  for $o \in O$ **do**
    $\textsc{Apply}(\mathcal{G}, o)$ ▷ creates or strengthens schema / intention nodes, writes `evidence_vdb_ref` edges
  end for
end for

Phase 3: Cross-domain sweeper
for all schema node $s \in \mathcal{G}$ **do**
  $b_s \leftarrow M_b(s)$; $E_{\text{beh}}(s) \leftarrow \textsc{Encode}(b_s)$
end for
$\mathcal{P} \leftarrow \{(s_i, s_j) \mid \text{Eq.(1) holds, different domains}\}$
$\mathcal{K} \leftarrow \textsc{UnionFind}(\mathcal{P})$ ▷ aggregate collision pairs into clusters
for all $K \in \mathcal{K}$ **do**
  $s_{\text{core}} \leftarrow M_c(K)$; $\textsc{InsertCoreSchema}(\mathcal{G}, s_{\text{core}}, K)$
end for

v2 (Jiang et al., 2025b) probes **inductive abstraction** because its answers are never explicitly stated and must be inferred from behavioural regularities that recur across multiple life domains.

Implementation details. The vector store $\mathcal{V}$ is backed by Qdrant with HNSW indexing (m=32, ef=128). All embeddings use `bge-m3` (Chen et al., 2024) at 1024 dimensions, with a hybrid retrieval score that linearly interpolates dense cosine similarity and sparse BM25 (α=0.7). The top-k retrieval budget for the reconcile step is k=5. For SYSTEM 2, two-stage DBSCAN uses ε_1=0.35 on semantic embeddings for the initial pass and ε_2=0.25 for the tighter subdivision, with min_samples=3. The behavioural encoder E_{beh} and the semantic encoder E_{sem} share the same `bge-m3` backbone but operate on different inputs: E_{beh} encodes the LLM-generated behavioural description while E_{sem} encodes the original schema text. The collision thresholds are θ_{beh}=0.72 and θ_{sem}=0.40, tuned on a held-out split of PersonaMem-v2.

3.2 Main Results

Table 1 reports accuracy on the three benchmarks under two extraction/inference model pairs, `deepseek-v4-flash` and `kimi-2.5`, both run in no-think mode, which keeps the comparison closer to current deployed assistants than chain-of-thought evaluation would. The reported numbers are broadly consistent with the original papers (Wu et al., 2024; Jiang et al., 2025a,b; Chhikara et al., 2025; Rasmussen et al., 2025).

System	PersonaMem (32k)		LongMemEval (S)		PersonaMem-v2 (32k)	
	deepseek	kimi	deepseek	kimi	deepseek	kimi
B1 Long-Ctx Agent	72.07	77.59	**88.43**	80.61	**50.82**	58.46
B2 Mem0	69.07	65.23	56.84	48.87	42.53	41.25
+ graph	68.82	65.82	57.14	51.26	43.28	40.16
B3 Zep / Graphiti	58.93	64.21	67.83	62.35	39.54	38.27
$\text{DPCM}_{\text{lite}}$ (S1)	69.15	74.32	84.73	85.81	46.85	54.10
$\text{DPCM}_{\text{full}}$ (S1+S2)	70.46	**76.73**	85.17	**86.14**	49.36	**59.30**

Table 1: Main results. Accuracy (%) on three long-horizon memory benchmarks under two extraction/inference model pairs (`deepseek-v4-flash` and `kimi-2.5`, both no-think). Best result per column in bold; our two DPCM configurations are highlighted. The two PersonaMem columns probe fact recall and explicit preference recall, whereas PersonaMem-v2 stresses *implicit* preference inference and is the most discriminative setting (Jiang et al., 2025b).

The pattern across the three benchmarks is consistent with the architectural prediction in Section 1. On PersonaMem-v2 (the most diagnostic column), $\text{DPCM}_{\text{full}}$ reaches the strongest `kimi-2.5` result among memory-based systems and closes most of the gap to the long-context oracle under `deepseek-v4-flash`. The gain from enabling SYSTEM 2 ($\text{DPCM}_{\text{full}}$ versus $\text{DPCM}_{\text{lite}}$) is largest on PersonaMem-v2 (+2.51/+5.20), modest on PersonaMem (+1.31/+2.41), and small on LongMemEval (+0.44/+0.33) — the asymmetric pattern the dual-process hypothesis predicts. The long-context agent is strongest on the original LongMemEval and PersonaMem-v2 deepseek columns, which were partly constructed to test long-context models directly; failure cases on PersonaMem-v2 suggest lost-in-the-middle effects when relevant evidence sits at an unfavourable position, a regime where structured memory provides a sharper retrieval surface.

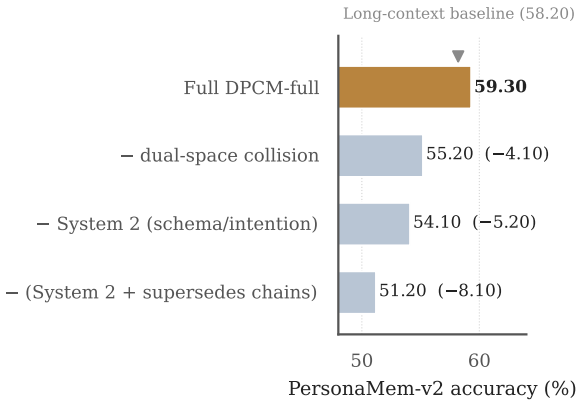

Figure 3: Ablation on PersonaMem-v2 with `kimi-2.5`. Each row removes one mechanism while keeping the rest of the pipeline intact. The triangle marker at the top indicates the long-context baseline (58.20). Numbers in parentheses are absolute drops versus the full system. See Section 3.3 for the row-by-row description.

3.3 Ablation

We ablate the SYSTEM 2 engine on PersonaMem-v2 using `kimi-2.5` as both M_e and M_i. Figure 3 reports the result alongside a strong long-context baseline (58.20) for the same model pair, computed by feeding the full session history into the inference model without any structured memory. The full $\text{DPCM}_{\text{full}}$ system exceeds this baseline by 1.10 points, and removing successive cognitive components degrades performance monotonically.

Two patterns are worth flagging. Dropping the dual-space collision alone (core-schema mechanism off) costs 4.10 points, whereas additionally dropping the schema and intention nodes costs only 1.10 more — the cross-domain collision step carries most of the upper-level contribution on this benchmark. Removing supersedes chains on top of disabling SYSTEM 2 costs an additional 2.90 points, the gap attributable to retaining diachronic structure in the writer. The figure thus separates three contributions: the supersedes mechanism ($\approx$ 2.9 pts), the schema and intention induction engine ($\approx$ 1.1 pts), and the cross-domain core-schema collision ($\approx$ 4.1 pts).

3.4 Analysis

Does the dual-process split do what the theory predicts? The architectural argument in Section 1 makes a falsifiable prediction: if SYSTEM 2 encodes a genuine cognitive role rather than a scheduling artefact, removing it should leave surface-recall benchmarks (LongMemEval, PersonaMem) essentially unchanged while collaps-

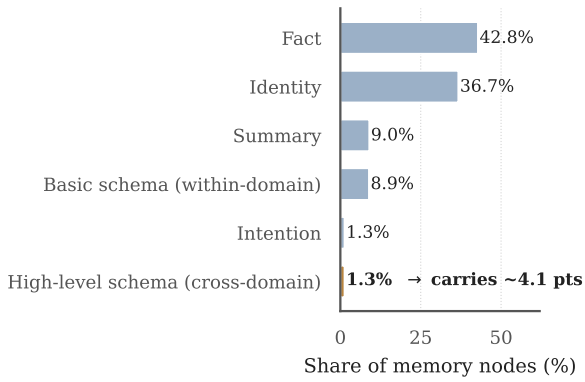

Figure 4: Distribution of memory node types in the $\text{DPCM}_{\text{full}}$ store on PersonaMem-v2 with `kimi-2.5`.

ing implicit cross-domain inference (PersonaMem-v2). A uniform drop across all three would instead suggest the dual-process framing is decorative. The $\text{DPCM}_{\text{full}}$ vs. $\text{DPCM}_{\text{lite}}$ gap in Table 1 is small on LongMemEval (+0.44/+0.33) and PersonaMem (+1.31/+2.41) but widens to +2.51/+5.20 on PersonaMem-v2, the asymmetric pattern the theory predicts; the ablation in Figure 3 sharpens the picture, with the cross-domain core-schema collision alone accounting for roughly four points on PersonaMem-v2.

Which kinds of memory actually live in the store? A natural concern is that the higher cognitive nodes might be empty most of the time, so the gains would reflect retrieval re-ranking rather than induced structure. Figure 4 reports the type distribution on PersonaMem-v2 with `kimi-2.5` across the five kinds of memory. Fact-level items dominate, as expected for a turn-by-turn writer; schemas of all kinds make up roughly 10% of the store, and the cross-domain core schemas surfaced by the dual-space sweep occupy only 1.3% of nodes, on par with intentions. That this 1.3% carries ≈ 4.1 ablation points (Figure 3) is consistent with the architectural claim that these few nodes encode high-leverage cognitive structure rather than additional surface content.

Why does DPCM exceed the long-context agent on LongMemEval kimi? The long-context agent falls 5.53 points below $\text{DPCM}_{\text{full}}$ on the kimi column (80.61 vs. 86.14). Failure cases show a dominant lost-in-the-middle pattern (Liu et al., 2024); structured memory side-steps this by promoting evidence to a dedicated retrieval surface.

Where current memory systems plateau. Figure 5 summarises per-question-type accuracy on LongMemEval (Wu et al., 2024) from public reports. Surface-recall types (user-fact, knowledge-update) are nearly saturated, while integration types (preference, multi-session, temporal) trail behind — the regime motivating the upper layers in DPCM.

Failure modes and cost. Misclassified PersonaMem-v2 items in development runs suggest supersedes chains can over-fire when the extraction or reconcile LLM treats a transient state as a stable revision, and θ_{beh} is benchmark-sensitive. The online SYSTEM 1 write path costs at most one extraction and one reconcile call per turn, comparable to retrieval-based baselines; SYSTEM 2 runs offline; read-time latency is identical to retrieval-only systems because no LLM is on the read path.

Supersedes chain statistics. Across the PersonaMem-v2 sessions, SYSTEM 1 writes an average of 1.8 supersedes chains per user (range 0–7), with a mean chain length of $\bar{k}$=2.4 revisions and a maximum of 5. On the knowledge-update subset of LongMemEval, 78% of correctly answered queries touch at least one supersedes pointer during read-time traversal, compared with 31% of incorrect ones, indicating that chain traversal is a strong predictor of successful belief-state recovery. Identity items are rarely superseded ($<3\%$ carry a pointer), consistent with their design as stable user attributes; the bulk of revisions occur on factual claims about preferences and habits.

3.5 Qualitative Illustration

We give two excerpts from PersonaMem-v2 runs. **Supersedes chain (fitness):** a user's stance on group fitness is revised five times across seven months, from n_1 *"finds working out with others a motivator that pushes them to new physical milestones"* through solitary, Zumba-return and Zumba-birthday phases to n_5 *"finds hiking more fulfilling than fitness classes, prefers welcoming interactive environments over competitive ones."* Update-in-place writers retain only n_5; DPCM preserves the full chain via SUPERSEDES pointers and exposes it at read time. **Cross-domain collision:** a wellness schema A and a gardening/teaching schema B, semantically distant in E_{sem}, are linked in E_{beh} and trigger Eq. (1). Both independently encode process-oriented stewardship at the behavioural level while sharing no surface vocabulary at the semantic level. The induced core schema reads *"User operates from an organic-cultivation mental model:*

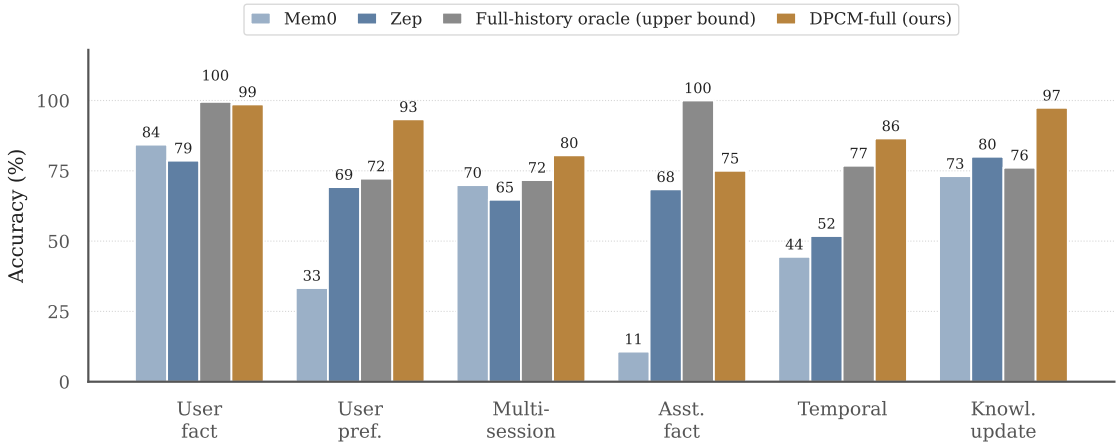

Figure 5: Reported LongMemEval per-question-type accuracy (%) of Mem0 v2 (Chhikara et al., 2025), Zep/Graphiti (Rasmussen et al., 2025), a full-history-in-context oracle, and $\text{DPCM}_{\text{full}}$. Mem0, Zep and the oracle numbers are taken from public reports. Question counts per type: 70 / 133 / 78 / 133 / 56 / 30.

meaningful outcomes are seen as emergent systems requiring patient, consistent micro-investment over time, with trust in accumulated attention over forced acceleration" — a cognitive abstraction no single within-domain schema captures. Without the dual-space test the two schemas would never be compared ($\cos(E_{\text{sem}})$=0.18, $\cos(E_{\text{beh}})$=0.81); the resulting core schema gives the agent a compact handle for proactive care across both domains.

4 Related Work

A first wave of agent memory work treats the problem as storage and graph organisation: MemGPT (Packer et al., 2023) mimics an OS page cache, Mem0 (Chhikara et al., 2025) uses update-in-place with an optional graph, Zep (Rasmussen et al., 2025) builds a temporal knowledge graph, and A-MEM (Xu et al., 2025), MemoryBank (Zhong et al., 2024) and HippoRAG (Gutiérrez et al., 2024) add Zettelkasten, forgetting-curve and hippocampal indexing respectively. A concurrent wave targets self-evolving stores: TSUBASA (Zhang and Wang, 2026), SEARL (Feng et al., 2026), FileGram (Liu et al., 2026) and the Experience Compression Spectrum (Zhang et al., 2026). DPCM is orthogonal, organising memory along a *cognitive-stages* axis with diachronic pointers and inductive abstraction above the fact layer. Retrieval-augmented generation (Lewis et al., 2020; Borgeaud et al., 2021; Gao et al., 2023) and long-context models (Llama Team, 2024) address a related but distinct static-corpus problem. Benchmarks include LongMemEval (Wu et al., 2024), LoCoMo (Maharana et al., 2024), PersonaMem (Jiang et al., 2025a,b), PerLTQA (Du et al., 2024), MSC (Xu et al., 2022) and LMEB (Zhao et al., 2026). Cognitive architectures — Generative Agents (Park et al., 2023), Voyager (Wang et al., 2024), Reflexion (Shinn et al., 2023), CoALA (Sumers et al., 2024) — organise memory at the working, episodic and procedural level; our perspective formalises an abstraction hierarchy *within* long-term memory, sharing the wake-sleep split with Xie (2026) but operating on consolidated facts. The hierarchy draws on Theory-of-Mind (Wellman, 2014; Apperly, 2012), dual-process theory (Kahneman, 2011; Evans and Stanovich, 2013), reconstructive memory (Bartlett, 1932; Schacter et al., 2012) and the episodic-semantic distinction (Tulving, 1972).

5 Conclusion

We argued that long-term memory for LLM agents should be organised by a *cognitive capability hierarchy* rather than by storage tiers, and presented DPCM as an instantiation. SYSTEM 1 captures belief revision over facts and identity items through a four-operation reconcile step with bidirectional supersedes pointers, while SYSTEM 2 lifts that structure into within-domain schemas, latent intentions and a cross-domain core-schema layer via a no-LLM preprocess, an LLM-induction phase and a dual-space sweeper. The read path remains LLM-free. Our evaluation shows the upper layers contribute most in the implicit-personalisation regime current systems handle least well, supporting a shift from *memory as recall* to *memory as cognition*.

Limitations

The current evaluation is restricted to English-language chat-assistant scenarios. The asynchronous SYSTEM 2 engine introduces an offline cost that grows with the size of the schema store, and the dual-space thresholds $(\theta_{\text{beh}}, \theta_{\text{sem}})$ are currently set by held-out tuning rather than learned. We do not study cross-lingual transfer, embodied agents, or settings in which user identity itself is uncertain. The hierarchy encodes an inductive bias that user behavior across life domains is governed by a small number of recurring cognitive patterns (Fiske and Taylor, 2013; Wellman, 2014); for users whose history is dominated by a single domain the cross-domain core schemas may give limited benefit, and in settings with high domain diversity the quadratic dual-space sweep becomes the binding cost.

Schema and pattern nodes are written in natural language, which inherits the verbosity and inconsistency of the extraction LLM's output. While interpretable, natural-language nodes are not amenable to algebraic operations such as set intersection over preference vectors. A hybrid representation that pairs natural-language summaries with structured slots is a natural extension compatible with the current hierarchy.

Ethics Statement

The proposed architecture stores progressively deeper inferred information about a user, including induced cognitive patterns the user may not have explicitly disclosed. While this enables richer personalization, it also raises privacy and consent considerations. Any deployment should expose the schema, intention and core-schema nodes to the user for inspection and removal, and should provide a hard-delete path that cascades through the supersedes chains. The benchmarks used in this work are all public and contain no real user data.

A second consideration concerns the asymmetry of inferred knowledge. Because core schemas can synthesize information across domains the user never explicitly linked, the system may surface inferences the user finds invasive even though every individual input was voluntarily disclosed. We recommend that production deployments offer per-layer opt-in granularity, with the cross-domain core-schema layer disabled by default for new users and re-enabled only after explicit consent that itemizes which behavioural patterns the system has currently inferred. We also caution against using core-schema patterns for any inferential task with high-stakes consequences (e.g. access decisions, eligibility scoring), since the patterns reflect statistical regularities in conversational behavior rather than verified ground truth.

References

Ian A. Apperly. 2012. What is "theory of mind"? concepts, cognitive processes and individual differences. *Quarterly Journal of Experimental Psychology*, 65(5):825–839.

Frederic C. Bartlett. 1932. *Remembering: A Study in Experimental and Social Psychology*. Cambridge University Press.

Sebastian Borgeaud, Arthur Mensch, Jordan Hoffmann, Trevor Cai, Eliza Rutherford, Katie Millican, George van den Driessche, Jean-Baptiste Lespiau, Bogdan Damoc, Aidan Clark, Diego de Las Casas, Aurelia Guy, Jacob Menick, Roman Ring, Tom Hennigan, Saffron Huang, Loren Maggiore, Chris Jones, Albin Cassirer, Andy Brock, Michela Paganini, Geoffrey Irving, Oriol Vinyals, Simon Osindero, Karen Simonyan, Jack W. Rae, Erich Elsen, and Laurent Sifre. 2021. Improving language models by retrieving from trillions of tokens. *arXiv preprint arXiv:2112.04426*.

Jianlv Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian, and Zheng Liu. 2024. BGE M3-Embedding: Multi-lingual, multi-functionality, multi-granularity text embeddings through self-knowledge distillation. *arXiv preprint arXiv:2402.03216*.

Prateek Chhikara, Dev Khant, Saket Aryan, Taranjeet Singh, and Deshraj Yadav. 2025. Mem0: Building production-ready ai agents with scalable long-term memory. *arXiv preprint arXiv:2504.19413*.

Yiming Du, Hongru Wang, Zhengyi Zhao, Bin Liang, Baojun Wang, Wanjun Zhong, Zezhong Wang, and Kam-Fai Wong. 2024. Perltqa: A personal long-term memory dataset for memory classification, retrieval, and synthesis in question answering. *arXiv preprint arXiv:2402.16288*.

Jonathan St. B. T. Evans and Keith E. Stanovich. 2013. Dual-process theories of higher cognition: Advancing the debate. *Perspectives on Psychological Science*, 8(3):223–241.

Xinshun Feng, Xinhao Song, Lijun Li, Gongshen Liu, and Jing Shao. 2026. SEARL: Joint optimization of policy and tool graph memory for self-evolving agents. *arXiv preprint arXiv:2604.07791*.

Susan T. Fiske and Shelley E. Taylor. 2013. *Social Cognition: From Brains to Culture*, 2nd edition. SAGE Publications, London.

Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, Meng Wang, and Haofen Wang. 2023. Retrieval-augmented generation for large language models: A survey. *arXiv preprint arXiv:2312.10997*.

Peter Gärdenfors. 1988. *Knowledge in Flux: Modeling the Dynamics of Epistemic States*. MIT Press, Cambridge, MA.

Bernal Jiménez Gutiérrez, Yiheng Shu, Yu Gu, Michihiro Yasunaga, and Yu Su. 2024. Hipporag: Neurobiologically inspired long-term memory for large language models. In *Advances in Neural Information Processing Systems (NeurIPS)*.

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

Bowen Jiang, Yuan Yuan, Maohao Shen, Zhuoqun Hao, Zhangchen Xu, Zichen Chen, Ziyi Liu, Anvesh Rao Vijjini, Jiashu He, Hanchao Yu, Radha Poovendran, Gregory Wornell, Lyle Ungar, Dan Roth, Sihao Chen, and Camillo Jose Taylor. 2025b. Personamem-v2: Towards personalized intelligence via learning implicit user personas and agentic memory. *arXiv preprint arXiv:2512.06688*.

Daniel Kahneman. 2011. *Thinking, Fast and Slow*. Farrar, Straus and Giroux, New York.

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. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *arXiv preprint arXiv:2005.11401*.

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

Shuai Liu, Shulin Tian, Kairui Hu, Yuhao Dong, Zhe Yang, Bo Li, Jingkang Yang, Chen Change Loy, and Ziwei Liu. 2026. FileGram: Grounding agent personalization in file-system behavioral traces. *arXiv preprint arXiv:2604.04901*.

Llama Team. 2024. The Llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

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

Charles Packer, Sarah Wooders, Kevin Lin, Vivian Fang, Shishir G. Patil, Ion Stoica, and Joseph E. Gonzalez. 2023. Memgpt: Towards llms as operating systems. *arXiv preprint arXiv:2310.08560*.

Joon Sung Park, Joseph C. O'Brien, Carrie J. Cai, Meredith Ringel Morris, Percy Liang, and Michael S. Bernstein. 2023. Generative agents: Interactive simulacra of human behavior. *arXiv preprint arXiv:2304.03442*.

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

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3980–3990.

Daniel L. Schacter, Donna R. Addis, Demis Hassabis, Victoria C. Martin, R. Nathan Spreng, and Karl K. Szpunar. 2012. The future of memory: Remembering, imagining, and the brain. *Neuron*, 76(4):677–694.

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

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

Endel Tulving. 1972. Episodic and semantic memory. In Endel Tulving and Wayne Donaldson, editors, *Organization of Memory*, pages 381–403. Academic Press, New York.

Endel Tulving and Donald M. Thomson. 1973. Encoding specificity and retrieval processes in episodic memory. *Psychological Review*, 80(5):352–373.

Guanzhi Wang, Yuqi Xie, Yunfan Jiang, Ajay Mandlekar, Chaowei Xiao, Yuke Zhu, Linxi Fan, and Anima Anandkumar. 2024. Voyager: An open-ended embodied agent with large language models. *Transactions on Machine Learning Research*.

Henry M. Wellman. 2014. *Making Minds: How Theory of Mind Develops*. Oxford University Press, New York.

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

Ying Xie. 2026. Learning to forget: Sleep-inspired memory consolidation for resolving proactive interference in large language models. *arXiv preprint arXiv:2603.14517*.

Jing Xu, Arthur Szlam, and Jason Weston. 2022. Beyond goldfish memory: Long-term open-domain conversation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (ACL)*.

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

Xing Zhang, Guanghui Wang, Yanwei Cui, Wei Qiu, Ziyuan Li, Bing Zhu, and Peiyang He. 2026. Experience compression spectrum: Unifying memory, skills, and rules in LLM agents. *arXiv preprint arXiv:2604.15877*.

Xinliang Frederick Zhang and Lu Wang. 2026. Tsubasa: Improving long-horizon personalization via evolving memory and self-learning with context distillation. *arXiv preprint arXiv:2604.07894*.

Xinping Zhao, Xinshuo Hu, Jiaxin Xu, Danyu Tang, Xin Zhang, Mengjia Zhou, Yan Zhong, Yao Zhou, Zifei Shan, Meishan Zhang, Baotian Hu, and Min Zhang. 2026. LMEB: Long-horizon memory embedding benchmark. *arXiv preprint arXiv:2603.12572*.

Wanjun Zhong, Lianghong Guo, Qiqi Gao, He Ye, and Yanlin Wang. 2024. Memorybank: Enhancing large language models with long-term memory. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*.