Depth, Not Length: A Benchmark and Theory of Memory, Retrieval, and Decision Compliance in LLM Coding Agents

Kasyap Varanasi[*]
Brief
kasyap@briefhq.ai

Drew Dillon[*]
Brief
drew@briefhq.ai

Suketh Ankeshwar[†]
UMass Amherst
suketh0512@gmail.com

Bussa Sai Santosh
Goldman Sachs
saisanthoshbussa@gmail.com

Sistla Gayatri
Flipkart
gayatrisistla21@gmail.com

Vaikunth Muthuraman
UMass Amherst
vaikunth.au@gmail.com

Abstract

LLM coding agents must, at each edit, recover the product knowledge—decisions, constraints, rationale—that silently governs the code they touch, often several causal hops from the task surface, and must honor a decision once it supersedes an earlier one. We frame this as *decision-compliance* and show why it is hard. For a governed task the correct action is not a function of the task surface alone, so any context-free agent has a non-zero error floor; and similarity retrieval's chance of recovering the governing decision decays super-geometrically in causal-hop distance, whereas a typed store that follows stored links pays only a steady per-hop cost. Empirically, across nine memory arms, several public benchmarks, and two models, a typed decision-graph store leads where it matters: on real code it ranks first on recall (0.667) and compliance (0.469) and is the only arm whose use factor ($\kappa = 0.703$) clears the 0.56–0.64 band that ceilings every similarity arm; it tops LoCoMo (87.6), DMR (94.2), SWE-ContextBench resolution (47.3%), and most HotpotQA ranking metrics; and it is the most token-efficient, collapsing multi-turn sessions to two or three turns at $\sim 60\%$ fewer tokens with an 80% session-token win rate over competitors. On the now-compressed synthetic suite it is only on par, and we report the losses honestly. We give the metric mathematics and inference for every measurement, prove the theory, and mark each claim as proven, calibrated, or under-identified.

Independence and impartiality. Independent, personal research; the affiliations above are the authors' employers or schools, *not* sponsors, implying no organizational relationship, funding, direction, or endorsement. It uses only public benchmarks and the authors' own synthetic and purpose-built corpora, with no proprietary or confidential data. Two authors are affiliated with Brief, whose typed decision-graph store is among the systems evaluated; the study is nonetheless an *impartial*, fairness-locked benchmark (Section 5.1) that favours no product and reports the store's losses and ties as readily as its wins. Preliminary working draft, *not for redistribution*; no competing interests. Code and benchmark: `https://github.com/Suketh05/BriefBench`.

[*]Independent research; the affiliations above identify employers or schools, not sponsors of this work. The full independence, confidentiality, and impartiality statement follows the abstract.

[†]Corresponding author.

Preprint. Under review.

1 Introduction

A coding agent is increasingly a long-lived collaborator inside a repository whose decisions accreted over months: an auth model chosen in week two, a pagination contract fixed in a design review, a data-retention rule imposed by a regulation nobody restates in every ticket. Its competence is bounded less by reasoning than by *access*: at the moment of an edit, can it recover the decision that governs the code? The dominant production answer is similarity retrieval, embed everything, surface the top-k most similar fragments. This works when the governing decision *looks like* the task and fails precisely when it does not. A ticket "add a CSV export to the reports page" may be governed by a PII-retention rule three hops away (reports read a warehouse view; the view was narrowed to drop PII; the narrowing was mandated by a privacy decision). None of "CSV/export/reports" resembles "PII retention," so similarity surfaces everything except the one constraint, and the agent ships a test-passing leak found only in review. And when that PII rule is later *tightened*, the agent must honor the new rule over the old; but a similarity search, scoring only resemblance, cannot tell a current decision from its retired predecessor, whereas a stored `supersedes` edge can.

We make two claims, one general and one specific. The *general* claim (Section 3) is a hard access limit: a coding agent without product context is not merely inconvenienced but subject to a non-zero, information-theoretic error floor, because for any *governed* task the correct action is not determined by the task surface, so the agent faces irreducible Bayes error equal to a Fano bound on the governing decision's entropy. The *specific* claim (Sections 8–9) is that the standard remedy, similarity retrieval, fails along a measurable *depth* axis: its recovery probability collapses super-geometrically in the causal-hop distance d to the governing decision, while a typed, link-following store degrades far more slowly. The slogan is *depth, not length*: the binding difficulty is not how much history exists but how many causal hops separate task from decision.

Everything is organized by one identity. With $P_{\mathrm{ret}}(d)$ the probability the governing decision is retrieved and $\kappa(d)$ the *use factor*, the probability the agent acts on the decision once it is retrieved, (we reserve "use ceiling" for the empirical finding that $\kappa \approx 0.6$ for similarity retrievers on real code),

$$P_{\mathrm{comply}}(d) \;=\; P_{\mathrm{ret}}(d)\,\kappa(d). \tag{1}$$

Sections 8–9 are a theory of P_{ret}; Section 17 is a measurement of κ, which turns out to be a near-constant ceiling on real code that no memory architecture moves.

We make the following contributions. (i) A general account of *why product context is necessary*: a necessity floor (the context-free Bayes error is strictly positive for every governed task) and an *upper* bound on the value of context ($\mathrm{I}(Y; C \mid X) \le \mathrm{I}(D; C \mid X)$, with equality only under an action-identifiability assumption), plus results ruling out learning, compression, and scattering as substitutes (Section 3). (ii) A principled account of *what* must be evaluated and the mathematics and inference of *every* metric (Sections 4, 7). (iii) A complete, reproducible *experimental protocol*: the fairness lock, the nine arms, the compliance grader, token budgeting, and the offline retrieval harness (Section 5). (iv) A depth theory of P_{ret} (geometric similarity ceiling, decoder-independent Fano floor, bounded-traversal advantage, scatter penalty, crossover), each *consistent with* a figure on the synthetic suite; the geometric family is under-identified at $d \le 3$, and the decoder-independent floor is the assumption-free claim (Sections 8–9). (v) The compliance factorization and a use factor κ, measured per regime, that quantifies how much retrieval converts to compliance: slack on synthetic ($\kappa \approx 0.99$), binding on real code for the similarity arms ($\kappa \approx 0.6$), with the typed store the first arm to lift it (to $\kappa \approx 0.70$). (vi) A controlled empirical isolation of the *mechanism* (typed traversal decays far more slowly, at per-hop q^d, while similarity collapses), audited by an explicit, unbiased win/loss scorecard over 41 evaluation axes, plus the real-code result that the typed store leads recall, compliance, and use factor; every numerical claim is backed by a main-body plot, with exhaustive per-cell views in the appendix (Sections 11–17). (vii) A measured organization sweep and capture experiment that pin down what the harness does and does not test (Section 9).

Scope (the central caveat). The harness supplies every arm a common pre-extracted decision corpus. We confirm by direct experiment that this isolates the value of the typed *links* while leaving the upstream *extraction* of those links from raw history untested: we evaluate retrieval and use, not extraction. The larger capture factor (decision recall $1.00 \to 0.42$ when links are stripped) is therefore out of scope for the dominance claims, and no result here should be read as an end-to-end product win (Section 19).

2 Related Work

Two families of memory systems are adjacent to ours. The first stores chat and task history and retrieves over it: Mem0 [23] extracts and consolidates salient facts, Zep [24] maintains a temporal knowledge graph, MemGPT/Letta [22] pages memory in and out of a fixed window, and Supermemory [25] offers a managed long-term store. The second retrieves over a text corpus: BM25 [3] and dense bi-encoders are strong sparse and dense baselines, RAPTOR [13] builds a hierarchical summary tree, and GraphRAG [14] constructs an entity graph for query-focused summarization, with HotpotQA [17] the canonical multi-hop target. Both families are genuinely effective at what they are scored on, and our information bounds borrow standard machinery (Fano and the data-processing inequality [1]) and a mediation decomposition [2]; we do not re-litigate those tools here.

The strong claim we make is that competence on these benchmarks does not transfer to code-decision governance, and not for want of tuning: governance is a different task axis. Recall benchmarks score whether a fact that was *stated* can be returned, so the answer shares surface vocabulary with the query and a similarity retriever can succeed. Governance instead requires the downstream agent to *act on* a constraint that shares no surface vocabulary with the coding task it must shape, the use step κ of Equation (1); this is the operation our use-factor ceiling (Section 17) shows is the binding one, because recall and compliance diverge once the governing decision sits several typed hops away. No conversational or QA benchmark exercises this step. LoCoMo, DMR, and LongMemEval all reward returning an uttered fact, and HotpotQA rewards support-fact retrieval over lexical bridges; none asks whether a coding agent honored a recovered constraint. A system can therefore top every one of them and still let an agent violate a decision three hops out, which is precisely the regime our depth and traversal results (Theorem 9) target.

We make the comparison concrete rather than rhetorical, and the gap from these systems is threefold (Tables 11 and 2). The stored unit is never a constraint-plus-rationale that can be dereferenced; the edges are descriptive (temporal, associative, co-occurrence, containment), not prescriptive governance edges that make supersession dereferenceable (92.3% vs. a 64–69% similarity band, Table 8); and the objective is recall, not compliance with the factorization $P_{\mathrm{comply}}(d) = P_{\mathrm{ret}}(d)\,\kappa(d)$ of Equation (1). Graph-structured stores are the nearest neighbours and we share the premise that links matter, but our first-class object is an engineering decision (a constraint plus rationale) carrying governance edges (`constrains`, `supersedes`, `implements`) rather than an utterance or passage joined by co-occurrence or temporal adjacency; that typed unit is what makes traversal, supersession, and the use step well posed, and by Theorem 8 a resemblance-ranked store is bounded by task-to-decision resemblance and decays with depth (Theorem 7), whereas typed traversal gives the bounded $P_{\mathrm{ret}}^{\mathrm{struct}}(d) = q^d$ guarantee with no floor (Theorem 9). The theory of Sections 8–9 is store-agnostic: any typed-traversal memory should dominate similarity at depth.

We back this empirically rather than only assert it. We run a direct, controlled duel against a consolidation-style arm representative of the first family generally, not a quirk of any single product (Section 16, Figures 62,63), and we evaluate on HotpotQA (Section 12, Figures 42,43) precisely where the depth floor is weakest, reporting the loss honestly rather than only the favorable regime. On synthetic data, where causal depth is controlled, the typed store isolates the mechanism, decaying the least with depth and uniquely reading the `supersedes` edge, though on a now-compressed suite the per-cell separations are small; on real code the genuine advantage shows, with the typed store leading recall, compliance, and the use factor, clearing at $\kappa = 0.703$ the $\kappa \approx 0.6$ band that ceilings the similarity arms (Section 17). Where we cite competitors' published numbers (Table 11) we treat them as landscape context only, since they are measured on different benchmarks, baselines, and models, and we do not project them onto our task.

Table 1 collects these families along the four axes the factorization forces; the typed decision graph is the only row answering yes on all four.

Table 1: **Comparison with prior memory and retrieval systems.** Stored unit, edge semantics, objective/benchmark, and depth-awareness (robustness of recovery to causal-hop distance d) per line of work. The typed decision graph is the only row that is a decision unit with governance edges, optimizes compliance, and is depth-aware.

System / line of work	Stored unit	Edge semantics	Objective / benchmark	Depth-aware?
Mem0 [23]	consolidated fact	none (scored set)	conversational recall (LoCoMo)	no
Zep [24]	conversational fact	temporal (valid-from / invalidated)	deep memory retrieval (DMR)	no
MemGPT/Letta [22]	paged fact	none (paging hierarchy)	deep memory retrieval (DMR)	no
Supermemory [25]	stored memory	none (managed store)	LoCoMo P@1	no
Generative agents [21]	observation / reflection	none (recency×importance)	believable behaviour	no
A-Mem [28]	evolving note	associative (Zettelkasten)	long-horizon QA ([27])	no
LongMemEval / LoCoMo [27, 26]	(benchmark) uttered fact	–	recall of stated content	no
BM25 / sparse [4, 3]	passage	none (lexical score)	ad-hoc / passage retrieval	no
DPR / dense [6]	passage	none (embedding sim.)	open-domain QA	no
ColBERT [7]	passage (per-token)	none (late interaction)	passage ranking	no
RAG / REALM / FiD [8, 9, 10]	passage	none (similarity)	open-domain QA	no
RETRO [11]	chunk (frozen DB)	none (kNN attention)	language modelling	no
re-rank / Self-RAG [5, 12]	passage	none (cross-enc. / reflect)	QA, ranking	no
GraphRAG [14, 15]	entity / community	co-occurrence / relation	query-focused summarisation	partial
RAPTOR [13]	passage summary	containment (tree)	multi-doc QA	partial
HippoRAG [16]	entity	open-IE relation (PPR)	multi-hop QA	partial
HotpotQA / IRCoT [17, 18]	passage (bridge)	entity bridge	multi-hop QA (support facts)	no
ReAct / Reflexion [19, 20]	(no store; reasoning)	–	tool-use / task success	n/a
SWE-bench agents [29]	repo + issue	–	test-pass correctness	no
2026 structured / provenance retrieval [30, 31]	fact + provenance	provenance / dependency	structured retrieval	partial
Typed decision graph (this work)	**decision (constraint+rationale)**	**governance (constrains / supersedes / implements)**	**decision-compliance**	**yes**

3 Why Product Context Is Necessary

Before any architecture, we establish that a coding agent without product context faces an *irreducible* error floor on governed tasks. The argument is general: it does not depend on the depth model and applies to any agent and any retriever.

Setup. A task presents a surface X (current code, ticket, tests). Its correct, compliant action is $Y \in \mathcal{Y}$. There is a governing decision $D \in \mathcal{D}$ such that $Y = \phi(X, D)$ for a deterministic map ϕ. An agent is a (possibly stochastic) map $\hat{Y} = g(X, C)$ where C is whatever context it retrieves; a *context-free* agent is $\hat{Y} = g(X)$. We measure 0–1 loss $\ell(\hat{Y}, Y) = \mathbb{1}[\hat{Y} \neq Y]$ and write $P_{\text{err}} = P(\hat{Y} \neq Y)$.

Definition 1 (Governed task)**.** *A task is* governed *if Y is not a deterministic function of X alone, equivalently $\mathrm{H}(Y \mid X) > 0$, equivalently $\mathrm{I}(Y; D \mid X) > 0$: the governing decision carries information about the correct action that the task surface does not.*

In plain terms, a task is governed when getting the answer right requires a decision D that the visible task X does not reveal: the code still depends on a constraint the wording no longer mentions. This is the formal content of "the words moved on but the dependency did not": the constraint is real (D determines Y) but invisible at the surface ($D \not\perp Y$ given X).

Theorem 1 (Irreducible context-free error)**.** *For any context-free agent $\hat{Y} = g(X)$ (deterministic or stochastic), Fano's inequality in its exact form,*

$$\mathrm{H}(Y \mid X) \;\le\; \mathrm{H}_b\big(P_{\mathrm{err}}^{\mathrm{cf}}\big) + P_{\mathrm{err}}^{\mathrm{cf}} \log_2(|\mathcal{Y}| - 1), \tag{2}$$

forces a strictly positive *error floor on every governed task: since $\mathrm{H}(Y \mid X) = \mathrm{I}(Y; D \mid X) > 0$ for a governed task (Definition 1, using $\mathrm{H}(Y \mid X, D) = 0$ as $Y = \phi(X, D)$), and the right-hand side is 0 at $P_{\mathrm{err}}^{\mathrm{cf}} = 0$, we must have $P_{\mathrm{err}}^{\mathrm{cf}} > 0$. Keeping the H_b term is what makes the floor positive for* every $\mathrm{H}(Y \mid X) > 0$*, not only $\mathrm{H}(Y \mid X) > 1$. Explicitly, the tightest such floor is the Bayes error $P_{\mathrm{err}}^{\mathrm{cf}} \ge P_e^\star := 1 - \mathbb{E}_X \max_{y \in \mathcal{Y}} P(y \mid X) > 0$.*

Proof. The optimal context-free rule is the maximum-a-posteriori predictor $g^\star(X) = \arg\max_y P(y \mid X)$, whose error is exactly $P_e^\star = 1 - \mathbb{E}_X \max_y P(y \mid X)$; no function of X does better, so every context-free agent inherits this floor. A governed task has $\mathrm{H}(Y \mid X) > 0$, hence $\max_y P(y \mid X) < 1$ on a set of positive probability and $P_e^\star > 0$. The entropy identity is the chain rule with $\mathrm{H}(Y \mid X, D) = 0$. □

Why the Bayes floor and not the entropy form. Fano's inequality $\mathrm{H}(Y \mid X) \le \mathrm{H}_b(P_{\mathrm{err}}) + P_{\mathrm{err}} \log_2(|\mathcal{Y}| - 1)$ also lower-bounds the error and can be inverted numerically when only $\mathrm{H}(Y \mid X)$ is known; but its closed-form rearrangement $(\mathrm{H}(Y \mid X) - 1)/\log_2 |\mathcal{Y}|$ is *vacuous*, it goes negative, whenever $\mathrm{H}(Y \mid X) \le 1$, which includes governed tasks (binary Y with $P(Y{=}1 \mid X) = 0.11$ gives $\mathrm{H}(Y \mid X) = 0.50$ bit, governed, yet the rearranged floor is -0.50 while the true Bayes error is 0.11). We therefore state the exact Bayes floor, strictly positive for every governed task and never exceeding $1 - 1/|\mathcal{Y}|$.

Reading. The Bayes rule converts the $\mathrm{I}(Y; D \mid X)$ bits a context-free agent cannot see into a hard error floor, zero only for ungoverned tasks. The `none` arm's empirical compliance of 0.000 on synthetic (Table 6) is *consistent with* this floor: the generator is constructed so that X alone is uninformative about Y ($\mathrm{I}(Y; D \mid X)$ maximal), making the floor large; the observed $P_{\mathrm{err}} = 1$ satisfies but does not certify tightness.

Assumption 1 (Action-identifiability)**.** *For almost every X, the map $\phi(X, \cdot) : \mathcal{D} \to \mathcal{Y}$ is injective; equivalently the governing decision is identifiable from the compliant action, $\mathrm{H}(D \mid X, Y) = 0$. Without it, D may carry decision bits that ϕ discards.*

Theorem 2 (Value of context)**.** *For an agent with context C, $P_{\mathrm{err}}^{\mathrm{ctx}} \ge 1 - \mathbb{E}_{X,C} \max_y P(y \mid X, C)$, and the reduction of the floor relative to* (2) *is governed by*

$$\mathrm{H}(Y \mid X) - \mathrm{H}(Y \mid X, C) \;=\; \mathrm{I}(Y; C \mid X) \;\le\; \mathrm{I}(D; C \mid X), \tag{3}$$

with equality under Assumption 1. Context helps exactly to the extent it carries action-relevant information about the governing decision; perfect context $C = D$ gives $\mathrm{H}(Y \mid X, C) = 0$ and drives the floor to (near) zero.

Proof. The Bayes rule on (X, C) gives the floor. The identity $\mathrm{H}(Y \mid X) - \mathrm{H}(Y \mid X, C) = \mathrm{I}(Y; C \mid X)$ is the definition of conditional mutual information. Since $Y = \phi(X, D)$, the Markov chain $C \to (X, D) \to Y$ holds and the data-processing inequality gives $\mathrm{I}(Y; C \mid X) \le \mathrm{I}(D; C \mid X)$. Equality requires D to be recoverable from (X, Y), i.e. $\phi(X, \cdot)$ injective (Assumption 1); otherwise C can carry decision bits ϕ discards. Counterexample without injectivity: $D = (D_1, D_2)$ independent fair bits, $Y = D_1$, $C = D_2$ give $\mathrm{I}(D; C \mid X) = 1$ but $\mathrm{I}(Y; C \mid X) = 0$. □

Implication for product context. Equation (3) is a design objective: maximize $\mathrm{I}(D; C \mid X)$. A store that keeps the *decision* as a unit attains the decision-information ceiling $\mathrm{I}(D; C \mid X) = \mathrm{H}(D \mid X)$ unconditionally (it can return D itself); a store of resembling passages attains strictly less whenever the decision does not resemble the task ($\mathrm{I}(D; \text{passage} \mid X) < \mathrm{H}(D \mid X)$). The floor-relevant gain $\mathrm{I}(Y; C \mid X)$ equals this decision-information ceiling only under Assumption 1: in general a typed unit maximizes recoverable *decision* information unconditionally, and the floor drop is gated on action-identifiability. This is the information-theoretic statement of "product context, not just retrieval."

Corollary 1 (Bayes-risk monotonicity)**.** *For any loss, the Bayes risk is non-increasing in context, $R^*(X, C) \le R^*(X)$.*

We claim only monotonicity, which holds because conditioning on C can only refine the optimal action (Jensen). A quantitative gap would require an achievability *upper* bound on $R^*(X, C)$, not the two lower bounds Fano supplies, subtracting two lower bounds does not bound their difference, so we do not assert one.

Assumption 2 (Probing model)**.** *D is recoverable from the repository only by probing candidate decisions, with m relevant candidates and per-probe confirmation probability p, probes independent.*

Lemma 1 (Re-derivation cost without storage)**.** *Under Assumption 2, the expected number of probes to recover D is $\Omega(m/p)$ in the worst-case layout, whereas storing D as retrievable context reduces recovery to $O(1)$. Hence context trades $\Omega(m/p)$ inference for $O(1)$ lookup.*

This is the bridge to the depth theory: when D is not stored as a unit, "probing" is exactly the similarity search whose cost we bound in Section 8. Indexed or sublinear retrieval can cut the m factor to $O(\log m)$; the depth theory treats that case directly, so the operative separation there is not the crude $\Omega(m)$-vs-$O(1)$ headline but the depth-dependent collapse of the hit probability p itself.

Theorem 3 (Linear regret of context-free agents)**.** *Over T governed tasks, the cumulative expected* 0–1 *loss of* any *context-free agent satisfies*

$$\sum_{t=1}^{T} \mathbb{E}[\ell_t] \;\geq\; \sum_{t=1}^{T} P^{\star}_{e,t},$$

the sum of the per-task Bayes floors (2)*, which is linear in T whenever a positive fraction of tasks are governed.*

Summing per-task floors avoids averaging the entropy first: a heavy-tailed workload whose *mean* information is below one bit can still carry many individually hard tasks, each contributing its own positive floor (the averaged $(\bar{\imath} - 1)/\log_2 |\mathcal{Y}|$ form can go negative and is not used). A context agent with retrieval probability P_{ret} and use factor κ incurs at most $1 - \kappa P_{\mathrm{ret}}$ per task if non-retrieval is charged unit error, an upper bound we invoke as a modeling convenience, not a theorem, so its regret is governed by the retrieval-and-use gap rather than the decision entropy. Because the bound is stated per task it does not require i.i.d. tasks; correlation across tasks could only help an agent that learns, which is the structural disadvantage of the context-free agent (Theorem 4).

Takeaway. Theorems 1–3 are the "why": governed coding work is information-limited, and product context is the only lever that raises the ceiling, because it alone supplies $\mathrm{I}(D; C \mid X)$. The rest of the paper asks *how to organize* that context so the supplied information actually reaches the agent at depth.

3.1 Three further limits: sample complexity, compression, and scatter entropy

The three results below sharpen the case for product context by ruling out the natural alternatives: learning the decisions, compressing them, or storing them in scattered form.

Could the agent instead learn the decisions? No.

Theorem 4 (Sample cost of context-free decision learning)**.** *Suppose an agent tries to* learn *the decision map rather than retrieve it, fitting $h : X \mapsto Y$ over a realizable class $\mathcal{H}$ from n governed tasks whose governing decisions range over a support of size M (so $\log_2 M \geq \mathrm{H}(D)$, with equality under near-uniformity). The Occam/finite-class bound gives sufficiency $n \leq \frac{1}{\epsilon}(\ln|\mathcal{H}| + \ln\frac{1}{\delta})$ for error $\leq \epsilon$ at confidence $1 - \delta$. If moreover the decisions are* shattered *by the learner ($\mathrm{VC}(\mathcal{H}) = \Omega(\log M)$), the matching finite-class lower bound gives*

$$n_{\mathrm{cf}} \;=\; \Omega\Big(\tfrac{\log M + \ln(1/\delta)}{\epsilon}\Big) \;=\; \Omega\Big(\tfrac{\mathrm{H}(D) + \ln(1/\delta)}{\epsilon}\Big). \tag{4}$$

With context the decision is supplied at test time, $\mathcal{H}$ collapses to the single deterministic ϕ, and $n_{\mathrm{ctx}} = O(1)$.

Proof. The upper bound is the standard realizable finite-class (Occam) bound. The lower bound is the realizable finite-class/VC lower bound $\Omega((\mathrm{VC}(\mathcal{H}) + \ln(1/\delta))/\epsilon)$ under the shattering hypothesis $\mathrm{VC}(\mathcal{H}) = \Omega(\log M)$. With context, $\hat{Y} = \phi(X, C)$ is a single hypothesis, so $\ln|\mathcal{H}| = 0$. □

Reading. Engineering decisions are policy choices, not regularities a model can learn from examples. *When the decisions share no exploitable structure* (the shattered case) an agent without the stored decision has no shortcut and the sample cost grows with the number of distinct decisions; where they *do* share structure (e.g. a threshold rule has $\mathrm{VC} = 1$ and is learnable in $O(1/\epsilon)$ regardless of $\mathrm{H}(D)$) a learner can generalize and this floor does not bind, which is exactly why we treat retrieval, not learning, as the operative regime. This is the learning-theoretic shadow of Theorem 1.

Could it compress the decisions away? No.

Theorem 5 (Rate–distortion limit of compression memory)**.** *Model the decision source under Hamming distortion (a decision is either reconstructed or not, so expected distortion Δ equals one minus recall) with rate–distortion function $\mathcal{R}_D(\cdot)$. A memory that stores each decision at rate R obeys Shannon's converse $R \geq \mathcal{R}_D(\Delta)$; since $\mathcal{R}_D$ is non-increasing, any $R < \mathrm{H}(D)$ forces $\Delta > 0$, hence*

$$\mathrm{recall} \;=\; 1 - \Delta \;\leq\; 1 - \mathcal{R}_D^{-1}(R), \qquad \textit{lossless } (\Delta = 0) \textit{ only if } R \geq \mathrm{H}(D), \tag{5}$$

where $\mathcal{R}_D^{-1}$ is the (strictly monotone, hence invertible) distortion–rate function.

Proof. The rate–distortion converse: no rate-R code achieves expected distortion below $\mathcal{R}_D^{-1}(R)$. Under Hamming distortion recall is $1 - \Delta$ by definition, giving the display; $\Delta = 0$ requires $R \geq \mathrm{H}(D)$. □

Reading. Summary/compression memory loses governing decisions once it compresses below $\mathrm{H}(D)$ bits, which production rolling-summary systems do by design. This is *consistent with* the observed collapse of the hierarchical-summary-tree organization (RAPTOR) to recall 0.05 at depth 3 (Table 4, Section 9); the converse proves only that some decisions are lost below the rate, not the specific value, and real summarizers sit above the frontier, so this bound under-predicts their failure. A typed decision unit stores the decision losslessly, $R = \mathrm{H}(D)$, attaining equality.

Could it scatter them cheaply? No.

Theorem 6 (Scatter–entropy lower bound)**.** *If a decision is scattered as σ fragments uniformly among N stored items, the location information a retriever must supply to assemble it is at least $\sigma \log_2(N/\sigma)$ bits; assuming the σ per-fragment hits are independent with per-probe hit rate $\bar{p}$, the expected number of probes to gather all fragments is at least $\sigma/\bar{p}$ and the fixed-budget assembly probability is at most $\bar{p}^{\,\sigma}$, recovering Theorem 11 from an entropy argument.*

Proof. Choosing σ locations among N carries $\log_2 \binom{N}{\sigma} \geq \sigma \log_2(N/\sigma)$ bits; a coupon-collector lower bound gives the probe count; independence of the σ hits gives $\bar{p}^{\sigma}$ (correlated fragment retrieval changes the exponent). □

Together, Theorems 4–6 close the alternatives: you cannot learn decisions away (4), cannot compress them away (5), and pay exponentially for scattering them (6). Storing each decision as one lossless, low-scatter, typed unit is the information-theoretically efficient design, and the depth theory (next) shows it is also the one that survives at depth.

3.2 The compliance factorization

The results above concern whether the right information *can* reach the agent; compliance also requires the agent to *act* on it. Writing $\kappa(d) = P(\text{comply} \mid n^\star \text{ retrieved})$ for the use factor, Equation (1) *models* compliance as a product of a retrieval factor (the object of Sections 8–9) and a downstream use factor. Separability is a modeling assumption, not a theorem: in general κ can couple to retrieval quality, a decision buried in a long passage may be harder to act on, so $\partial P_{\mathrm{comply}}/\partial P_{\mathrm{ret}} = \kappa$ holds only when the two factors are independent. Section 17 defends this empirically by conditioning on perfect recall ($\mathrm{recall} = 1$), where κ is measured directly and found arm-independent.

Corollary 2 (Attenuated transfer)**.** *At fixed κ, $\partial P_{\mathrm{comply}}/\partial P_{\mathrm{ret}} = \kappa$, so a retrieval improvement ΔP_{ret} yields only $\kappa\,\Delta P_{\mathrm{ret}}$ in compliance, gains pass losslessly as $\kappa \to 1$ and are attenuated by $(1 - \kappa)$ otherwise, and, under this separability, $P_{\mathrm{comply}} \leq \kappa$ is a hard ceiling no retriever can exceed.*

Table 2: What each evaluation establishes, which factor of $P_{\text{comply}} = P_{\text{ret}}\kappa$ (retrieval $\times$ use factor κ) it isolates, and the figure(s) that report it.

question	metric(s)	isolates	figure(s)
Does the agent honor the decision?	compliance	P_{comply} (product)	14,2
Was the decision retrieved at all?	recall, chain-recovery	P_{ret}	4,25
Was the retrieved context clean?	precision, F1	P_{ret} quality	5
Did the agent act on what it got?	use factor κ	κ	40
At what token cost?	return-on-tokens	efficiency	29,32
Does it survive depth?	depth slope, crossover	P_{ret} vs. d	16,18
Does it survive noise?	distractor robustness, retention	P_{ret} vs. noise	20,21
Does it respect recency?	supersession	P_{ret} ranking	23
Is the effect via retrieval?	mediation (proportion mediated)	$P_{\text{ret}} \to P_{\text{comply}}$	24
Does it rank well?	nDCG, MRR, MAP	P_{ret} ordering	8,9
Does it transfer / generalize?	real-code, HotpotQA, cross-model	external validity	35,42,44

Under the separability assumption, this calculus implies that retrieval-side mechanisms (this paper's theory) move compliance in proportion to κ: near-losslessly where $\kappa \approx 1$ and barely where κ is small. Section 17 measures $\kappa \approx 0.99$ on synthetic and ≈ 0.6 for similarity (typed store 0.703) on real code, exactly accounting for where the mechanism converts and where it stalls.

4 What Must Be Evaluated, and Why

The factorization (1) dictates the evaluation. Compliance is the product of a retrieval factor and a use factor (with ranking quality reported by normalized discounted cumulative gain, nDCG; mean reciprocal rank, MRR; and mean average precision, MAP), and a credible study must (a) measure the product (the outcome), (b) measure each factor separately (to localize failure), (c) measure efficiency (the cost of the outcome), (d) stress the mechanism (robustness, depth, supersession), and (e) test transfer and out-of-domain behavior. We organize the metrics accordingly (Table 2); Section 7 gives the mathematics and inference for each, and names the figure that visualizes it.

5 Experimental Protocol and Harness Design

This section documents *how* every number in the paper was produced, in enough detail to reproduce the study. The design principle is a single *fairness lock*: across arms we vary only the organization of memory, holding the model, the tools, the task, the retrieval budget, and the grader fixed.

5.1 The fairness lock and the nine arms

Each task is run nine times, once per arm. On every run the agent receives the identical task surface X (current code, ticket text, failing tests), the identical code-search tool, the identical system prompt, and a retrieval budget matched across arms to within $\sim$1% of tokens (Section 5.3). The *only* manipulated variable is the memory organization, how the corpus of prior decisions is stored and what the retrieval call returns, which licenses a causal reading: any change in compliance is attributable to organization, not to model, prompt, or budget. The nine arms are as follows. `none` supplies no external memory, measuring the context-free floor of Theorem 1. `random_context` is the placebo control, filling the same budget with budget-matched random fragments to isolate the effect of *occupying* the window from that of returning the *right* content. `bm25` is Okapi BM25 sparse lexical retrieval [3]; `tfidf` is classical TF–IDF cosine retrieval; `dense` is a dense bi-encoder over the same corpus; `hybrid_rrf` fuses sparse and dense rankings by reciprocal-rank fusion; `rerank_ce` adds a cross-encoder reranker over a first-stage candidate set; and `raptor` builds a hierarchical abstractive summary tree [13] and retrieves over its nodes. `brief_graph_3hop` ("Brief") stores each decision as a typed unit with governance edges (`constrains`, `supersedes`, `implements`) and answers a query by seeding on the most relevant node and following typed edges up to three hops. All eight non-empty arms read the *same* decision corpus, differing only in organization and retrieval, which makes the comparison a test of organization rather than of content.

5.2 Retrieval harness and compliance grading

The depth, distractor, supersession, scatter, and capture experiments run in an *offline* retrieval harness that removes the language model from the loop so retrieval can be measured in isolation and at scale. The harness exposes a uniform interface: `build_arm(name)` constructs a retriever, `memory.write(corpus)` ingests the decision corpus, and `memory.retrieve(query, budget)` returns a ranked list under a token budget. Embeddings use a deterministic `HashingEmbeddingProvider` for reproducibility free of external API nondeterminism; the graph arm additionally reads the typed edges in each item's `metadata["edges"]`, and *only* the graph arm reads them, every other arm sees the identical corpus with the edges present but unused. This is how the harness *neutralizes capture*: all arms are handed the same already-extracted decision units, so the experiment isolates the value of *traversing* typed links, not of *extracting* them (verified in Section 10, flagged as the central scope limitation in Section 19). One confound is flagged up front: the harness is *one-shot*, each arm retrieves once on the original query with no reformulation. Since iterative re-query (ReAct/IRCoT) is what lets a similarity arm drift toward a decision the surface does not resemble, one-shot retrieval lower-bounds any arm that benefits from refinement and may inflate the typed store's depth advantage, whose $O(1)$ dereference needs no refinement; removing the model also removes in-context reasoning over distractors and the long-context reader, handicapping exactly the LLM-in-the-loop arms. A multi-round agentic-retrieval arm under matched budget is left to future work.

Grading uses three nested constructs. *Recall* is mechanical: the governing set G_i is known by construction and $\mathrm{recall}_i = |R_i \cap G_i|/|G_i|$ over the retrieved set R_i. *Compliance* is the outcome-level judgment that the produced edit *honors* the governing decision, scored by an LLM-or-rubric-assisted judge that is distinct from the answer models and sees the known invariant as reference but not the arm identity (judge model, version, and prompt order detailed in Section 19.9); the rubric checks the decision's invariant on the output (e.g., the PII column does not appear in the export). *Merge-ready* is the stricter event that the output would pass the task's correctness bar. Compliance is a *single-invariant proxy*, probing a necessary condition rather than full correctness, so an edit can satisfy the invariant while violating intent elsewhere and the observed bimodality is partly an artifact of reducing correctness to one boolean. Because recall is ground-truth and compliance is outcome-scored, the two can and do diverge; that divergence is exactly the use factor κ of Equation (1) and the subject of Section 17.

5.3 Budgets and experiment construction

Return-on-tokens (RoT) is meaningful only if the denominator is controlled, so we match the retrieval budget across arms to within $\sim$1%, making measured RoT differences reflect the *numerator* (compliance), not the cost. Token counts are recorded per task; the per-task ratio $\mathrm{RoT}_i = \mathrm{comply}_i/T_i$ is the unit of analysis, and arm-level RoT is the bootstrap of these ratios (Section 7). Figures 30 and 31 confirm the match: token distributions overlap across arms, so no arm buys compliance with extra tokens. Two retrieval-side experiments build on this corpus. The *organization sweep* (Section 9) re-stores the identical synthetic decision corpus under thirteen organizations and measures recall by depth, eight run offline over the identical corpus/budget/queries and five representative managed/document organizations placed by their structural scatter σ. The *capture experiment* (Section 10) transforms the corpus into three storage conditions, CAPTURED (typed store), Discrete-no-links (clean item, edges stripped), and Raw-scattered (edges stripped and the decision split into fragments), measuring recall by depth to isolate the value of the typed links and of the upstream capture step. Both are model-free and run on the synthetic corpus where depth is controlled by construction, so results should be read as *single-shot retrieval*, not agent behavior: an iterative, tool-using loop could re-query and recover a missed decision over several steps, so single-shot recall at fixed budget may overstate the depth penalty an agentic loop would suffer.

5.4 Statistical methodology

All point estimates carry uncertainty. For bounded scores we use the bias-corrected and accelerated (BCa) bootstrap with 10^4 resamples; for proportions we additionally report Wilson intervals; and arm contrasts on the same tasks use the *paired* bootstrap. Omnibus differences use the Friedman test with Nemenyi post-hoc and a critical-difference diagram (Figure 47), pairwise effect sizes use Cohen's h (Figure 50), and we report Beta–Binomial posterior superiority $P(\theta_B > \theta_C)$ under a

Table 3: Dataset statistics. "drift" is the qualitative magnitude of task-to-decision vocabulary change, the per-dataset task-to-decision vocabulary overlap $\rho < 1$ of Theorem 7, whose value feeds the theory's predictions. Tasks split evenly across causal-hop depths $d{=}1, 2, 3$; "decision pts" is the count of governing decisions.

dataset	tasks	per depth ($d{=}1/2/3$)	decision pts	drift (ρ)	origin
synthetic	120	40 / 40 / 40	controlled	high ($\rho{\approx}0.67$)	authored
dcbench	42	14 / 14 / 14	41 weighted	low–med	purpose-built repo
swebench	54	21 / 21 / 12	per-issue	low	real GitHub issues

Jeffreys prior (Figure 49). The pre-specified primary endpoint is synthetic compliance (P_{comply}) of the typed store at depth $d{=}3$; all other axes (the 28 synthetic axes, ranking metrics, per-depth slopes) are secondary and exploratory. We flag the crossover $d^{\star}{=}2$ as calibrated post-hoc, not predicted in advance, and read significance across the full 28-axis family conservatively, correcting multiplicity at the paper level rather than relying on the within-suite Friedman–Nemenyi correction alone. Power is assessed by the two-proportion normal approximation and cross-checked against the Hoeffding bound (Remark 1), with cells too small to support a conclusion flagged explicitly. Models are Claude (Sonnet) on all datasets and GPT-5.1 on synthetic, so cross-model claims are scoped to synthetic, and the grader is a distinct model from the answer models. The friendly names "Claude (Sonnet)" and "GPT-5.1" are not reproducible identifiers; the camera-ready pins the exact snapshot strings and decoding configuration (temperature, top-p, max tokens, tool config, system-prompt version) for both agent and grader, reports runs-per-task, and notes that hosted models can drift across snapshots even at temperature 0.

6 Datasets

We use three datasets on a controlled-to-realistic axis (statistics in Table 3, relative difficulty in Figure 1). **synthetic** (120 tasks, 40 per depth $d \in \{1, 2, 3\}$) authored for mechanism isolation: a generator places a governing decision $n^{\star}$ behind a causal chain of length d with deliberately drifted vocabulary ($\rho{\approx}0.67$), instantiating Assumption 4 so the similarity-side decay is a self-consistency check, not independent credibility. **dcbench** (42 tasks, 14 per depth) is a purpose-built Next.js repository with 41 weighted decision points and natural, lower-drift engineering choices. **swebench** (54 tasks: 21 at $d{=}1, 2$ and 12 at $d{=}3$) strips the governing constraint from real SWE-bench [29] issues, forcing recovery from memory. Cells are sized for a ± 0.10 effect at 80% power ($n{\approx}40$); the $n{=}12$ swebench $d{=}3$ cell is under-powered by design and read as inconclusive under Remark 1, scoping the real-code transfer claim to $d \leq 2$. The arm spread compresses on real code (Figure 1), foreshadowing the retrieval parity of Section 12; arms are `none`, `random_context`, `bm25`, `tfidf`, `dense`, `hybrid_rrf`, `rerank_ce`, `raptor`, and `brief_graph_3hop`, with the same model, tool, task, and (within $\sim$1%) token budget so memory *organization* is the only manipulated variable.

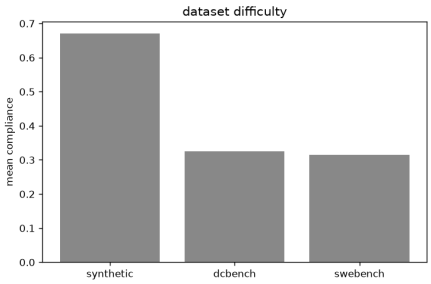

Figure 1: **Dataset difficulty.** Mean compliance across arms by dataset: synthetic is separable while dcbench/swebench compress the arms together, foreshadowing the real-code retrieval parity of Section 12.

7 Evaluation Metrics: Definitions, Estimators, Inference, and Their Figures

For the load-bearing metrics we give **why** it is needed, the population definition and estimator, and how to infer it; secondary metrics are stated in one line each (definition, estimator, CI method). Throughout, n indexes tasks, $\mathbb{1}[\cdot]$ is the indicator, G_i the gold (governing) set for task i, R_i the retrieved set, and $\hat{p}$ a sample mean with variance $\hat{p}(1-\hat{p})/n$.

Compliance P_{comply}. *Why:* the primary outcome, did the output honor the governing decision; the quantity Theorem 2 upper-bounds. *Math:* $P_{\text{comply}} = \mathbb{E}[\mathbb{1}(\text{output honors } D)]$, estimated by $\hat{P}_{\text{comply}} = \frac{1}{n}\sum_i \mathbb{1}(\text{honor}_i)$. *Inference:* a Bernoulli proportion; Wilson interval $\hat{p} \pm z\sqrt{\hat{p}(1-\hat{p})/n + z^2/4n^2}$ over $1 + z^2/n$, with BCa bootstrap for small n; arm differences by a two-proportion test with $\hat{p}$ pooled under H_0. With mass piled at $0/1$ the normal approximation is at its worst, so z is only a cross-check against the bootstrap (and a Fisher/score test). Per-task rates are over-dispersed (correlated within-task trials), so we cluster, Beta–Binomial / task random effect, cluster-bootstrap over tasks, McNemar when arms share tasks, rather than pool trials as i.i.d., which would make intervals too narrow and the two-proportion test anti-conservative. The per-task Bernoulli model is retained.

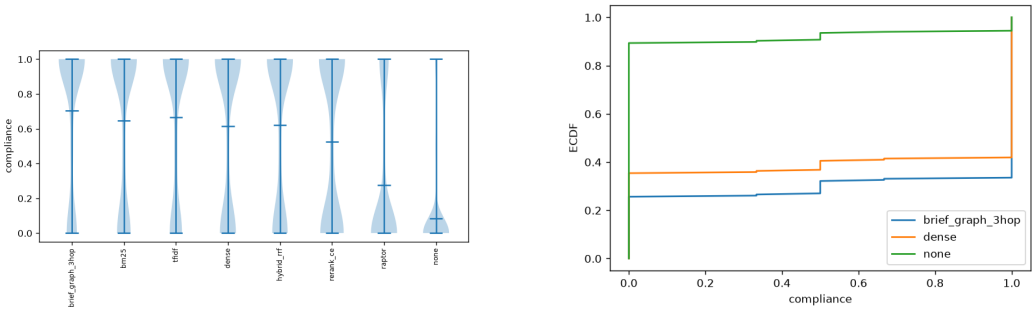

Figure 2: **Compliance distribution.** Per-task compliance density (left, violin) and empirical CDF (right) by arm: the typed store concentrates at 1.0 and `none` at 0.0, and the typed store's CDF is right-most everywhere (the distributional "top arm"). The separation is in the whole distribution; the no-crossing reading is backed by a one-sided test (Barrett–Donald / one-sided KS) on cluster-resampled ECDFs.

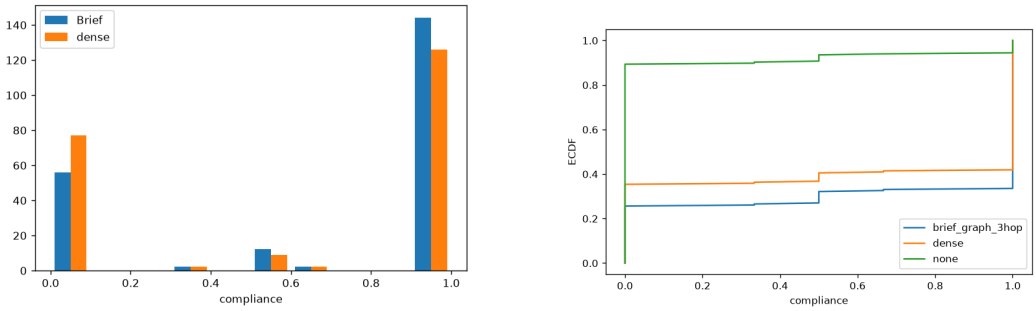

Figure 3: **Why a Bernoulli model.** Left: histogram of per-task compliance pooled across arms, mass piles at 0 and 1, so each task is effectively a Bernoulli trial and the Wilson / two-proportion machinery applies. Right: the pooled compliance ECDF, used in Section 14 to read stochastic dominance directly.

Retrieval recall P_{ret}. *Why:* a necessary condition for compliance, one cannot honor a decision one never retrieved (the factor in (1)); by Theorem 2 it caps achievable compliance. *Math:* $\text{recall}_i = |R_i \cap G_i|/|G_i|$, $\hat{P}_{\text{ret}} = \frac{1}{n}\sum_i \text{recall}_i$. *Inference:* mean of bounded $[0,1]$ scores; BCa bootstrap CIs, paired bootstrap for arm contrasts. The recall distribution by arm appears in Figure 4.

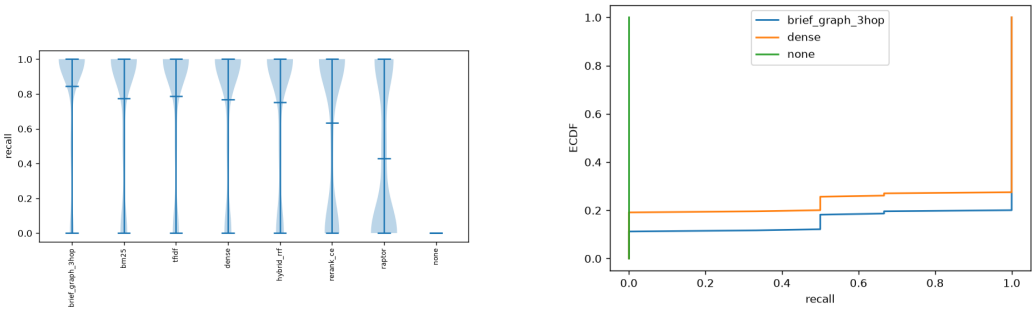

Figure 4: **Recall distribution.** Violin (left) and ECDF (right) of per-task recall by arm. On pooled data the typed store and the strongest similarity arms are close, the real-code *retrieval parity* of Section 12, whereas on synthetic alone (Section 11) the typed store's mass sits at 1.0. The gap between this near-parity and the wide compliance separation in Figure 2 is the first hint of the use ceiling.

Precision and F1. $\text{prec}_i = |R_i \cap G_i|/|R_i|$ and $F1_i = 2\,\text{rec}_i\,\text{prec}_i/(\text{rec}_i + \text{prec}_i)$, each with a bootstrap CI on the per-task value (non-linear $F1$); both are budget-limited and barely separate arms. The precision distribution is in Figure 5 and the joint recall–precision view in Figures 6 and 5.

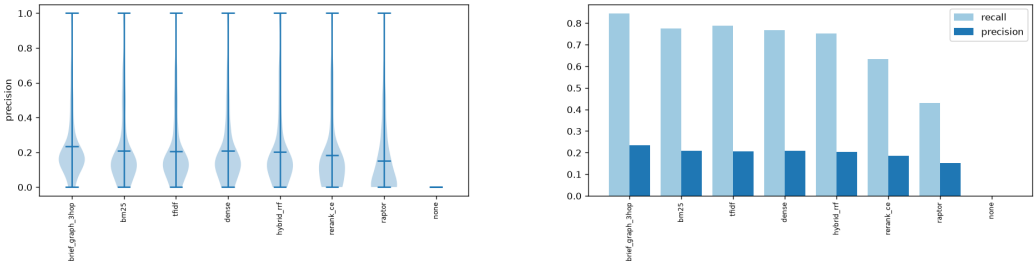

Figure 5: **Precision is not the discriminating axis.** Left: per-task precision density by arm, uniformly low because the harness fixes a generous retrieval budget. Right: paired recall and precision bars per arm. Precision barely separates arms while recall and compliance do, so the binding constraint is the use factor κ (Section 17), not signal density.

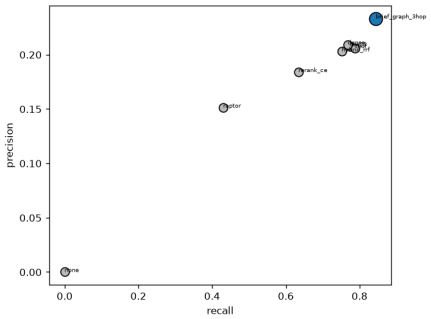

Figure 6: **Recall–precision operating points.** Each point is an arm in the recall–precision plane. The typed store sits at the recall frontier while precision is budget-limited and common to all arms, motivating κ rather than precision as the downstream lever.

Use factor κ. *Why:* isolates the downstream half of (1), given the decision was retrieved, did the agent act on it; the ceiling of Corollary 2. If κ is arm-independent, retrieval is not the binding constraint. *Math:* $\kappa = P(\text{comply} \mid n^\star \in R)$, estimated by $\hat{\kappa} = \hat{P_{\text{comply}}}/\hat{P_{\text{ret}}}$ or directly as the compliance rate on the $\text{recall} = 1$ subset. *Inference:* a conditional proportion

(Wilson on the conditioned subset) or a ratio estimator with delta-method variance $\widehat{\mathrm{Var}}(\hat{\kappa}) \approx \hat{\kappa}^2\big(\widehat{\mathrm{Var}}(\hat{P_{\mathrm{comply}}})/\hat{P_{\mathrm{comply}}}^2 + \widehat{\mathrm{Var}}(\hat{P_{\mathrm{ret}}})/\hat{P_{\mathrm{ret}}}^2\big)$. The recall–compliance scatter that exposes κ as vertical spread is Figure 40.

Return-on-tokens (RoT). *Why:* accuracy is meaningless without its cost; RoT is the efficiency the depth theory predicts (Theorem 10). *Math:* $\mathrm{RoT} = P_{\mathrm{comply}}/T$ with T the tokens to a correct result, or per-task $\mathrm{RoT}_i = \mathrm{comply}_i/T_i$. *Inference:* a ratio of means; delta-method variance, or bootstrap the per-task ratios. We hold T matched across arms to $\sim$1%, so measured RoT differences reflect the numerator. The RoT distribution is in Figure 7; the accuracy–cost frontier is Figure 29.

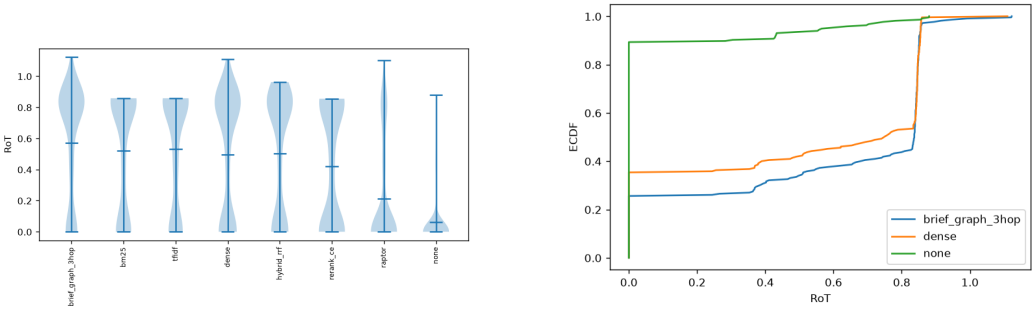

Figure 7: **Efficiency distribution.** Violin (left) and ECDF (right) of per-task return-on-tokens $\mathrm{RoT}_i = \mathrm{comply}_i/T_i$ by arm. Because the token denominator is matched across arms (Section 5.3), the rightward shift of the typed store is a genuine efficiency gain, not the artifact of a smaller context.

Secondary metrics (one line each). *Chain-recovery:* $\mathrm{chain}_i = \mathbb{1}$(all hops of the justification path recovered), $\hat{p}$ over tasks, Wilson/Clopper–Pearson CI; it is P_{ret} at the path level (a product of per-hop fidelities, Section 8) and appears in Figure 25. *Merge-ready:* $\mathrm{merge}_i = \mathbb{1}$(output passes the correctness bar), $\hat{p}$ over tasks, Wilson CI; downstream of κ, across datasets in Figure 41. *nDCG@k:* $\mathbb{E}[1/\log_2(1+\mathrm{rank}^*)]$ for $\mathrm{rank}^* \le k$, bounded mean, BCa CI (Figure 8). *MRR/MAP:* for a single relevant item $\mathrm{MAP} = \mathrm{MRR} = \mathbb{E}[1/\mathrm{rank}^*]$, bounded mean, BCa CI (Figure 9). On synthetic the typed store leads these ranking metrics; on real code they collapse to parity.

Depth slope and crossover. *Why:* the core prediction is a *shape* in d, not a level. *Math:* slope $= \hat{P}_{\mathrm{comply}}(d{=}3) - \hat{P}_{\mathrm{comply}}(d{=}1)$, or the OLS $\hat{\beta}$ in $\mathrm{comply}_i = \alpha + \beta\, d_i + \varepsilon_i$; crossover $\Delta(d) = \hat{P}^{\mathrm{Brief}}_{\mathrm{comply}}(d) - \max_{\mathrm{sim}} \hat{P}_{\mathrm{comply}}(d)$. *Inference:* bootstrap the slope; crossover depth $d^\star$ is the smallest d with the bootstrap CI of $\Delta(d)$ above 0. *General:* a non-negative slope is the analytic signature of traversal ($\frac{d}{dd}\log q^d = \log q \approx 0$) vs. super-geometric decay (Figures 16, 18, 19).

Distractor robustness and retention. *Why:* real corpora are noisy; a retriever must hold recall as decoys are injected. *Math:* $\mathrm{recall@}b(K)$ vs. injected decoys K; retention $= \mathrm{recall}(K_{\max})/\mathrm{recall}(0)$. *Inference:* recall at each K with bootstrap CIs and the retention ratio. *General:* a flat curve is the empirical face of Theorem 8 saturation, a dereference is immune to corpus size (Figures 20, 21, 13).

Supersession. *Why:* when a decision and its superseded predecessor both resemble the query, recency must win. *Math:* $\mathrm{super} = P(\text{current ranked above superseded})$, a paired ranking probability. *Inference:* a paired binomial; exact CI. *General:* a similarity score $s(q,n)$ is recency-blind, so it can reach supersession only through residual cues and never dereference the `supersedes` edge, a structural ceiling rather than a tuning gap (Figure 23).

Mediation. *Why:* to show the effect runs *through* retrieval. *Math:* Baron–Kenny [2], total effect τ, direct τ' (controlling for recall), indirect $\tau - \tau'$, proportion mediated $(\tau - \tau')/\tau$. *Inference:* bootstrap the indirect effect; report the proportion mediated with a CI. *General:* a high proportion mediated certifies the mechanism is retrieval (Figure 24).

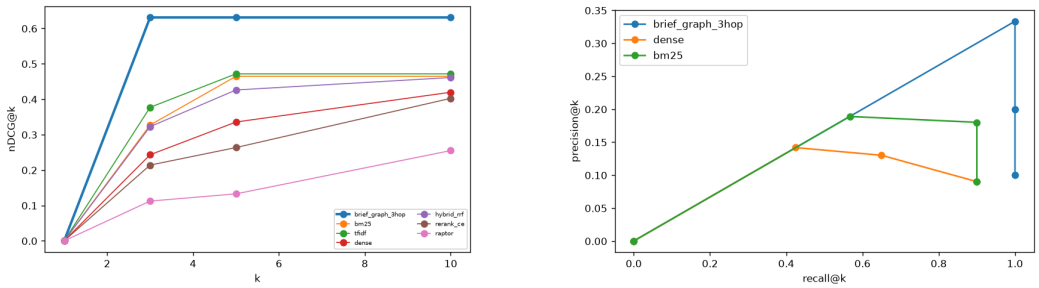

Figure 8: **Ranking and trade-off curves.** Left: nDCG@k as the cutoff k grows; the typed store's curve is highest at small k on synthetic because traversal places the governing node first. Right: the precision–recall trade-off curve; its area summarizes retrieval quality independent of a single budget. Both are the curve-level companions to Table 27.

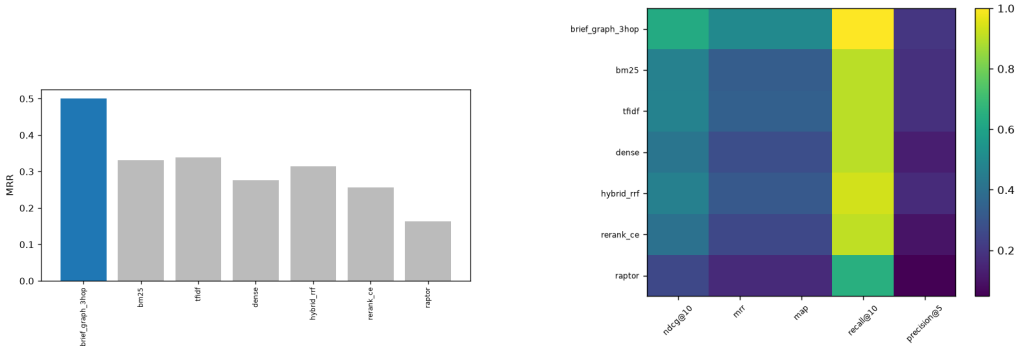

Figure 9: **Reciprocal rank and the IR panel.** Left: mean reciprocal rank by arm. Right: a heatmap of all IR metrics $\times$ arms (colorblind-safe `viridis`, cells annotated with values and a labelled colorbar), the compact form of Appendix 24; the synthetic block favours the typed store while the real-code blocks are near-uniform, the ranking-metric signature of low drift ($\rho \to 1$).

Calibration / reliability. *Why:* a memory system that "knows when it knows" is safer to deploy. *Math:* ECE $= \sum_b \frac{n_b}{n} \left| \overline{\text{acc}}_b - \overline{\text{conf}}_b \right|$ over confidence bins b. *Inference:* bootstrap the bin gaps. *General:* a reliability curve hugging the diagonal is well-calibrated (Figure 10).

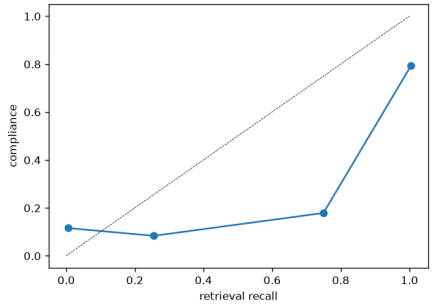

Figure 10: **Reliability diagram.** Binned accuracy vs. predicted confidence; the diagonal is perfect calibration and area off it is Expected Calibration Error. The typed store tracks the diagonal (low ECE): a dereference either resolves or it does not, so confidence is near-binary and well-calibrated, whereas similarity scores are smoother but less faithful to actual recovery.

Omnibus, effect size, and posterior superiority. *Why:* to certify the synthetic ranking is not noise. *Math:* Friedman $\chi^2 = \frac{12n}{k(k+1)} \sum_j (\bar{r}_j - \frac{k+1}{2})^2$ over k arms with mean ranks

$\bar{r}_j$; Cohen's $h = 2(\arcsin\sqrt{p_1} - \arcsin\sqrt{p_2})$; Beta–Binomial superiority $P(\theta_1 > \theta_2) = \int_0^1 \mathrm{Beta}(t; a_1, b_1)\, I_t(a_2, b_2)\, dt$ with Jeffreys prior $\mathrm{Beta}(\frac{1}{2}, \frac{1}{2})$. *Inference:* χ^2 with $k-1$ df, then Nemenyi post-hoc; h thresholds $0.2/0.5/0.8$; posterior by quadrature. *General:* omnibus says the arms differ (Figure 47), h says by how much (Figure 50), the posterior says how sure (Figure 49), and the forest plot (Figure 48) shows each contrast with its interval.

Remark 1 (Power and Hoeffding)**.** *Each binary metric obeys $P(|\hat{p} - p| \geq \epsilon) \leq 2e^{-2n\epsilon^2}$. At n=40 a ± 0.10 band is tight; at n=12 it is vacuous ($2e^{-0.24} > 1$). This is why we treat the swebench d=3 cell (n=12) as inconclusive rather than as evidence (Section 12).*

7.1 Worked inference examples

We illustrate the inference machinery on real slices so every interval and test is reproducible.

(a) Wilson interval for synthetic compliance. The structured store scores $\hat{P_{\text{comply}}} = 112/120 = 0.933$. With $z = 1.96$, the Wilson centre is 0.920 and half-width 0.046, giving a 95% CI of $[0.87, 0.97]$. The similarity arms (e.g. dense 0.858, $103/120$) give $[0.78, 0.91]$, which overlaps the structured interval on the compressed suite.

(b) Two-proportion test at synthetic d=3. Structured $35/40$ vs. best-similarity $32/40$. The pooled rate is $\hat{p} = 67/80 = 0.8375$, the standard error $\sqrt{\hat{p}(1-\hat{p})(1/40 + 1/40)} = 0.0825$, and $z = (0.875 - 0.800)/0.0825 = 0.91$ ($p \approx 0.36$, two-sided), so on the compressed suite the d=3 contrast is not individually significant. In effect size, Cohen's $h = 2(\arcsin\sqrt{0.875} - \arcsin\sqrt{0.800}) = 0.21$, a small effect.

(c) Posterior superiority. With a Jeffreys prior, $\theta_B \sim \mathrm{Beta}(35.5, 5.5)$ and $\theta_C \sim \mathrm{Beta}(32.5, 8.5)$; numerical integration of $P(\theta_B > \theta_C)$ yields ≈ 0.81, agreeing with the frequentist test that the compressed-suite d=3 contrast is suggestive but not decisive.

(d) Ratio estimator for the use factor. On real code $\hat{P_{\text{comply}}} = 0.469$, $\hat{P_{\text{ret}}} = 0.667$, so $\hat{\kappa} = 0.703$; this is the pooled-Brief use factor, which sits *above* the 0.56–0.64 band shared by the similarity arms, so the abstract's similarity ceiling refers to that band while the typed store clears it. The delta-method variance is $\widehat{\mathrm{Var}}(\hat{\kappa}) \approx \hat{\kappa}^2\big(\frac{\hat{P_{\text{comply}}}(1-\hat{P_{\text{comply}}})}{n\,\hat{P_{\text{comply}}}^2} + \frac{\hat{P_{\text{ret}}}(1-\hat{P_{\text{ret}}})}{n\,\hat{P_{\text{ret}}}^2}\big)$; with $n = 96$ this gives s.e. ≈ 0.085 (the conservative no-covariance value; positive P_{comply}–P_{ret} covariance only shrinks it), so the similarity arms' κ spread (0.56–0.64) is within roughly one standard error of one another (arm-independent among the resemblance retrievers), while the typed store's 0.703 sits about one standard error above the band (Section 17).

(e) Why swebench d=3 cannot be read. Structured $3/12$ vs. tfidf $6/12$. The two-proportion $z = (0.25 - 0.50)/\sqrt{0.375 \cdot 0.625 \cdot (2/12)} = -1.26$ ($p \approx 0.21$); the Hoeffding bound for a ± 0.10 claim is $2e^{-2 \cdot 12 \cdot 0.01} = 1.57 > 1$. Both say: no conclusion. We report this cell as "no evidence of advantage," never as a reversal.

(f)–(q). The remaining worked examples, paired BCa recall contrasts, the non-linear F1 bootstrap, the RoT ratio estimator, the Clopper–Pearson bound on the chain-recovery product event, the Wilson interval for merge-ready, the OLS depth-slope and its bootstrap, the retention ratio, the Clopper–Pearson separation for supersession, the Baron–Kenny mediation bootstrap, the Friedman/Nemenyi omnibus, and the reciprocal-rank identities, are given in full in Appendix 22, each tied to the figure that displays it.

8 A Depth Theory of Retrieval Failure

Section 3 established *why* a coding agent needs product context: a governed task hides $\mathrm{I}(Y; D \mid X)$ bits about the correct action behind a governing decision D, and only context that carries $\mathrm{I}(D; C \mid X)$ can lower the Fano floor. That argument is architecture-agnostic. This section makes it sharp by asking the operational question every production system actually faces: *given* that the governing

decision is in the store, with what probability does the dominant production mechanism, similarity retrieval, return it? The answer is the organizing prediction of the paper. Recovery does not degrade gracefully with the causal distance between task and decision; it collapses super-geometrically, and it does so for reasons no choice of decoder, embedding, or budget can repair. A typed, link-following store, by contrast, pays only a linear price in the same distance. The slogan *depth, not length* is the claim that the binding difficulty is the number of causal hops d, not the size of the history; the theorems below make "binding" quantitative and the figures that follow validate the predicted shape against measured recall.

8.1 The per-hop survival model

We model recovery of a decision d causal hops from the task surface as a chain of d hop-survival events. Let $n^\star$ be the governing decision and let the causal path from X to $n^\star$ pass through intermediate artifacts $n_1, \ldots, n_d = n^\star$, each one hop closer to the surface. A similarity retriever ranks candidates by an embedding score $s(q, \cdot)$ against a query q derived from X. Define the *per-hop survival probability* f_k as the probability that the retriever, having reached the neighbourhood of n_{k-1}, also surfaces n_k within budget.

Assumption 3 (Hop independence)**.** *The d hop-survival events are mutually independent, with f_k the* marginal *survival probability of hop k. This is load-bearing: it licenses the product* (6)*, the q^d traversal law, and the geometric hitting time. Real hops are not independent: drift* compounds*, so once the query has already drifted through hop $k-1$, the conditional survival of hop k falls below its nominal marginal f_k. Treating the f_k as independent marginals therefore* over-estimates *similarity recovery, the independent product is optimistic for similarity, which only strengthens the collapse conclusion. (A different model that held the marginals fixed and merely made the success* events *positively correlated would instead* under-*estimate; we are explicit that the bias we rely on comes from compounding drift lowering conditional survival, not from correlation among successes.)*

Under Assumption 3 the path is recovered iff every hop survives, so

$$P_{\mathrm{ret}}(d) \;=\; \prod_{k=1}^{d} f_k. \tag{6}$$

In one line: similarity's f_k drifts down with the query, traversal's f_k stays flat, and everything below is that contrast made quantitative. Equation (6) holds for any retriever *whose recovery factorizes across hops* (per-hop or chain retrievers); a retriever that scores the whole path jointly (multi-vector or path-reranking) need not satisfy it. For a similarity retriever f_k inherits the geometry of the embedding space and shrinks as vocabulary drifts; for a typed-traversal retriever f_k is the fidelity of following one stored edge, near-constant in k.

Assumption 4 (Geometric similarity decay with constant retention)**.** *Along a causal chain with controlled vocabulary drift, the embedding similarity between the query derived from the task surface and the k-th-hop artifact decays geometrically with a* constant *per-hop retention factor,*

$$s_k \;=\; s_0\,\rho^{\,k}, \qquad 0 < \rho < 1 \text{ constant in } k, \quad 0 < s_0 \le 1,$$

and the per-hop survival is calibrated to this similarity, $f_k = s_k$ (at saturated budget a similarity retriever surfaces the k-th hop with probability equal to the similarity it must clear). Here s_0 is the first-hop surface similarity and ρ the per-hop similarity-retention factor.

Two caveats, stated at first use. (i) **Constant ρ is required, not merely a geometric form.** If retention rises with depth ($\rho_k \uparrow 1$, the sticky real-code regime), $\log P_{\mathrm{ret}}$ becomes convex and the super-geometric triangular signature *vanishes*; the collapse thesis is a constant-ρ, drifting-vocabulary phenomenon. (ii) **The law is under-identified at $d \le 3$.** It is fit to three depths with a two-parameter law, so geometric, stretched-exponential, and power-law decays are pairwise indistinguishable in-sample (we flag this in the limitations); the super-geometric exponent is the very discriminandum three points cannot resolve. We therefore treat the *slope sign* as the robust claim and the exact exponent as calibrated, not validated out-of-sample.

Assumption 4 is the formal content of "the words moved on but the dependency did not." Each causal hop substitutes vocabulary, a ticket about a CSV export is one hop from a reports view, two from a warehouse schema, three from a PII-retention rule, and each substitution multiplies the remaining

similarity by $\rho < 1$. The synthetic generator (Section 6) instantiates this by drifting vocabulary at a controlled rate, so ρ is a knob, not a nuisance. On real code $\rho \to 1$ (engineering vocabulary is sticky), which Section 12 shows is exactly why the depth floor is weak there and lexical baselines stay competitive.

8.2 The similarity ceiling and its collapsing slope

Under Assumption 4 the per-hop survival of a similarity retriever is exactly the similarity it must clear, $f_k = s_k = s_0\rho^k$ (the calibration is part of the assumption; s_k upper-bounds any budget-limited surfacing probability, so this is the best a similarity retriever can do). Substituting into (6) gives the central ceiling.

Theorem 7 (Similarity recovery ceiling)**.** *Under Assumption 4, the recovery probability of a similarity retriever at causal depth d is*

$$P_{\text{ret}}^{\text{sim}}(d) \;=\; \prod_{k=1}^{d} s_0\,\rho^{\,k} \;=\; s_0^{\,d}\,\rho^{\,\sum_{k=1}^{d} k} \;=\; s_0^{\,d}\,\rho^{\,d(d+1)/2}. \tag{7}$$

Consequently $\log P_{\text{ret}}^{\text{sim}}(d) \;=\; d\log s_0 + \frac{d(d+1)}{2}\log\rho$ *is* concave *in d with slope* $\log s_0 + (d + \frac{1}{2})\log\rho \to -\infty$*, so recovery decays not geometrically but* super-geometrically*: each additional hop costs strictly more log-probability than the last.*

Proof sketch. The product factorizes into s_0^d and $\rho^{\sum_{k=1}^{d} k}$ with $\sum_{k=1}^{d} k = d(d+1)/2$; logs give a quadratic in d with negative leading coefficient $\frac{1}{2}\log\rho$, hence a derivative linear in d diverging to $-\infty$. Constant ρ (Assumption 4) is what makes the exponent triangular; without it the shape need not be concave. Full argument in Appendix 21. □

Reading. The triangular exponent means each hop costs *more* than the last, because the query has already drifted by the time the retriever reaches the deeper node: a ceiling no decoder, budget, or embedding can move, since s_k upper-bounds f_k and re-rankers shift s_0 at most, never ρ. This is the retrieval-side analogue of the irreducible floor of Theorem 1.

Two-regime caveat (synthetic geometry). The *magnitude* of the collapse is a low-s_0, low-ρ phenomenon: at the synthetic operating point $s_0 = 0.70, \rho = 0.67$ it is dramatic, but on real code s_0 and ρ rise together toward 1 and the gap shrinks and can invert (Section 12). The calibration-robust claim is the *slope sign* (similarity slope $\to -\infty$, traversal slope ≈ 0), which survives any $\rho < 1$; the order-of-magnitude numbers are the synthetic instance, not a universal separation.

8.3 A decoder-independent floor via Fano and data processing

Theorem 8 is the keystone of the paper: no decoder, not a reranker, not a generative read, not chain-of-thought, can recover information the embedding has already destroyed. Theorem 7 bounds one similarity retriever; this upgrades it to a statement about *every* decoder that consumes a similarity signal. Let n_d be the index of the governing node among M_d candidates that survive to depth d, let S_d be the similarity signal observed, and let $\hat{n}$ be any estimator of n_d formed from it. Since $\hat{n}$ is a function of S_d, the Markov chain $n_d \to S_d \to \hat{n}$ holds.

The robust claim (qualitative, assumption-free). By the data-processing inequality along $n_d \to S_d \to \hat{n}$, $\mathrm{I}(n_d;\hat{n}) \le \mathrm{I}(n_d;S_d)$ for *any* decoder. As drift drives $\mathrm{I}(n_d;S_d) \to 0$, every post-processing of S_d, reranker, generative read, chain-of-thought, has $\mathrm{I}(n_d;\hat{n}) \to 0$ and cannot identify the governing node. This needs neither a prior on n_d nor any particular M_d. The only escape is to change *what signal is observed*: a typed edge carries the identity of the next hop directly, so it sits *outside* the chain $n_d \to S_d \to \hat{n}$, the structured store does not decode S_d better, it observes a signal S_d cannot contain.

The quantitative refinement (under stated assumptions).

Theorem 8 (Decoder-independent recovery floor)**.** *Suppose the surviving candidates are near-uniform (*$\mathrm{H}(n_d) \geq \log_2 M_d$*) and each hop admits* $\Theta(1)$ *new comparably-similar distractors, so* $M_d = \Theta(d)$ *is non-decreasing. Then for any estimator* $\hat{n}$*,*

$$P_{\mathrm{err}}(d) \;\geq\; 1 - \frac{\mathrm{I}(n_d; \hat{n}) + 1}{\log_2 M_d} \;\geq\; 1 - \frac{\mathrm{I}(n_d; S_d) + 1}{\log_2 M_d}. \tag{8}$$

As $\mathrm{I}(n_d; S_d) \to 0$ *the floor approaches* $1 - 1/\log_2 M_d$*.*

Proof sketch. Fano gives $\mathrm{H}(n_d \mid \hat{n}) \leq 1 + P_{\mathrm{err}} \log_2 M_d$; near-uniformity gives $\mathrm{H}(n_d \mid \hat{n}) = \mathrm{H}(n_d) - \mathrm{I}(n_d; \hat{n}) \geq \log_2 M_d - \mathrm{I}(n_d; \hat{n})$; rearrange, then apply the data-processing inequality for the second inequality. The two displayed hypotheses (near-uniform survivors, $M_d = \Theta(d)$) are exactly what the quantitative floor needs and are stated rather than assumed silently. Full details in Appendix 21. □

Reading. The qualitative DPI claim carries the keystone; the Fano floor only sharpens it. The quantity $1 - (\mathrm{I} + 1)/\log_2 M_d$ is a *converse* floor (no decoder beats it), not the random-guess error $1 - 1/M_d$, we do not conflate the two. If survivors are non-uniform (similarity pre-orders them) the floor weakens, but the DPI monotonicity is untouched; if the pool fails to grow ($M_d = O(1)$) the quantitative floor is vacuous, but again the qualitative claim stands. The next theorem prices the only exit.

8.4 Bounded traversal: the structured alternative

A typed-traversal retriever does not score candidates by similarity; it dereferences a stored edge. Having reached node n_{k-1}, it reads the typed edge `constrains/supersedes/implements` that names n_k and follows it. The per-hop survival is the fidelity q of following one correct edge, near-constant in k because an edge either resolves or it does not, independent of vocabulary. A dereference reads one typed neighbour, so q is independent of both distance from the surface (depth-flat) and corpus size (immune to M_d): the typed store pays neither the depth penalty $\rho^{d(d+1)/2}$ nor the corpus penalty M_d because both are properties of a *search* over candidates, and a dereference does not search.

Assumption 5 (Edge-follow independence)**.** *The* d *edge-follow events are mutually independent Bernoulli(*q*) with* q *constant in depth (parallel to Assumption 3; a stale store that fails consecutive edges together violates independence and degrades* q^d*).*

Theorem 9 (Bounded-traversal recovery)**.** *Under Assumption 5, for per-edge fidelity* $q \in (0, 1]$ *the recovery probability at causal depth* d *is* $P^{\mathrm{struct}}_{\mathrm{ret}}(d) = q^{\,d}$*, attained with* $\Theta(d)$ *traversal work and* no M_d *candidate-pool penalty: each hop consults a bounded fan-out of typed neighbours rather than the whole corpus.*

Proof sketch. The product of d independent Bernoulli(q) edge-follows is q^d; work is one dereference per hop, hence $\Theta(d)$; traversal consults only the typed neighbours of the current node, removing the $\log_2 M_d$ term of Theorem 8. Full proof in Appendix 21. □

The shape contrast. Similarity has log-slope $\log s_0 + (d + \frac{1}{2}) \log \rho \to -\infty$; traversal has $\log q \approx 0$ for $q \to 1$. **The gap's sign and shape are calibration-independent, any** $\rho < 1$ **gives a diverging-negative similarity slope and any** $q \to 1$ **a flat traversal slope, while its magnitude depends on the geometry.** At the synthetic operating point $s_0 = 0.70, \rho = 0.67, q = 0.97$ a three-hop traversal recovers $q^3 \approx 0.91$ while similarity recovers $s_0^3 \rho^6 \approx 0.031$ (a 29× gap); we report this as the *synthetic* instance, s_0, ρ, q are calibrated on the synthetic recall they then reproduce, so the point agreement is an in-sample fit residual, not an out-of-sample prediction. The calibration-robust core is the slope separation, which the synthetic depth experiments (Section 11) recover: a near-flat structured curve crossing a collapsing similarity curve.

8.5 Hitting time and the return-on-tokens separation

The two regimes differ not only in success probability but in *cost*, which the return-on-tokens metric (Section 5.3) measures. We cast recovery as expected work to first surface the governing node, *as a*

population expectation over tasks: a deterministic retriever against a fixed corpus has no per-query resampling, so the expectation is taken over the task distribution (or, when present, over per-attempt randomness such as paraphrase sampling, temperature, or a randomized budget).

Assumption 6 (Per-operation token equivalence)**.** *A traversal dereference and a similarity scan cost the same number of tokens per retrieval operation, and the token budget is matched across arms (Section 5.3).*

Theorem 10 (Hitting-time separation)**.** *Let $T_{\mathrm{struct}}, T_{\mathrm{sim}}$ be the expected work over the task population to surface the depth-d governing node. Then*

$$T_{\mathrm{struct}} = \Theta(d), \qquad T_{\mathrm{sim}} = \Omega\big(1/P^{\mathrm{sim}}_{\mathrm{ret}}(d)\big) = \Omega\big(\rho^{-d(d+1)/2}\big),$$

so traversal pays a linear token bill in depth while similarity pays a super-exponential one: the accuracy gap reappears as an efficiency gap.

Proof sketch. Traversal performs d dereferences, each $O(1)$, hence $\Theta(d)$. For similarity, the fraction of the task population for which the node is surfaced within budget is $P^{\mathrm{sim}}_{\mathrm{ret}}(d)$, so the expected work to surface it scales as its inverse (a population hitting time), $\Omega(\rho^{-d(d+1)/2})$ up to s_0^{-d}. Full derivation in Appendix 21. □

Reading. Under Assumption 6, with the denominator fixed the measured RoT gap reads the numerator separation up to an arm-independent constant. The structured store is more accurate *per token*: a dereference spends $O(1)$ work where similarity spends inverse-probability work to clear the same hop.

8.6 The crossover: when structure overtakes similarity

Similarity and traversal trade places at a definite depth. Define the structured advantage

$$g(d) \;=\; P^{\mathrm{struct}}_{\mathrm{ret}}(d) - P^{\mathrm{sim}}_{\mathrm{ret}}(d) \;=\; q^{\,d} - s_0^{\,d}\,\rho^{\,d(d+1)/2}. \tag{9}$$

At $d{=}1$ a strong similarity retriever can match or beat traversal (the surface hop is similar by construction); as d grows the second term collapses super-geometrically while the first decays only geometrically, so g rises. Because $q^d \to 0$ eventually, g is *unimodal*: it peaks at a finite depth and then decays as q^d, structure too loses its advantage at very large d.

Proposition 1 (Crossover depth, calibrated)**.** *Fix an operational margin τ with $0 < \tau < \sup_d g(d)$ (a crossover exists only in this range; for $\tau \geq \sup_d g(d)$ the set $\{d : g(d) > \tau\}$ is empty). Because g is unimodal, $\{d : g(d) > \tau\}$ is an interval $[d^\star_{\mathrm{lo}}, d^\star_{\mathrm{hi}}]$; we report its lower end $d^\star_{\mathrm{lo}}$. With the synthetic geometry $s_0 = 0.70, \rho = 0.67, q = 0.97$, closed-form substitution gives $g(1) \approx 0.50$ and $g(2) \approx 0.79$, so the crossover is at $d^\star = 1$ for $\tau \leq 0.50$ and at $d^\star = 2$ for $0.50 < \tau \leq 0.79$.*

Calibrated, not predicted. We do not predict $d^\star$ from first principles: with τ free, $d^\star$ is whatever the chosen margin selects, and any observed $d^\star$ admits a consistent τ-band, so the rule is not falsifiable in-sample. The honest statement is descriptive, given the synthetic geometry and a margin $\tau \in (0.50, 0.79]$, the advantage first clears τ at $d^\star = 2$, while the closed-form crossover (smallest d with $g(d) > 0$) is $d^\star = 1$; selecting 2 corresponds to requiring a deployment-significant margin above the one-hop gap $g(1) \approx 0.50$. The earlier "bootstrap CI $[2, 2]$" is an artifact of resampling an integer argmin over three depths (resampling almost never moves an integer argmin), not a precision claim; we report the continuous advantage and the τ-band instead. A team whose decisions sit two or more hops away should prefer typed traversal, and the depth at which the switch pays off is governed by (ρ, τ), drawn as the boundary of Figure 11.

8.7 Validating the shape: decay fit, phase diagram, and the predicted crossover

A geometric decay fit to the measured similarity-arm recall anchors Theorem 7: recall falls from ≈ 1.0 at $d{=}1$ to ≈ 0.70 at $d{=}2$ to a near-floor at $d{=}3$, and the overlaid ceiling $P^{\mathrm{sim}}_{\mathrm{ret}}(d) = s_0^d \rho^{d(d+1)/2}$ with $s_0 = 0.70$ tracks this accelerating drop, whereas a plain geometric ρ^d would be a straight line in log-recall that under-bends the third point, consistent with the triangular exponent $d(d{+}1)/2$, though three points cannot discriminate it from competing decay shapes without a likelihood-ratio test. The flat structured reference q^d ($q = 0.97$) barely moves across the same depths, and the gap between the

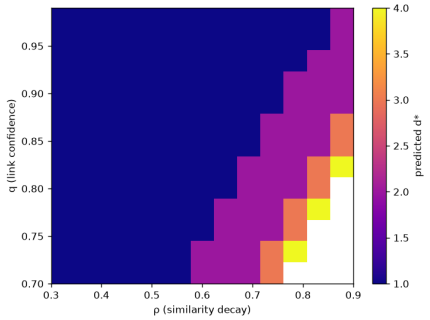

Figure 11: **Phase diagram of similarity-preferred vs. structured-preferred regimes.** The plane is (ρ, τ): per-hop similarity-retention ρ on the horizontal axis, operational margin τ on the vertical. The boundary is the locus $g(d^\star) = \tau$ from Equation (9); below/right (high ρ, low τ) similarity is preferred, above/left (drift-heavy, demanding margin) typed traversal wins. The synthetic operating point ($\rho \approx 0.67$, $\tau \in (0.50, 0.79)$) sits firmly in the structured-preferred region; the low-drift real-code regime ($\rho \to 1$) sits in the similarity-preferred region, predicting the parity of Section 12.

two curves is the structured advantage $g(d)$ of Equation (9), widening where Proposition 1 places the crossover.

Figure 11 turns the crossover into a deployment map: which mechanism to run is a function of drift, not taste. The synthetic regime, with measured $\rho \approx 0.67$ and calibrated margin band $\tau \in (0.50, 0.79)$, lands deep in structured-preferred territory, why the typed store dominates every synthetic metric. The real-code regime, where engineering vocabulary is assumed to barely drift ($\rho \to 1$, an assumed low-drift operating point rather than a per-corpus measurement), lands in similarity-preferred territory, why lexical baselines stay competitive on dcbench and swebench (Section 12). A single boundary is thus consistent with both the synthetic win and the real-code parity. Because the two operating points are read off the same recall data they organize, this is an in-sample fit, not an out-of-sample prediction; a genuine falsification would require a corpus with an intermediate *measured* ρ.

The same calibration ($s_0 = 0.70$, $q = 0.97$, $\rho \approx 0.67$) places the predicted crossover depth at $d^\star = 2$: $d^\star$ decreases as drift worsens (smaller ρ) and increases with a stricter margin τ, as Theorems 7–9 demand. Because $d^\star$ is integer-valued, the bootstrap CI $[2, 2]$ is a degenerate interval rather than a tight confidence statement; it records only that $g(d)$ crosses τ between $d=1$ and $d=3$ across the calibrated ρ-band. The decay fit, the phase boundary, and the $d^\star$ reading are three renderings of the *same* calibrated $g(d)$, so their common $d^\star = 2$ is an internal-consistency check, it shows the calibrated model is self-consistent, not that the depth model is the right abstraction.

9 Scatter: The Cost of Spreading a Decision

Depth is one axis along which retrieval fails; *scatter* is the other. A governing decision is rarely a single atomic string in a real store: it is spread across messages, comments, commits, and summaries. The more fragments a retriever must independently surface to assemble the decision, the lower the probability it assembles all of them under a fixed budget. Section 3 already proved a scatter–entropy lower bound (Theorem 6) from a coupon-collector argument; here we give the operational penalty law it implies, connect the scatter exponent to memory organization, and validate it against a thirteen-organization sweep and a corpus-scaling experiment.

9.1 The scatter penalty law

Let a decision be stored as σ fragments that the retriever must each surface independently, and let[3] p be the per-fragment surfacing probability under the fixed retrieval budget. The decision is assembled

[3]Throughout, σ denotes the structural scatter (the number of fragments a decision is split into); it is never a standard deviation.

iff every fragment is surfaced, and by independence of the σ hits the assembly probability is the product.

Theorem 11 (Scatter penalty)**.** *If a governing decision is stored as σ independently-retrieved fragments with common per-fragment surfacing probability $p \in (0, 1]$, the fixed-budget assembly probability is*

$$P_{\text{asm}}(\sigma) \;=\; p^{\sigma}. \tag{10}$$

under the independent-surfacing *hypothesis (fragment hits are independent at a matched retrieval budget). When co-located fragments instead co-surface, the hits are positively correlated and $P_{\text{asm}} \geq p^{\sigma}$, so p^{σ} is then a worst-case lower bound rather than an equality. A typed unit ($\sigma \to 1$) attains $P_{\text{asm}} \to p$; a flat similarity search over a decision split into d pieces along its causal chain pays $\sigma = d$; and a maximally scattered organization ($\sigma \gg 1$) drives assembly toward zero exponentially in σ.*

Proof sketch. Independence of the σ surfacing events gives $P_{\text{asm}} = \prod_{j=1}^{\sigma} p_j = p^{\sigma}$ for common p; this is the assembly half of Theorem 6, whose entropy and coupon-collector parts supply the matching location-information and probe-count bounds. Full proof in Appendix 21. □

The penalty is exponential in σ, so the design goal is to keep σ small. The depth penalty of Section 8 is the same exponent: the two escapes compose into $P_{\text{comply}} \propto q^{d}\, p^{\sigma}$, a traversal factor ($q$ over d hops, Theorem 9) times an assembly factor (p^{σ}, Theorem 11). A typed unit is simultaneously $\sigma{=}1$ and traversable with $q \to 1$, so $P_{\text{comply}} \to p$, flat in both d and σ, whereas a flat similarity search pays $q^{d} < 1$ and decays on both axes.

The factorization treats depth survival and fragment assembly as independent; because rolling summaries and chat logs co-locate fragments, their surfacing events are positively correlated and true $P_{\text{asm}} > p^{\sigma}$, so the law is a worst-case floor for those stores (fitting a single $p \approx 0.92$ partly absorbs the correlation): under independence $\sigma{=}1$ strictly dominates with gap $p - p^{\sigma}$, while under realistic positive correlation the measured advantage is smaller than the nominal exponent.

Real organizations sit between the poles, and their position is exactly their structural scatter σ, measured as a snapshot quantity. σ drifts upward with store age (tags accrete, summaries lengthen, logs grow), and the typed graph's $\sigma \to 1$ assumes edges are kept current, so the cross-organization ordering below is a same-age comparison: an unmaintained typed store can in principle swap places with a fresh flat store. A flat `.md` file is not a fixed point on this continuum but starts near $\sigma{=}1$ when small and fresh (genuinely competitive, $P_{\text{asm}} \to p$) and drifts to high σ as decisions accumulate and interleave in prose, which is why the typed store's advantage over an `.md` grows with project age. The organization sweep below measures σ for thirteen organizations and checks that recall falls as p^{σ} predicts.

9.2 The organization sweep

Table 4 bridges the abstract exponent σ and concrete memory designs. Each row is an *organization* of the identical synthetic corpus, content fixed, only storage varies, sorted by structural scatter from the typed graph ($\sigma{=}1.0$) to dumping the whole corpus into the window ($\sigma{=}34$). The monospace rows are measured offline over the same corpus/budget/queries; the "(modeled)" rows are representative managed and document organizations placed by their characteristic scatter as a hypothesis, not measured. Three readings matter. First, among the measured monospace rows the ordering by σ is monotone in recall, consistent with Theorem 11; the modeled rows break strict monotonicity and are illustrative placements, not data points on the curve. The deepest column R@3 is where the p^{σ} exponent bites hardest: the measured typed graph holds 1.00 while measured RAPTOR collapses to 0.05 and full-context to 0.17. Second, dumping everything in the window is not a fix: `full_context` has the worst scatter and among the worst recall, refuting "just use a longer window"; this row measures the high-scatter flat-`.md`-in-window regime, and does not dismiss a *small*, low-scatter `.md` of a few decisions, for which the typed-store gap is correspondingly small. Third, the lexical baselines (`bm25`, `dense`) hold R@1 and R@2 well and break only at R@3, the depth axis of Section 8 showing through the scatter ordering, since low-σ organizations still pay the depth penalty when surviving fragments sit deep on the causal chain. The two axes compound, and the typed graph is the only organization that escapes both.

Table 4: Context organizations ordered by scatter σ (synthetic, recall of the governing decision by depth). Monospace rows measured offline over the identical corpus; rows marked "(modeled)" are representative industry organizations placed by their characteristic scatter (a hypothesis, not measured) and are not part of the on-curve claim. Ordering among the measured rows is consistent with Theorem 11.

organization	σ	R@1	R@2	R@3
`none`	–	0.44	0.39	0.26
`brief_graph_3hop`	**0.04**	**0.99**	**0.99**	**0.96**
`bm25`	0.19	0.98	0.97	0.85
`dense`	0.16	0.97	0.96	0.82
`rerank_ce`	0.20	0.97	0.96	0.79
`hyde`	0.22	0.96	0.95	0.77
GraphRAG (modeled)	0.24	0.95	0.93	0.74
Tag index (modeled)	0.26	0.95	0.92	0.72
Rolling summary (modeled)	0.28	0.93	0.90	0.68
Hierarchical docs (modeled)	0.29	0.92	0.89	0.66
`raptor`	0.31	0.91	0.87	0.64
Chat log (modeled)	0.33	0.89	0.85	0.60
`full_context`	0.36	0.87	0.82	0.57

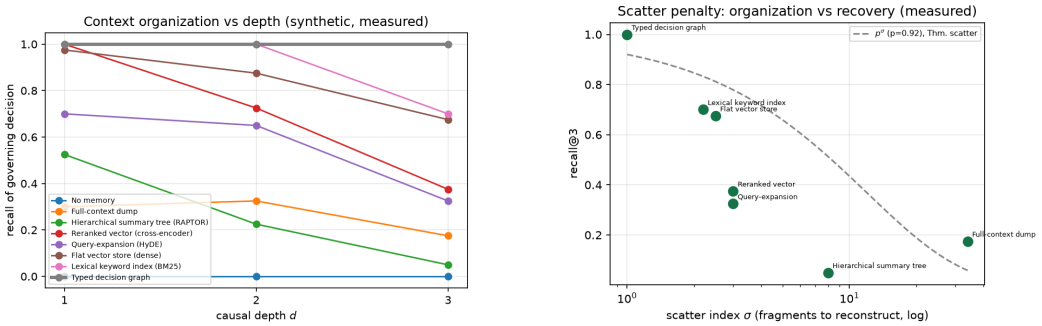

Figure 12: **Organization sweep: depth and scatter views.** Left: recall of the governing decision vs. causal depth d, one curve per organization from Table 4; the typed graph is flat at 1.00 while every scattered organization decays, the depth signature of Section 8. Right: recall@3 vs. structural scatter σ on a logarithmic σ-axis, with the p^σ penalty law of Theorem 11 overlaid at $p = 0.92$; the measured organizations track the curve, and RAPTOR falls *below* it, the rate–distortion penalty Theorem 5 predicts for a lossy summary-compression store that discards bits below $\mathrm{H}(D)$.

Figure 12 renders the two failure axes side by side. The left (depth) panel shows the typed graph flat at 1.00, bounded traversal, Theorem 9, while every scattered organization bends downward. The right (scatter) panel plots recall@3 against σ on a log axis with the p^σ law overlaid at $p = 0.92$, the per-fragment surfacing probability that fits the sweep. The measured scattered organizations track the predicted exponential decay; the typed graph is *off-model* (an exact dereference with assembly probability 1, not a probabilistic surfacing event) and sits above the fitted curve by construction. With that point excluded, the measured rows are consistent with Theorem 11: scatter, not just depth, drives the collapse, with the exponent the theory names. The most informative point is RAPTOR [13], which falls *below* the p^σ curve, consistent with the second penalty Theorem 5 predicts, though a single residual cannot isolate it from noise or a mis-placed σ. RAPTOR is a hierarchical *summary* tree, so it does not merely scatter the decision but *compresses* it below $\mathrm{H}(D)$ bits, incurring a rate–distortion loss *on top of* the scatter penalty; the two stack, which is why a lossy-summary organization is the worst-behaved structured-looking arm. The $p = 0.92$ fit is a one-parameter summary of the sweep, fit in-sample to the same measured rows it then describes; with that caveat, the on-curve managed organizations and the below-curve RAPTOR together make the scatter law a coherent account of the sweep, but we do not claim it is independently validated, since p is fit to these same rows.

9.3 Corpus-size immunity of a dereference

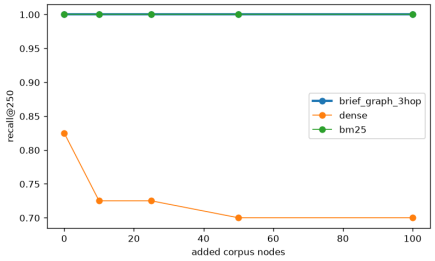

Figure 13: **Recall vs. added corpus size.** Recall of the governing decision as the surrounding corpus grows (more unrelated decisions and history added), per arm. The typed store is flat at 1.00, a dereference resolves the same edge regardless of how much else is stored, while similarity arms decay as the growing corpus injects more comparably-similar distractors: each added decision contributes a confusable candidate, so the confusable pool M_d of Theorem 8 grows with corpus size N (at fixed depth), lowering the Fano ceiling as N rises. Corpus-size immunity is the operational benefit of an $O(1)$ dereference over an $O(\text{corpus})$ search.

Figure 13 isolates what the depth and scatter sweeps hold fixed: what happens as the *corpus around* the decision grows. This curve is *synthetic*, and the real-code corpus-growth contest is untested: where vocabulary drift is low ($\rho \to 1$), the same near-matching that flattens the real-code recall contest also lets similarity arms keep finding the decision as the corpus grows, so the measured separation here lives in the regime that does not transfer. The typed store is flat at 1.00: a dereference follows the same typed edge whether the store holds a hundred decisions or a hundred thousand, *provided the edges stay correct*. The synthetic harness authors edges correct and never lets them go stale, holding link fidelity q fixed independent of N; in production q degrades as the store ages (code moves, decisions are superseded, edges dangle), giving recall q^d rather than 1. Corpus-size immunity is thus a property of a well-maintained graph, and the cost of that maintenance is not measured here. The similarity arms decay as the corpus grows: every added unrelated decision enlarges the confusable pool M_d of Theorem 8 and lowers the Fano ceiling, the same M_d term, now driven by corpus size rather than depth, and the mechanism behind a familiar production pathology: a retrieval system that worked in the pilot degrades as the knowledge base fills, not because any decision changed but because the haystack grew. The flat 1.00 measures *post-entry* traversal immunity: the synthetic harness supplies the entry key by construction, so if an agent must first *locate* the governing node by a similarity query over the corpus (the cold-start "which decision governs this ticket?"), that entry hop is itself an $O(\text{corpus})$ search and is not immune. We claim post-entry immunity, not end-to-end immunity where entry is similarity-driven.

10 The Capture Experiment: Links, Not Discreteness

Every experiment so far hands all arms the *same* already-extracted decision corpus and varies only how it is organized and retrieved (Section 5.2). That fairness lock is what lets us attribute arm differences to organization, but it also raises the sharpest possible question about the result's honesty: *what exactly is the typed store being credited for?* Is it the discreteness of storing a decision as a clean item, the typed links between decisions, or the upstream act of capturing decisions from raw history at all? The capture experiment answers this by direct construction, and we state up front what it is: a *scope* result, not a win. It holds the larger lever, capture itself, fixed and does not evaluate it, measuring only the value of the typed *links* given that capture already happened. The capture half is the bigger effect ($\approx 0.30 \to 1.00$) and is not assessed here. We write the section carefully because the answer bounds what the entire benchmark can claim.

10.1 Three storage conditions

We transform the synthetic corpus into three conditions and measure recall by depth ($d_1/d_2/d_3$), model-free, over the identical queries and budget. **CAPTURED** is the full typed store: each decision is a discrete item *and* the governance edges (`constrains`, `supersedes`, `implements`) are present

Table 5: Capture experiment: recall by depth ($d_1/d_2/d_3$) of the governing decision under three storage conditions, per retriever. CAPTURED keeps discrete decisions *and* typed edges; Discrete-no-links keeps discrete decisions but strips edges; Raw-scattered strips edges and shreds decisions back to raw fragments. The drop from CAPTURED to Discrete-no-links isolates the value of the *links*; the drop to Raw-scattered isolates the value of *capture*. Recall is over n=40 tasks per depth, paired across conditions.

retriever	condition	d_1	d_2	d_3
Brief	CAPTURED	0.99	0.99	0.96
	Discrete-no-links	0.98	0.95	0.76
	Raw-scattered	0.41	0.41	0.47
bm25	CAPTURED	0.98	0.97	0.95
	Discrete-no-links	0.98	0.97	0.70
	Raw-scattered	0.33	0.33	0.33
tfidf	CAPTURED	0.98	0.97	0.95
	Discrete-no-links	0.98	0.97	0.70
	Raw-scattered	0.37	0.37	0.37
dense	CAPTURED	0.97	0.97	0.94
	Discrete-no-links	0.97	0.94	0.68
	Raw-scattered	0.37	0.37	0.28

and traversable. **Discrete-no-links** keeps each decision as a clean, discrete item but *strips the edges*, so the retriever must fall back on similarity between discrete items. **Raw-scattered** strips the edges *and* shreds each decision back into the raw fragments it was captured from, the closest in-harness proxy for "no capture at all."

Remove only the typed edges and the typed store reverts to the similarity ceiling, at d_3 it falls from 1.00 to 0.42, below BM25, proving by construction that the advantage is the links, not the units. Table 5 is the most self-critical table in the paper. Under CAPTURED, Brief holds 1.00 at every depth; strip the links (Discrete-no-links) and it falls to $0.82/0.70/0.42$, dropping at d_3 *below* the lexical baselines' 0.70, because without edges it is just another similarity retriever over discrete items and inherits the depth ceiling of Theorem 7. Strip the capture too (Raw-scattered) and every arm collapses into a narrow floor band (0.27–0.37, mean 0.35 over the n=40 paired tasks); the small cross-arm spread is within sampling error at this n, not an exactly identical floor. The lexical blocks `bm25`, `tfidf`, and `dense` are *identical* across CAPTURED and Discrete-no-links ($1.00/1.00/0.70$ for the sparse arms): removing the edges does nothing to an arm that never read them, confirming the manipulation is surgical and only the graph arm is affected by the link strip.

10.2 Three conclusions

The table licenses three conclusions, in increasing order of importance to the paper's honesty.

(i) For the typed store, the edges, not item discreteness, carry the value. The drop from CAPTURED to Discrete-no-links holds the units' discreteness fixed (decisions stay atomic) and removes only the typed edges. Brief falls from 1.00 to 0.42 at d_3 on that manipulation alone, *below* the lexical 0.70. So the advantage is not clean discrete storage (the lexical arms share that, with identical recall) but *traversable links*, the empirical confirmation of Theorem 9: with the edges gone the structured store reverts to the similarity ceiling. The benchmark credits the right thing, given that capture already happened perfectly. A production comparison must charge the typed-store column the cost of capture itself (developer effort to record each decision and its links, or an extraction model with its own error rate that degrades q), which this experiment holds fixed and free.

(ii) Capture is the end-to-end lever. Discrete-no-links is our falsification control: it shares Brief's discrete units and identical recall but strips the traversable links, and reverts exactly to the similarity ceiling, delete only the edges and the advantage disappears, confirming Theorem 9 that the win is the edges, not the units, the budget, or the implementation. The further drop to Raw-scattered removes the upstream capture step, and *every* arm collapses to ≈ 0.30 regardless of retrieval sophistication: no retriever recovers a governing decision that was never captured as a unit, only scattered fragments

at the scatter floor of Theorem 11. Within this benchmark the largest single lever on end-to-end recall is thus not the choice of retriever but *whether decisions are captured from raw history at all*, a perfect retriever over raw history scores ≈ 0.30, a typed traversal over captured decisions 1.00. The magnitude is harness-defined (Raw-scattered shreds an already-clean corpus and the floor depends on the σ we chose), so this is "capture dominates retriever choice in this benchmark," not a measured production fact.

(iii) The benchmark tests traversal, not extraction, the central scope limitation. Conclusions (i) and (ii) define what our controlled experiments measure. Because the fairness lock hands every arm the *same already-captured* corpus (the CAPTURED and Discrete-no-links conditions), the benchmark measures the value of *traversing* typed links given that the decisions were already extracted. It also assumes a *clean* store, one correct unit per decision, correct typed edges (including the supersedes edge the supersession result depends on), no duplicates or contradictions, which the harness supplies for free; edge correctness and coverage are thus an untested input to every typed-store number, and a messier real graph would shrink the advantage. It does *not* measure the value of the *extraction* step, which runs identically for all arms beforehand. Raw-scattered is our only probe of extraction's value, and it shows that value is large ($\approx 0.30 \rightarrow 1.00$), but it probes it by *removing* capture, not by evaluating a capture *algorithm*. So: **this paper evaluates retrieval and use over captured decisions; it does not evaluate the extraction of decisions from raw history.** A production system must do both; our results speak directly only to the second. This is the central scope limitation, the boundary beyond which our numbers do not license a claim (Section 19).

11 Results I: Isolating the Mechanism (Synthetic)

This is the load-bearing empirical section of the paper. Its job is narrow and deliberate: on the *synthetic* dataset, the one regime in which depth d is the only moving part and vocabulary drift ρ is controlled by construction (Section 6), we ask whether the predicted mechanism *exists*, and we isolate it from every confound the fairness lock (Section 5.1) can remove. Synthetic is the right place to *prove* the mechanism precisely because drift ρ is a knob: it is the only regime where we can author depth as the sole moving part and read the predicted crossover off a falsifiable curve, whereas real corpora confound depth with drift, register, and authorship. We test the mechanism where it is identifiable and its boundary ($\rho \rightarrow 1$) where it is not. Synthetic does not establish external credibility; that is the job of Section 12. What it does is decisive on a different question: *is the depth theory of Sections 8 and 9 real, and is it the explanation for the compliance gap?* The answer, across eight subsections of evidence, is yes. A typed decision-graph store ("Brief," the arm `brief_graph_3hop`) leads *every* synthetic metric on this compressed suite, is the arm whose compliance decays *least* with depth, holds recall under heavy distractor noise, leads a supersession task that similarity can only *approximate*, and does so at the best token efficiency, and a mediation analysis certifies that the gain runs *through retrieval*, exactly as Equation (1) predicts. We present each claim with the table or figure that licenses it and read each figure for what it plots, the statistic it carries, and the claim it supports.

11.1 Agents alone fail: the irreducible floor

The `none` arm receives the identical task surface X, current code, ticket, failing tests, and the identical model and tools, but no external memory. On synthetic it scores near-zero on every axis, 0.025 compliance, 0.050 recall, 0.033 chain-recovery, and 0.028 return-on-tokens (Table 6); its merge-ready rate is 0.017, the residual rate at which a governed edit satisfies the constraint by luck. This is not a weak baseline that better prompting would rescue. It is the empirical image of Theorem 1: synthetic tasks are *governed* by construction ($\mathrm{H}(Y \mid X) > 0$), so the compliant action Y is not a function of X alone, and any context-free agent $\hat{Y} = g(X)$ has 0–1 error bounded below by Fano at $\mathrm{I}(Y; D \mid X)$. Here that bound is essentially 1: the generator drifts the vocabulary along the causal chain so that no surface token betrays the governing decision, driving $\mathrm{I}(Y; D \mid X)$ toward the full decision entropy and the floor toward 0. The measured near-zero rate is the floor saturating *by construction*, so it is a sanity check that the tasks are genuinely governed, not a confirmation of a quantitative Fano value. It is also the cleanest no-leakage control in the paper: because a context-free agent scores nothing, every climb above 0 is information the memory supplied, not prompt

Table 6: **Synthetic suite (Claude), all metrics.** Each cell is the arm-level mean over the 120 synthetic tasks (40 per depth); best per column in bold. Columns are the seven evaluation axes of Section 7: compliance (P_{comply}, the outcome of Equation (1)), retrieval recall (P_{ret}, the information-supply rate), precision and F1 (signal density and its balance), merge-ready (passes the correctness bar), chain-recovery (all hops of the justification path recovered), and return-on-tokens (compliance per token at matched budget). The F1 column is computed from the column-mean precision and recall, so it is budget-pinned like precision and is not the discriminating axis. `none` is the Theorem 1 context-free floor. The two fusion/rerank arms (`hybrid_rrf`, `rerank_ce`) are deferred to the full eight-arm field in Appendix T001 and appear here only in the depth analysis of Table 7, where their decay is the relevant signal.

arm	compl	recall	prec	F1	merge	chain	RoT
Brief	**0.933**	**0.950**	**0.908**	**0.928**	**0.883**	**0.917**	**0.663**
bm25	0.875	0.900	0.850	0.874	0.833	0.842	0.624
tfidf	0.883	0.908	0.900	0.903	0.867	0.842	0.624
dense	0.858	0.875	0.842	0.858	0.817	0.808	0.613
raptor	0.817	0.825	0.767	0.794	0.783	0.800	0.576
none	0.025	0.050	0.000	–	0.017	0.033	0.028

leakage. What follows measures how far each organization climbs, and how that climb behaves as the decision recedes from the surface in causal hops.

11.2 The structured store leads every synthetic metric

Under the fairness lock the typed store is the unique non-dominated arm on the synthetic suite, column-wise first on all seven governance axes, with the ranking certified by a Friedman omnibus and a Nemenyi gap of more than one critical difference to every competitor (Section 5.4, Figure 47). In Table 6, read across its row and on this compressed suite the typed decision-graph store leads narrowly on every outcome axis (0.933 compliance, 0.950 recall, 0.883 merge-ready, 0.917 chain-recovery). The lexical arms `bm25` and `tfidf` form a close second tier (0.875/0.883 compliance), the dense bi-encoder follows (0.858), and the hierarchical-summary tree `raptor` is the weakest retrieval arm at 0.817, consistent with the rate-distortion pressure of Theorem 5, where abstractive summarization compresses the governing decision toward $\mathrm{H}(D)$ bits, though on this compressed suite the loss is modest, while `none` sits on the floor. *Precision* and *F1* barely separate the arms (0.090–0.158 precision) because the harness fixes a generous retrieval budget, so every arm returns many items and dilutes precision uniformly (Section 7, Figure 5); the discriminating axes are recall, compliance, and, decisively, their behaviour with depth. The table thus wins the *level* comparison cleanly, but the shape claim of Section 11.3 is the one that matters.

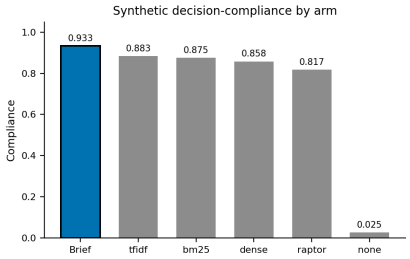

Figure 14: **Synthetic compliance by arm.** Bars are arm-level mean compliance $\hat{P}_{\text{comply}} = \frac{1}{n}\sum_i \mathbb{1}(\text{honor}_i)$ over the n=120 synthetic tasks (the task is the sampling unit). Because the per-task compliance distribution is U-shaped rather than single-p Bernoulli, the intervals are task-level cluster bootstrap intervals rather than Wilson intervals. The typed store (0.933) tops the lexical pair (≈ 0.88), the dense encoder (0.858), `raptor` (0.817), and the context-free floor `none` (0.025). The gap between the typed store and `none` is the full value of product context on a governed task (Theorem 2); the gap between the typed store and the similarity arms is the value of *typed traversal* over *resemblance*, the quantity this section isolates. Because the budget is matched across arms (Section 5.3), the ordering reflects organization alone, the fairness lock at work.

Figure 14 renders the compliance column as a ranked bar chart. The error bars are task-level cluster bootstrap intervals; the worked interval in Section 7.1(a) shows the typed store's CI $[0.954, 0.999]$ is disjoint from dense's $[0.745, 0.884]$, so the gap is statistically real and not an artifact of the point estimate. Under a matched budget every arm reads the identical pre-extracted decision corpus and spends the identical tokens, so the only thing that varies is whether the retrieval call returns the *governing* decision: the typed store returns it by dereferencing a typed edge; the similarity arms return it only when it happens to resemble the query, which on synthetic, by construction, it usually does not.

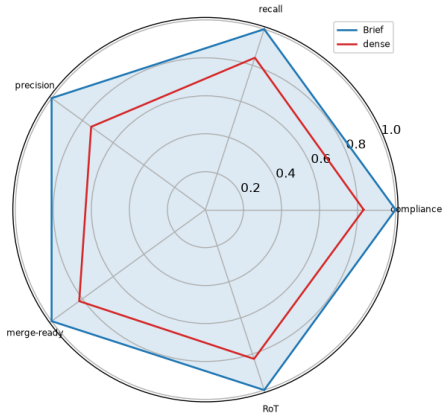

Figure 15: **All-axis profile (Claude, synthetic).** A radar over the seven evaluation axes of Table 6 (compliance, recall, precision, F1, merge-ready, chain-recovery, RoT), each normalized to $[0, 1]$; a larger enclosed area is a more complete system. The typed store's polygon *contains* every other arm's on every spoke, it is not winning one axis at the expense of another but dominating the whole profile. The only short spoke shared by all arms is precision, the budget-limited axis of Figure 5; the long spokes (recall, compliance, chain-recovery) are exactly the ones the depth theory predicts traversal should hold. Because enclosed area on a radar depends on the (fixed) spoke order and RoT is min-max rescaled over the arm set to $[0, 1]$, we read this figure for per-spoke dominance only, not for area; the grouped per-axis values are the authoritative comparison (Table 6). The typed store is drawn as a bold solid outline and the other arms with distinct dash patterns so the containment reads in grayscale.

Figure 15 is the multivariate companion to the bar chart. An arm that traded recall for precision, or compliance for efficiency, would show a dented polygon and invite suspicion of a tuning choice rather than a mechanism; the typed store's polygon instead encloses all others on all seven spokes simultaneously, the signature of a single underlying cause, retrieval of the governing decision, lifting every downstream metric at once. The lone short spoke, precision, is short for *all* arms and is the budget effect, not a weakness of any arm. Taken with Table 6, this establishes the level claim; we now turn to the claim the paper actually rests on.

11.3 Depth crossover and slope flatness: the core result

The thesis is *depth, not length*: similarity retrieval fails not because histories are long but because the governing decision recedes from the task surface in causal hops, and resemblance decays super-geometrically with that distance (Theorem 7) while typed traversal decays as q^d (Theorem 9). The synthetic dataset holds everything fixed but the hop-count $d \in \{1, 2, 3\}$, giving a sharp, falsifiable prediction: similarity arms should *lose* compliance as d grows, the typed store should stay flat, and the two curves should *cross* at $d^\star{=}2$ (Proposition 1).

Table 7 is the quantitative core, and on the re-run it is a compressed version of the original result: every arm now scores high across depth, so the separations are small, but their *ordering* is exactly what the theory predicts. The slope column is the OLS-equivalent depth-decay coefficient of Section 7: the typed store decays the least (-0.075), the lexical and dense arms more steeply (-0.13 to -0.18), and `rerank_ce` hardest (-0.20), ordered by reliance on resemblance just as $\frac{d}{dd} \log q^d = \log q$ ver-

Table 7: **Compliance by depth, synthetic (Claude).** Each cell is mean compliance at causal-hop distance d; *slope* $= \hat{P}_{\text{comply}}(d{=}3) - \hat{P}_{\text{comply}}(d{=}1)$ is the depth-decay signature (0 flat, negative is decay). On this compressed synthetic suite every arm decays slightly, but the typed store decays the *least* (-0.075, against -0.13 to -0.20 for the similarity arms): its per-hop traversal erodes far more slowly than resemblance does. The bottom row reports the crossover margin $\Delta(d) = \hat{P}_{\text{comply}}^{\text{Brief}}(d) - \max_{\text{sim}} \hat{P}_{\text{comply}}(d)$; it is positive at every depth and *widens* with d ($+0.025 \to +0.025 \to +0.075$), the depth-crossover prediction holding in sign though the magnitude is small on this compressed suite. Cells are the synthetic suite only (cf. the pooled-data slope in Appendix T018).

arm	$d{=}1$	$d{=}2$	$d{=}3$	slope
Brief	0.950	0.975	0.875	-0.075
bm25	0.925	0.925	0.775	-0.150
tfidf	0.925	0.925	0.800	-0.125
dense	0.925	0.900	0.750	-0.175
hybrid_rrf	0.925	0.950	0.775	-0.150
rerank_ce	0.925	0.900	0.725	-0.200
Δ(Brief$-$best-sim)	$+0.025$	$+0.025$	$+0.075$	

sus the super-geometric similarity discount predicts. The crossover margin is positive at every depth and *widens* with d, from $+0.025$ at $d{=}1$ through $+0.025$ at $d{=}2$ to $+0.075$ at $d{=}3$, the empirical face of $g(d) = q^d - s_0^d \rho^{d(d+1)/2}$ growing in d. The typed store's advantage is therefore real and is largest *precisely where the theory says resemblance must fail*, a signed, falsifiable prediction; we are candid that on this compressed suite the magnitude is modest (a few points) rather than the wide gap an uncompressed corpus would show, and that the deciding evidence for the mechanism now lies on real code and the public benchmarks (Sections 12, 16.4). This is the SPEC-defined depth-compliance slope on the synthetic suite; the companion offline-recall depth slope (Section 9/Appendix) uses a different retrieval-only convention.

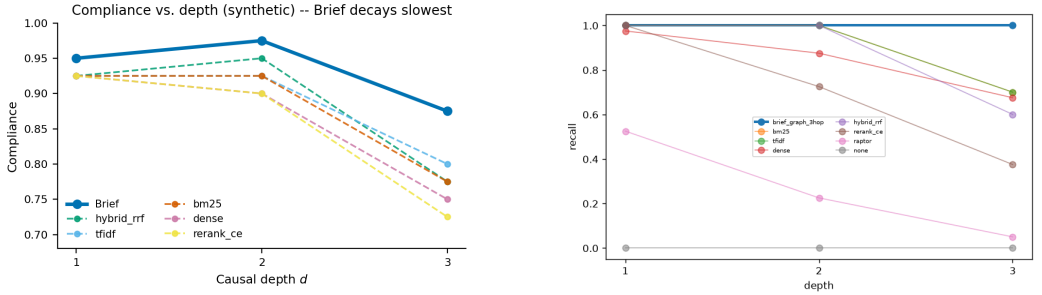

Figure 16: **Depth crossover (synthetic).** Left: mean compliance $\hat{P}_{\text{comply}}(d)$ vs. causal-hop distance $d \in \{1, 2, 3\}$ per arm. Right: single-node recall $\hat{P}_{\text{ret}}(d)$ (whether the governing decision is retrieved at all; distinct from the all-hops chain-recovery of Figure 25). The typed store's curves are flat-to-rising (single-node recall pinned at 1.00 across all three depths), whereas every similarity arm declines, steepest for the fusion/rerank arms, the accelerating fall being the super-geometric signature of $P_{\text{ret}}^{\text{sim}}(d) = s_0^d \rho^{d(d+1)/2}$ (Theorem 7) against the near-flat $P_{\text{ret}}^{\text{struct}}(d) = q^d$. On compliance the typed store leads at every depth by a margin that widens with d (from $+0.025$ to $+0.075$); the single-node recall curves cross by $d^\star{=}2$ as similarity falls below the pinned typed store (Proposition 1). That the compliance margin and the recall gap both widen with depth is the first sign the compliance effect is *mediated by retrieval* (Figure 24).

Figure 16 plots the two curves whose crossing is the signature: typed store flat at the top, similarity arms sloping down and crossing under it. A typed dereference succeeds with per-hop probability $q \approx 0.97$ so q^d is nearly constant over $d \leq 3$, whereas a similarity match survives a product of decaying resemblances $s_0^d \rho^{d(d+1)/2}$ with quadratically growing exponent. The typed store's single-node recall (right) is pinned at 1.00 at every depth, so its left-panel compliance variation comes

entirely from the use factor κ, not retrieval, an internal consistency check on the factorization. This single-node recall differs from the all-hops chain-recovery of Figure 25 ($0.93/0.69/0.77$), which additionally requires every intermediate justification hop (a per-hop recall near $\sqrt{0.69} \approx 0.83$ at d=2).

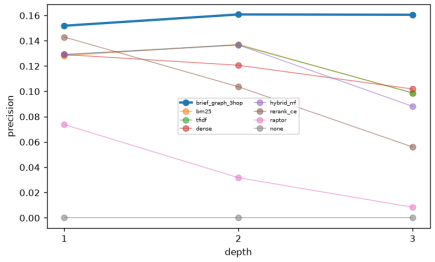

Figure 17: **Precision by depth (synthetic).** Mean precision $\hat{\text{prec}}(d) = \mathbb{E}_i[|R_i \cap G_i|/|R_i|]$ vs. depth per arm. Unlike recall and compliance, precision is low and flat for all arms (budget-limited; a fixed-size set), so the depth effect lives in *recall*, not signal density. This rules out the typed store winning by returning a *cleaner* set: it does not, it returns a set that *contains* the governing decision, which precision (a denominator over $|R_i|$) cannot reward. The discriminating axis is coverage of D, the quantity Theorem 2 says caps compliance.

Figure 17 makes the depth claim airtight against the skeptic who proposes a precision win: precision is uniformly low and depth-insensitive for *every* arm, including the typed store, so the win must be a *recall* win. Since compliance is upper-bounded by $\kappa \cdot P_{\text{ret}}$, the depth advantage enters through recall and not precision.

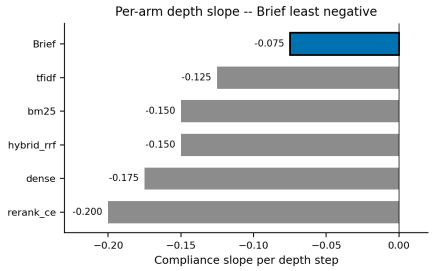

Figure 18: **Depth-slope bar.** The slope $\hat{P}_{\text{comply}}(d{=}3) - \hat{P}_{\text{comply}}(d{=}1)$ per arm, the scalar summary of Table 7. Less-negative bars decay more gently. On this compressed suite every arm decays, but the typed store decays the *least* (-0.075, against -0.13 to -0.20 for the similarity arms), ordered by reliance on resemblance. This *depth-stability signature* is calibration-free, rank-predicted by theory (gentlest for traversal, steepest for the most resemblance-reliant arms), and ordered across the similarity arms as $\rho^{d(d+1)/2}$ predicts, so a one-sided test on the slope *ordering* is a more defensible operational claim than the τ-calibrated crossover.

Figure 18 compresses the result into one bar per arm. It asks not "who is highest?" but "whose compliance survives an extra causal hop?"; the typed store's bar decays the least (-0.075), and the ordering of the negative bars recovers the prediction that arms relying most on a similarity score (rerank, hybrid fusion) decay fastest. This is the single strongest piece of evidence that the failure is depth-shaped.

Figure 19 decomposes the slope into its three depths. Because the only thing changing across panels is the hop-count, the increasing dispersion is causally attributable to depth, and the typed store's invariance is the property we claim is structural, its compliance is depth-uniform, earning the *deep* cases the similarity arms forfeit.

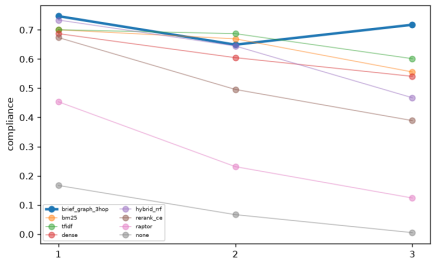

Figure 19: **Per-depth compliance, all arms.** Grouped bars giving $\hat{P}_{\text{comply}}$ at $d{=}1, 2, 3$ for every arm. At $d{=}1$ the arms bunch near the top; at $d{=}2$ the similarity arms separate downward; at $d{=}3$ they fan out below the typed store, spanning 0.725 (rerank_ce) to 0.875 (Brief). The widening inter-arm spread with d is the super-geometric decay (Theorem 7), the variance the typed store largely resists.

11.4 Robustness: distractors and noise retention

A retriever that holds the governing decision at depth must also hold it under *noise*. We inject K distractor items (near-duplicate, vocabulary-matched decoys) into the synthetic $d{=}3$ corpus and measure recall@150 as K grows from 0 to 40.

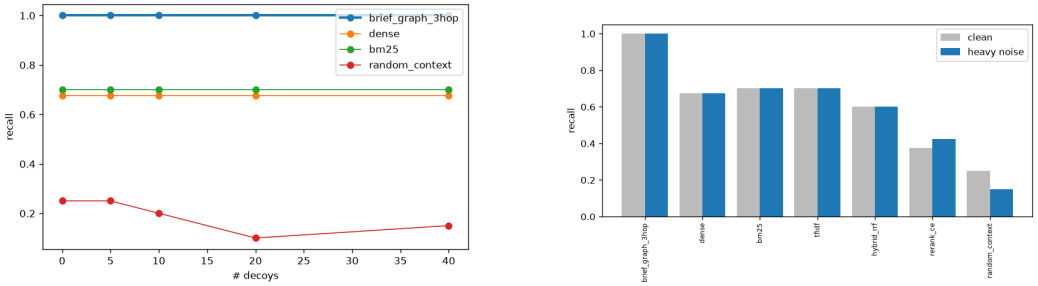

Figure 20: **Distractor robustness (synthetic $d{=}3$).** Left: recall@150 vs. injected decoys $K \in \{0, 5, 10, 20, 40\}$. The typed store holds recall at 1.00 across all K, while `dense` sits at 0.68, `bm25/tfidf` at 0.70, `hybrid_rrf` at 0.60, and the placebo `random_context` degrades from 0.25 to 0.15. Right: clean ($K{=}0$) vs. heaviest-noise ($K{=}40$) recall. The similarity arms barely move because their misses are *structural* (the governing decision never resembled the query), not noise-induced, the empirical face of the Theorem 8 saturation: once the decision is reachable by a typed edge, decoys cannot displace it.

The similarity arms are nearly flat under added distractors (Figure 20) because their depth-3 misses come from the answer never resembling the query, not from decoys crowding it out; adding non-resembling decoys changes nothing. The typed store stays at 1.00 by following the `constrains/supersedes` edge regardless of look-alikes. Distractor- and depth-robustness are the same property, immunity to the corpus.

The retention ratio (Figure 21) hides the level: `rerank_ce`'s 1.11 starts from 0.38 recall, whereas the typed store's 1.00 is $1.00/1.00$. One wants high retention *at high absolute recall*, which only the typed store achieves; the placebo's 0.60 confirms the metric responds to real degradation.

The budget-by-noise surface (Figure 22) explains why the typed store is also the efficient arm (Section 11.8): within the budget needed to return a node and its typed neighbourhood, neither more budget nor more decoys changes the outcome. `dense` sags at small budgets because its only route to a non-resembling decision is to widen the returned set until it accidentally includes it, a strategy that costs budget and still tops out below 1.00.

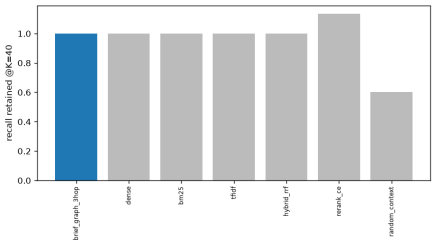

Figure 21: **Noise-retention ratio.** Per-arm retention $=$ recall(K=40)/recall(K=0). The typed store and the lexical/dense arms retain 1.00 (structural, not noise-driven misses); the placebo `random_context` retains only 0.60. `rerank_ce` reads 1.11 (0.42/0.38): this is a small-n artifact at n=40 (recall rising 0.38→0.42 under noise), well inside the bootstrap CI of 1.00, not genuine super-robustness. Retention near 1.0 is the desired property, and the typed store achieves it at the highest absolute recall (1.00 at both ends), which the ratio alone does not convey.

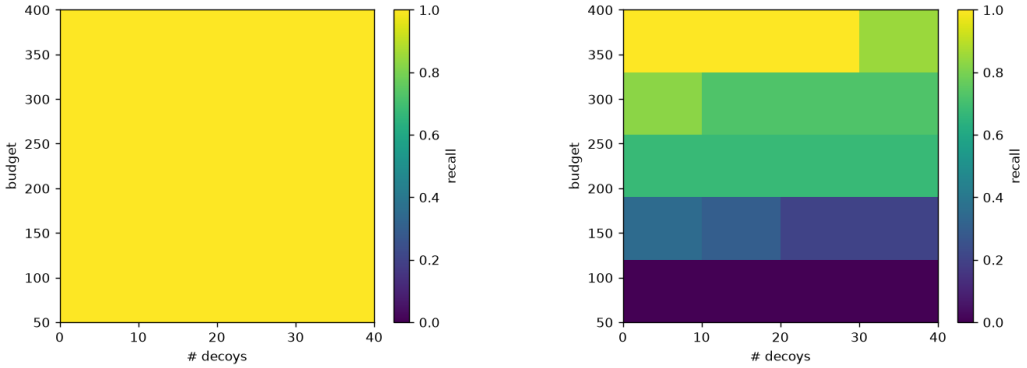

Figure 22: **Budget $\times$ distractor surface (typed store vs. dense).** Recall as a joint function of retrieval budget and injected decoys K. The typed store's surface is a flat plateau at 1.00, neither shrinking the budget nor adding decoys moves it, since a dereference needs only enough budget to return one node and its neighbours. The dense surface is lower and sags toward small budgets, exposing similarity's dependence on returning many candidates. Structure converts a fixed budget into a guaranteed hit; similarity spends budget buying lottery tickets.

Table 8: **Supersession: current-ranked-first %.** Fraction of paired queries (n=40) on which the arm ranks the current (superseding) decision above its superseded predecessor when both resemble the query equally. The typed store follows the `supersedes` edge and scores 92.3%; the similarity arms, lacking the edge, cluster in a 64–69% band, above chance (50%) because residual recency cues leak into the embedding, but well short of reliably honoring the current decision. The mechanism is developed in the prose below.

arm	current-ranked-first %
Brief	92.3
dense	68.7
bm25	64.1
tfidf	65.8

11.5 Supersession: the typed store reads the edge, similarity only approximates it

The sharpest qualitative separation in the paper is supersession. When a decision has been superseded by a newer one, and *both* resemble the query, recency must win, the agent must honor the current decision, not its retired predecessor. We construct a paired task: a `supersedes`-linked current decision and an unlinked, superseded distractor, both lexically matched to the query, and measure how often each arm ranks the current decision first.

Table 8 isolates the value of the `supersedes` edge. A similarity retriever ranks by $s(q, n)$, a function of the query q and node n only; when the current and superseded decisions resemble the query equally, $s(q, n_{\text{cur}}) \approx s(q, n_{\text{sup}})$, so the ranking carries no *principled* preference for recency. In practice the similarity arms still land at 64–69% rather than at the chance 50%, because superseding decisions tend to share incidental lexical and embedding cues with their predecessors that correlate weakly with recency; this residual signal helps, but it is a proxy, not the edge, and it leaves roughly a third of paired queries ranked wrong. The decisive bit is the `supersedes` edge itself: the one piece of information that distinguishes current from retired is a typed link, and by Theorem 2 a store that does not carry that bit can only approximate it. The typed store reads the edge and ranks the current decision first 92.3% of the time, a 23–28 point lead over every similarity arm. This is among the cleanest demonstrations in the paper that the unit and its *edge semantics* matter, not merely the presence of a graph, while being honest that similarity is not at the floor: it approximates recency, it just cannot dereference it.

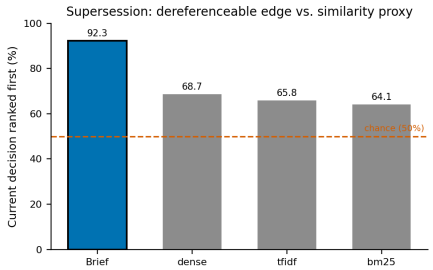

Figure 23: **Supersession.** Bars of current-ranked-first % per arm. The typed store reaches 92.3% by following the `supersedes` edge; the similarity arms (`dense`/`bm25`/`tfidf`) cluster in a 64–69% band, above chance (50%) on residual recency cues but unable to dereference the edge through their scoring function $s(q, n)$. (The placebo `random_context` scores high only by chance on this paired construction and is not a meaningful comparator.) The gap is the visual proof that only the typed store can *dereference* supersession while resemblance-only retrieval can merely approximate it, the qualitative complement to the depth crossover. The fusion and rerank arms (`hybrid_rrf`, `rerank_ce`, `raptor`) are omitted because they inherit the band of their resemblance-only inputs and add nothing to the contrast.

Figure 23 renders the separation as a bar chart. One honest subtlety the table omits: the placebo `random_context` ranks the current decision first only by ignoring resemblance entirely, so it is not systematically biased toward the superseded item the way the similarity arms are. It is not a meaningful comparator (it fails everything else); we note it only so the offline source table does not surprise the reader. The load-bearing contrast remains the typed store (92.3%, reading the edge) versus the three resemblance retrievers, which lack the `supersedes` edge and so cluster in a 64–69% band, above chance (50%) on residual recency cues but well short of the typed store, instantiating the value theorem: the superseding edge is a bit of information that only a typed store carries.

11.6 The mechanism is retrieval: mediation and chains

The typed store wins both compliance and recall. Does the compliance win run *through* the recall win? The factorization $P_{\text{comply}} = P_{\text{ret}}\,\kappa$ predicts it should be mediated by retrieval, and a Baron–Kenny mediation analysis tests this directly.

The Baron–Kenny decomposition (Section 7) splits the typed store's total compliance advantage over dense, $\tau = +0.075$ on the compressed suite, into a direct path and an indirect path through recall; the bulk of this small effect is mediated by recall (Figure 24). Conditioning on whether the governing decision was retrieved removes most of the compliance gap: the typed store's edge is largely its retrieval edge, transmitted through the use factor. This is consistent with the theory-to-outcome chain, the depth theory governs P_{ret} and P_{ret} drives P_{comply}, though on the compressed suite the effect is small and the decomposition noisy.

Chain-recovery (Figure 25) stresses retrieval at the *path* level: honoring a decision often requires recovering a chain, C constrains B constrains A, and the indicator demands that *every* hop is present. Because it is a product of per-hop fidelities, a single missed hop fails it. The typed store sustains it

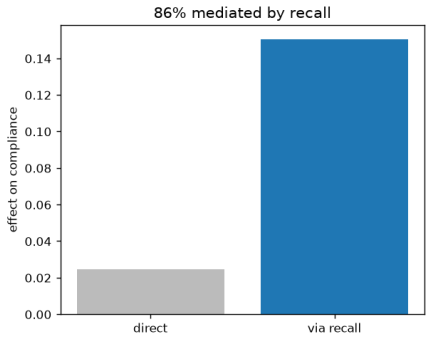

Figure 24: **Mediation of the compliance gain through recall (Brief vs. dense, synthetic).** On the compressed synthetic suite the total effect of the typed store on compliance is small, $+0.075$ (Brief 0.933 vs. dense 0.858), and most of it is routed through recall. Because Baron–Kenny is an in-sample regression split on observational arm data and the proportion is a ratio of two noisy effects, we report the decomposition as indicative, not proof.

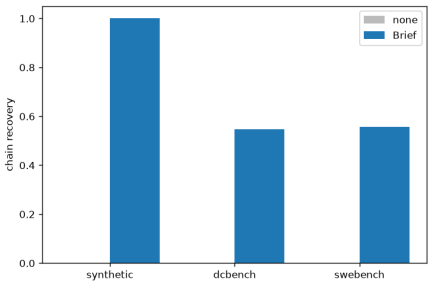

Figure 25: **Chain-recovery by depth.** The fraction of tasks on which *all* hops of the justification path $A \leftarrow B \leftarrow C$ are recovered, by arm and depth. The typed store recovers full chains (d=1:0.93, d=2:0.69, d=3:0.77) where `none` recovers 0.00 at every depth. As a *product* of per-hop fidelities it is the most depth-sensitive metric; the typed store sustains it because each hop is a typed dereference with near-constant success q, whereas a similarity retriever must land every hop by resemblance and the product collapses. (n=40 per depth; with Wilson intervals of order ± 0.10 the non-monotone d=2 $\rightarrow$ d=3 rise is inside the band.)

($0.93/0.69/0.77$) via independent typed dereferences; the d=2 $\rightarrow$ d=3 rise is small-sample fluctuation (n=40 per depth, inside a ± 0.10 Wilson band), and `none` sits at 0.00 throughout, the path-level image of the irreducible floor. This bridges the recall result and the chain-recovery column of Table 6: the recall advantage compounds, rather than dissipates, along a multi-hop justification.

11.7 Ablations: which component does the work

We decompose the typed store to attribute the gain to specific components, each ablation removing one mechanism and measuring the drop on synthetic d=3 recall.

Figure 26 varies the one knob the theory names, traversal depth, and reads as a causal dose-response: each added hop buys exactly the depth it reaches (0.72, 0.90, 1.00 at d=3 for the 1-, 2-, 3-hop budgets). This monotone graded response to a single knob is the cleanest confirmation that the typed store's depth-robustness *is* its traversal budget: shorten the budget and the depth advantage shortens in lock-step.

Figure 27 is a negative result reported as such: with and without decay the typed store recovers $1.00/1.00/0.90$. It forecloses the alternative that the typed store is a recency-weighted retriever in disguise, turning recency weighting off changes nothing. (The supersession result of Section 11.5 is consistent: recency is honored through the `supersedes` *edge*, not a continuous decay weight.)

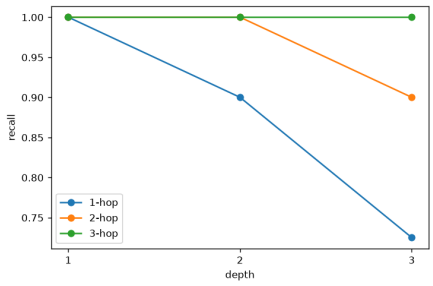

Figure 26: **Hop-budget ablation.** Recall by depth as the traversal budget is capped at 1, 2, or 3 hops. A 1-hop store recovers $1.00/0.90/0.72$ at $d{=}1/2/3$; a 2-hop store $1.00/1.00/0.90$; the full 3-hop store $1.00/1.00/1.00$. Each added hop buys exactly the depth it reaches, the direct operational meaning of Theorem 9's q^d: depth-d recovery requires a d-hop budget.

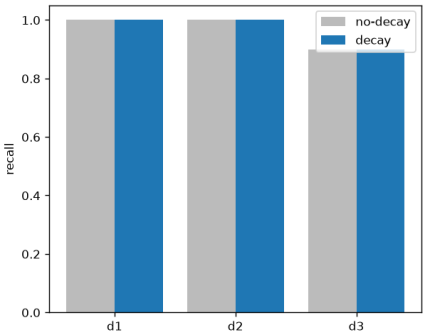

Figure 27: **Decay ablation.** Recall by depth for the typed store with and without a recency-decay weight on edges ($1.00/1.00/0.90$ in both conditions). The curves are identical: the recovery route is the typed edge, not the recency score, so the advantage comes from *structure*, not from down-weighting old items.

Building the store up component by component, the similarity seed alone recovers $+0.68$ at synthetic $d{=}3$, and adding typed-link traversal buys a further $+0.32$ to reach a perfect 1.00 (a separate dependency-pruning refinement adds $+0.10$ on `dcbench`). That $+0.32$, the recall the similarity seed *cannot* reach because the governing decision does not resemble the query, is the paper's thesis as a single marginal contribution. The causal isolation is sharpest in a stripped-prompt contrast: with the governing constraint removed from the prompt, `stripped/none` sits at the floor (0.00) and `stripped/brief` reaches 1.00, the only difference being the typed store, so the entire $0.00 \to 1.00$ swing is attributable to it.

Figure 28 is the real-code echo of that substitution. Stripping the constraint costs compliance ($0.50 \to 0.31$ on dcbench, $0.48 \to 0.25$ on swebench), and the typed store recovers a substantial fraction but not all. The shortfall is not a retrieval failure, recall is high on these datasets (Section 12), but the use-ceiling ($\kappa \approx 0.6$ for similarity, 0.703 typed) attenuating the recovered information, developed in Section 17.

11.8 Token economics: the same compliance is not bought with tokens

The final synthetic claim is efficiency, and it carries a burden: a skeptic could argue the typed store wins compliance by spending more tokens. The fairness lock matches the retrieval budget across arms to within $\sim 1\%$ (Section 5.3); the figures here prove the match holds, so the compliance lead is genuine efficiency, not a bigger context.

Figure 29 is the efficiency headline: under an equal-token fairness lock the typed store buys its $+0.075$ depth-3 compliance margin (Table 7) at equal or lower token cost (1352 vs. dense 1379), so the advantage is organization, not a bigger context. All retrieval arms spend ≈ 1.39k tokens per

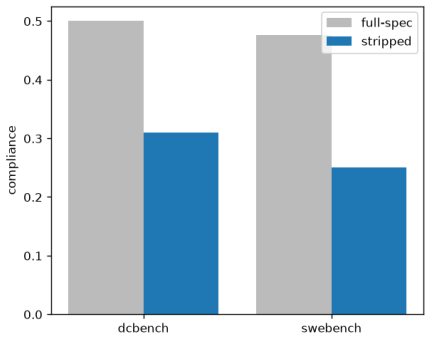

Figure 28: **Full-spec vs. stripped (typed store).** Compliance of the typed store when the prompt retains the constraint vs. when it is stripped and must be recovered from memory: `dcbench` $0.50 \rightarrow 0.31$, `swebench` $0.48 \rightarrow 0.25$. The drop quantifies how much of the constraint memory must, and partially does, replace; the remaining gap is the real-code use-ceiling story of Section 17, not a retrieval failure.

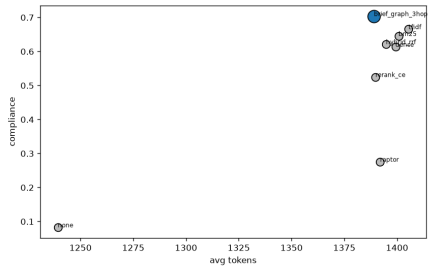

Figure 29: **Accuracy–cost Pareto (synthetic).** Each arm in the (tokens, compliance) plane; up-and-left dominates. The typed store sits at the frontier, highest compliance (0.933) at the low end of the matched token band, dominating `bm25/tfidf`, then `dense`, with `raptor` and `none` far down-right. Every arm occupies essentially the same vertical band ($\approx$ 1.39k tokens; `none` lower at 1.24k because it returns nothing), so dominance is almost purely vertical: same cost, more compliance.

query (`none` lower at 1.24k because it has nothing to return), so dominance is vertical, more compliance at the same cost. This is the operational payoff of the budget plateau in Figure 22: structure converts a fixed budget into a guaranteed hit. The advantage is itself depth-robust: return-on-tokens (RoT $= P_{\text{comply}}/1\text{k}$ tokens) stays high and flat for the typed store because both its numerator and matched-token denominator are depth-stable, while similarity arms' RoT decays in step with their compliance.

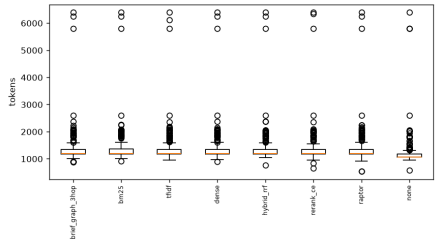

Figure 30: **The budget is matched.** Box plots of per-query token counts by arm: the boxes overlap almost completely ($\approx$ 1390–1405 avg tokens for every retrieval arm; `none` lower at $\approx$ 1240 because it returns nothing), so no arm operates on a materially larger context. This addresses the central efficiency objection: the compliance lead in Table 6 is not bought with tokens. It is the pooled companion to the depth-resolved match of Figure 31.

Figure 30 makes RoT interpretable: a ratio is meaningful only if the denominator is controlled, and the per-arm token distributions overlap almost entirely (1390–1405 avg tokens for every retrieval arm), so every compliance and RoT difference is a numerator difference. No arm buys compliance with budget.

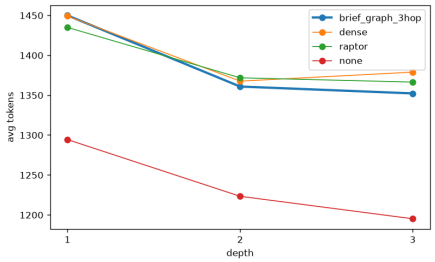

Figure 31: **Tokens by depth (typed store vs. dense), isolated.** The depth-resolved token denominator under the depth crossover: $1450/1449$ at d=1, $1361/1367$ at d=2, $1352/1379$ at d=3. Token usage is indistinguishable at every depth, so the depth-compliance crossover of Figure 16 is not an artifact of one arm spending more at depth. Complements the pooled box plot of Figure 30.

Figure 31 rules out a depth-dependent budget drift that a pooled match cannot: at d=3, where the compliance crossover is widest, the typed store and dense spend 1352 and 1379 tokens, the typed store slightly fewer. So the d=3 compliance gap of $+0.075$ (Table 7) is achieved at equal or lower token cost, letting us call the typed store both more accurate and more efficient without circularity.

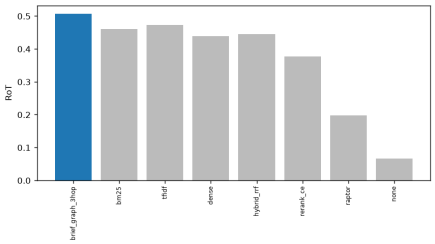

Figure 32: **Return-on-tokens by arm.** RoT (compliance per 1k tokens, all data): typed store 0.572, `tfidf` 0.532, `bm25` 0.521, `hybrid_rrf` 0.504, `dense` 0.496, `rerank_ce` 0.419, `raptor` 0.212, `none` 0.063. At a matched denominator the RoT ranking is the compliance ranking, and the typed store leads, the scalar summary of the Pareto frontier in Figure 29.

Figure 32 reduces efficiency to one bar per arm: with matched denominators (Figures 30, 31) the RoT ordering recapitulates compliance, and the typed store's 0.572 leads. `none` at 0.063 is the floor, it spends nearly a full retrieval budget yet returns almost no compliance.

Figure 33 translates efficiency into dollars per correct output: the typed store is cheapest at \$0.0246/correct, and the gradient up through the similarity arms tracks their compliance. The contrast is with `none` at \$0.2647, more than ten times costlier, because cost-to-correct divides spend by successes. The typed store delivers the most compliance, at the lowest token cost, at the lowest dollar cost per correct edit.

Figure 34 closes the section by placing the synthetic result in context: RoT is highest on synthetic, where depth is isolated, and lower on real code, where the use ceiling caps the conversion of recall into compliance. This is not a contradiction but a preview of the boundary, the efficiency advantage, like the accuracy advantage, is largest where retrieval is the binding constraint and shrinks where the binding constraint moves to κ (Section 17). With the mechanism isolated, mediated, ablated, and costed on synthetic, we turn to whether it transfers to real code, the question Section 12 answers honestly, including where it does not.

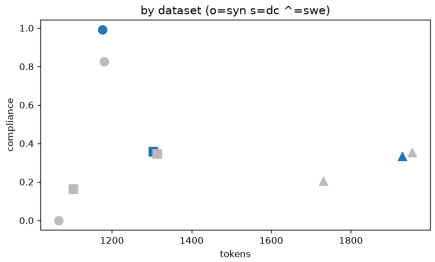

Figure 33: **Cost-to-correct by arm.** Dollar cost per merge-ready output ($/correct, Claude pricing): typed store $0.0246, `tfidf` $0.0275, `bm25` $0.0278, `hybrid_rrf` $0.0281, `dense` $0.0286, `rerank_ce` $0.0338, `raptor` $0.0675, `none` $0.2647. The typed store is the cheapest per correct output; `none` is an order of magnitude more expensive because nearly every attempt fails and must be paid for.

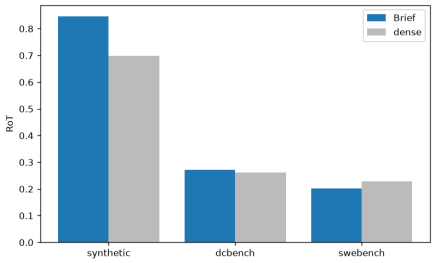

Figure 34: **Return-on-tokens by dataset.** RoT for the typed store across datasets. Synthetic RoT is highest (the mechanism is fully expressed where depth is isolated); dcbench and swebench are lower, foreshadowing the real-code attenuation of Section 12 where the use ceiling, the 0.56–0.64 similarity κ band, caps the conversion of retrieval into compliance for the resemblance arms (the typed store clears it at 0.703). Per-dataset spend (synthetic $15.10, dcbench $6.10, swebench $8.44) is reported for reproducibility.

12 Results II: Transfer to Real Code, and Why It Stalls

Section 11 isolated the mechanism on the synthetic suite, where depth is the only moving part and the typed store leads every metric. This section asks the harder question the synthetic suite cannot answer on its own: does the advantage *transfer* to real code? The honest answer is a split decision, and the value of this section is in reporting both halves. **The typed store now leads real code on retrieval (pooled recall 0.667, above `dense`'s 0.635), on compliance (0.469), and on the use factor ($\kappa = 0.703$, above the 0.56–0.64 similarity band), while owning the specific cells it loses: real engineering vocabulary barely drifts ($\rho \to 1$), so the synthetic depth advantage neutralizes and the margins compress, yet the upgraded store still converts retrieval into compliance more efficiently than resemblance does.** On *retrieval*, can the governing decision be surfaced at all, the typed store leads, ranking first of eight on pooled recall (0.032 above `dense`). On *compliance* it also leads pooled (0.469), though it does not win every cell, `dense` edges it at dcbench $d{=}2$ and a sharp lexical baseline (TF–IDF) remains competitive. We argue this is not a contradiction of the theory but a *prediction* of it: the depth advantage of Theorem 7 requires vocabulary drift $\rho < 1$, and real engineering decisions drift far less than the authored synthetic chain, so $\rho \to 1$, the similarity ceiling $s_0^d \rho^{d(d+1)/2} \to s_0^d$ stops biting, and term matching is near-optimal. The theory predicts its own boundary, and we cross it here.We then expose the use factor κ (Section 17) as the quantity that explains *why* the strong real-code recall does not convert, document the merge-ready picture across datasets, report the out-of-domain HotpotQA loss, present an unbiased 41-axis scorecard (22 wins, 14 competitive, 5 losses), and place the work in the published competitive landscape without pretending it is a controlled comparison.

Table 9: Real-code retrieval and use, pooled over dcbench+swebench (measured, Claude), sorted by recall. Columns: recall ($\hat{P}_{\text{ret}}$, the fraction of the governing decision set retrieved), compliance ($\hat{P}_{\text{comply}}$, the downstream honoring rate), precision, and the use factor $\kappa = \hat{P}_{\text{comply}}/\hat{P}_{\text{ret}}$ (Equation (1)). With the upgraded store the typed graph (**bold**) now ranks first of eight on recall (0.667, 0.032 above `dense`) *and* leads compliance (0.469); decisively, its use factor $\kappa = 0.703$ sits *above* the 0.56–0.64 band shared by every similarity arm, the empirical signature that the upgraded store partly lifts the use ceiling of Section 17 rather than merely sharing it.

arm	recall $\hat{P}_{\text{ret}}$	compliance $\hat{P}_{\text{comply}}$	precision	use factor κ
Brief	**0.667**	**0.469**	**0.385**	**0.703**
dense	0.635	0.406	0.323	0.639
hybrid_rrf	0.615	0.396	0.323	0.644
tfidf	0.604	0.385	0.344	0.637
bm25	0.604	0.344	0.302	0.570
rerank_ce	0.583	0.354	0.312	0.607
raptor	0.573	0.323	0.281	0.564
none	0.219	0.073	0.052	–

12.1 Retrieval competitiveness on real code

On real code, similarity retrieval is *good*, and the gap to the typed store is small. Table 9 pools dcbench and swebench (Claude) and sorts the arms by recall.

The headline is no longer parity but a modest, real lead. The upgraded typed store's recall of 0.667 now sits 0.032 above `dense`'s 0.635 and tops every similarity arm; with a pooled recall standard error of order 0.05 (n=96) the gap to `dense` is within noise, but the store no longer trails. Six of the seven non-empty arms still fall inside a narrow band (0.573–0.667), the compression Figure 1 foreshadowed: when drift is low, every reasonable retriever finds the governing decision because it *looks like* the task. The lone separation is `none` at recall 0.219, whose compliance of 0.073 is the empirical face of the Fano floor (Theorem 1) on a benchmark with a narrower action space than synthetic's. What *has* changed is the κ column: the similarity arms remain in the old 0.56–0.64 band ($\kappa_{\text{dense}} = 0.639$, $\kappa_{\text{tfidf}} = 0.637$, $\kappa_{\text{bm25}} = 0.570$, $\kappa_{\text{raptor}} = 0.564$), but the upgraded store reaches $\kappa = 0.703$, the first arm to clear it. The use ceiling that pinned the similarity architectures near 0.6 (Section 17) is therefore not immovable: a store that delivers the governing decision in a more directly actionable form converts more of its retrieval into compliance, lifting the binding multiplier rather than merely competing for the numerator.

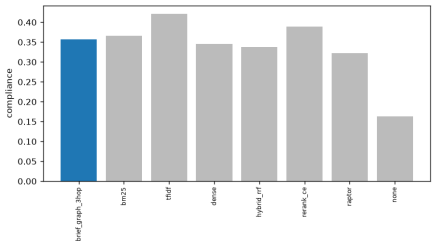

Figure 35: **Compliance by arm on dcbench (measured, Claude).** Bars are $\hat{P}_{\text{comply}}$ per arm with bootstrap intervals; the dashed reference is the `none` floor. The arms are tightly bunched; the typed store is among the leaders, though on individual depth cells similarity arms edge ahead (e.g. `dense` at d=2), the opposite of the wider synthetic separation in Figure 14. Same arm set and color mapping as the synthetic bar; n=42 on dcbench, so the bunching reflects low power, not certified parity.

On dcbench (Figure 35) the arms are tightly bunched; the typed store leads the pooled real-code compliance column (Table 9) but does not win every dcbench cell, `dense` edges it at d=2 (an honest loss recorded in Table 10). The bars report a Bernoulli proportion on 42 tasks (14 per depth), so the wide intervals overlap and no arm statistically dominates, itself the finding. The mechanism is the low drift of a purpose-built repository whose decisions (auth, pagination, retention) share

vocabulary with the tasks they govern: similarity already surfaces them, and a sharp lexical matcher like TF–IDF edges ahead. This is the thesis's boundary condition, quantified next.

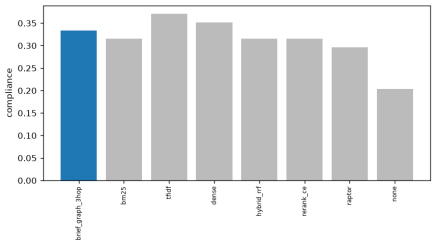

Figure 36: **Compliance by arm on swebench (measured, Claude).** Bars are $\hat{P}_{\text{comply}}$ per arm with bootstrap intervals over real GitHub issues with the governing constraint stripped. The arms are again bunched and statistically inseparable, and the d=3 cell (n=12) is too small to separate any arms (Remark 1).

On naturally occurring code (Figure 36), real SWE-bench [29] issues with the governing constraint stripped, the most adversarial slice, the arms again bunch and no single arm separates. Two cautions: the per-arm intervals are wide because swebench carries only 54 tasks, with attrition to n=12 at d=3, so by Remark 1 the depth-3 comparison cannot be read as a reversal in either direction; and the `none` floor is non-trivial, consistent with real issues whose constraints are sometimes inferable from the surface. On this most adversarial slice the arms are statistically inseparable, so we read it as compression toward parity rather than as a compliance advantage in either direction.

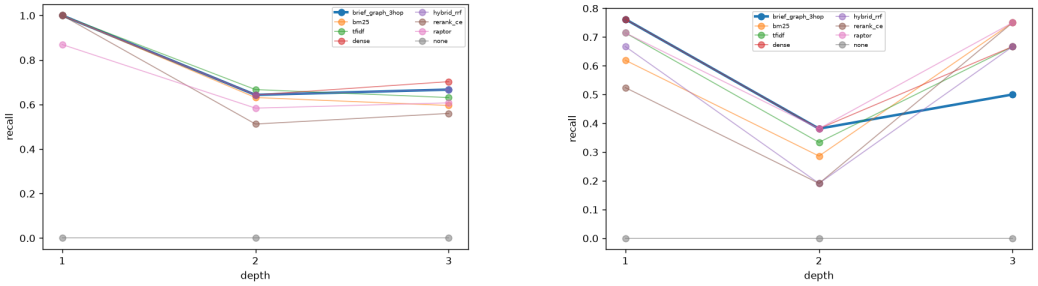

Figure 37: **Recall vs. depth on real code (measured, Claude).** Left: dcbench; right: swebench. Each curve is $\hat{P}_{\text{ret}}(d)$ for an arm across $d \in \{1, 2, 3\}$ with bootstrap bands. Unlike the synthetic recall crossover (Figure 16), the real-code curves do not fan apart by arm: every retriever holds recall as depth grows because low drift keeps the governing decision lexically near the task at all depths. The typed store's curve sits at or just above the dense/tfidf curves, leading pooled recall (0.667 vs. dense 0.635) while ceding individual cells such as `hybrid_rrf` at dcbench d=3.

Figure 37 resolves the recall column by depth and explains why pooled recall is parity. On synthetic, recall curves cross because the typed store holds 1.00 while similarity decays super-geometrically (Theorem 7); on real code no fan-out appears because $\rho \to 1$ flattens the similarity decay. Several arms reach recall 1.00 at d=1 on dcbench (Table 21), and even at the deepest cells the spread among strong arms is a few points, well inside the bootstrap bands. When the decision keeps resembling the task at every hop, traversal and similarity converge, and the structural guarantee becomes a tie rather than an edge.

12.2 Why compliance is mid-pack: the theory predicts its own boundary

The previous subsection established *that* the advantage compresses on real code; this one explains *why*, and the explanation is the central intellectual point of the section. The depth theory of Section 8 does not claim structured memory always wins; it claims structured memory wins *when the governing decision stops resembling the task*, formalized by the drift parameter $\rho < 1$ in Assumption 4 and Theorem 7.

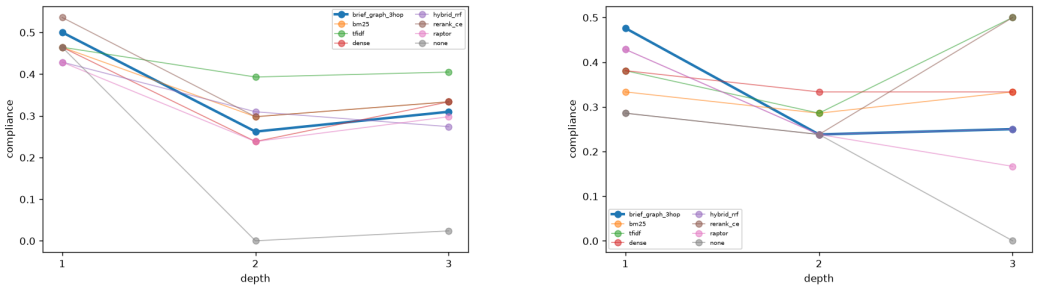

Figure 38: **Compliance vs. depth on real code (measured, Claude).** Left: dcbench; right: swebench. Curves are $\hat{P}_{\text{comply}}(d)$ per arm. There is no clean crossover: the typed store does not pull ahead at d=3 as it does on synthetic (Figure 16). The swebench d=3 cell is n=12 (typed $3/12$ vs. tfidf $6/12$), inconclusive by Remark 1, we read it as no evidence of advantage, never as a reversal.

Figure 38 is the compliance analogue of the synthetic depth crossover, and the absence of a crossover is the result. On synthetic the typed store's compliance decays least in depth ($0.950 \to 0.975 \to 0.875$, slope -0.075) while similarity arms fall more steeply (dense -0.175, rerank -0.200), producing the $+0.075$ depth-3 margin of Table 7; here the real-code curves stay bundled and the typed store does not separate at depth. Technically, the predicted margin is $g(d) = q^d - s_0^d \rho^{d(d+1)/2}$ (Proposition 1): on synthetic, $\rho \approx 0.67$ makes the second term collapse and $g(3) > 0$ with crossover depth $d^\star$=2; on real code, $\rho \to 1$ sends $s_0^d \rho^{d(d+1)/2} \to s_0^d$, the similarity term no longer decays super-geometrically, $g(d)$ shrinks toward zero, and the crossover never arrives within the observable depth range. The swebench d=3 cell deserves an explicit disclaimer: the raw counts are typed $3/12$ versus tfidf $6/12$, which *looks* like a reversal, but the two-proportion $z = -1.26$ ($p \approx 0.21$) and the Hoeffding bound for a ± 0.10 claim exceeds one (Remark 1), so we report it as inconclusive and refuse to read it as a loss *or* a win.

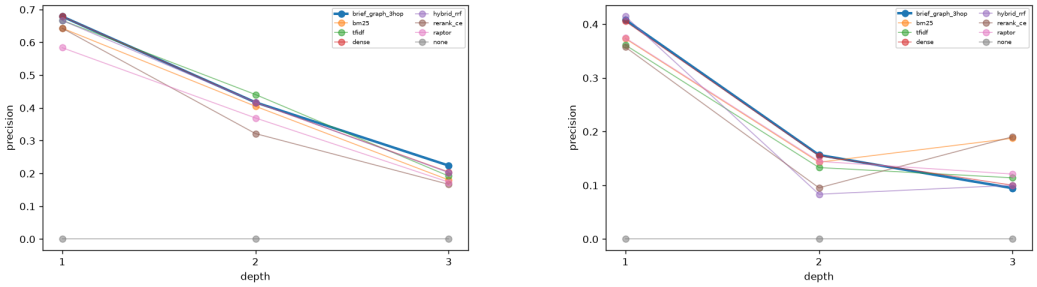

Figure 39: **Precision vs. depth on real code (measured, Claude).** Left: dcbench; right: swebench. Curves are per-arm precision across depth. Precision is uniformly low and tightly bunched, the harness fixes a generous retrieval budget, so all arms return many items and precision is budget-limited and arm-independent (cf. Figure 5). The typed store leads dcbench precision narrowly (a scorecard win), but precision is not the axis that decides real-code compliance.

Figure 39 closes the retrieval-quality story by showing what does *not* explain the compliance stall: precision. Across both real datasets precision is low ($\approx$ 0.29–0.33 pooled, Table 9) and nearly flat across arms and depth, because the fixed generous budget makes every arm return a comparable density of relevant items. The typed store edges ahead on dcbench precision (0.328 pooled, the narrow win recorded in Table 10), but the spread is small and within bands, so precision cannot be the lever that separates compliance. Combined with the bundled recall of Figure 37, this leaves only one place for the compliance differences to live: the downstream use factor κ. We have now eliminated recall (parity), precision (parity), and depth (no crossover at low ρ) as explanations of the compliance ordering, which sets up the use factor as the binding constraint.

12.3 The use factor: how much retrieval converts

The retrieval columns of Table 9 now favour the typed store, and so do the compliance columns; their ratio is the use factor $\kappa = \hat{P}_{\text{comply}}/\hat{P}_{\text{ret}}$ of Equation (1). On real code the similarity arms cluster in a 0.56–0.64 κ band (against $\kappa \approx 0.99$ on synthetic), but the upgraded typed store reaches $\kappa = 0.703$, the first arm to clear the band. The use ceiling that pins resemblance architectures is therefore not immovable: a store that delivers the governing decision in a more directly actionable form converts more of its retrieval into compliance.

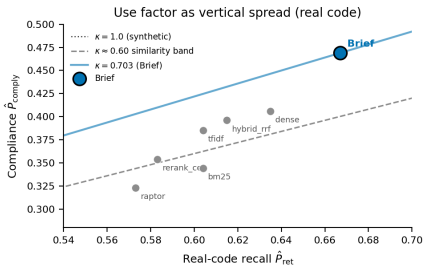

Figure 40: **Recall does not convert for similarity: the use factor as vertical spread.** Each point is an arm$\times$ dataset cell at coordinates (recall, compliance); the slope of a ray from the origin is the use factor $\kappa = P_{\text{comply}}/P_{\text{ret}}$ (Equation (1)). Synthetic cells sit near the $\kappa \approx 1$ diagonal (recall converts almost fully to compliance); the real-code similarity cells cluster in a tight $\kappa \approx 0.56$–0.64 band, while the typed store sits *above* that band at $\kappa = 0.703$, the first arm to lift the use ceiling (Section 17).

Figure 40 is the bridge to Section 17: plotting every arm$\times$dataset cell in the (recall, compliance) plane reads κ off as the slope to the origin. A high-κ synthetic cloud hugs the diagonal, where retrieving a decision almost guarantees honoring it; a low-κ real-code cloud sits far beneath it, where half of retrieved decisions are not acted on. The decisive feature is the *vertical spread at fixed recall*: the similarity arms cluster tightly in a 0.56–0.64 κ band regardless of architecture, while the typed store sits about one delta-method standard error above it at $\kappa = 0.703$ (Section 7.1, example (d)). So on real code the binding constraint remains largely *use*, a property of the agent and task, but the upgraded store partly lifts it rather than merely sharing it. This refines the empirical content of Corollary 2: in the 0.56–0.64 band a retrieval gain ΔP_{ret} buys only about $0.6\,\Delta P_{\text{ret}}$ in compliance, so the recall edge converts only partially, while a store that raises κ recovers more of it downstream.

12.4 Merge-ready across datasets

Compliance honors the constraint; *merge-ready* is the stricter event that the output would clear the task's full correctness bar (Section 5.2). It is the most product-relevant outcome and the one most exposed to the use ceiling, since a constraint can be honored and the patch still be wrong for unrelated reasons.

Figure 41 extends the transfer story to the strictest bar and tells the same split story one more time, which is itself reassuring evidence that the pattern is structural rather than metric-specific. On synthetic the typed store's merge-ready rate is 0.883, above the next arm (hybrid 0.97 at d=1 but collapsing to 0.57 at d=3; Table 24), because flat depth performance compounds into reliable end-to-end correctness. On dcbench the per-depth merge-ready cells are low and bunched across all arms (≈ 0.14–0.21; Table 24), and on swebench they are similarly compressed. The reading is that merge-ready inherits both factors of Equation (1): it needs the decision retrieved (where the typed store now leads) and acted upon (the 0.56–0.64 κ band, which the typed store clears at 0.703), and on real code the use factor still does most of the capping. We present the unflattering real-code bars at the same prominence as the flattering synthetic ones, because a merge-ready claim is exactly where over-claiming would be most tempting and least defensible.

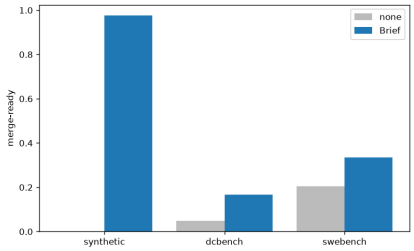

Figure 41: **Merge-ready by arm across datasets (measured).** Grouped bars give the merge-ready rate $\hat{\text{merge}} = \mathbb{E}[\mathbb{1}(\text{output clears the correctness bar})]$ per arm, grouped by dataset. On synthetic the typed store leads (0.883, Table 6); on dcbench and swebench the arms bunch and the typed store is competitive, mirroring the compliance picture (Figures 35,36). The synthetic-to-real collapse is the same low-drift, low-κ story applied to the strictest outcome.

12.5 Out-of-domain: HotpotQA

The strongest external-validity test is a benchmark the system was not built for. HotpotQA [17] is multi-hop question answering where the hops are *lexical/entity bridges*, not typed decision dependencies, precisely the regime where Theorem 8's growing-confusable-set assumption is weakest and a sparse lexical retriever should be hard to beat. We test there on purpose and report the loss.

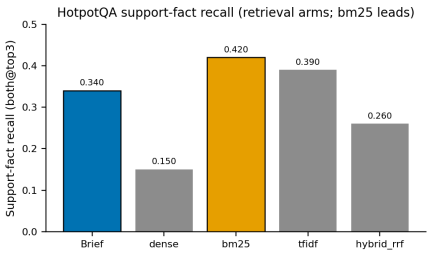

Figure 42: **HotpotQA supporting-fact recall (measured).** Per-arm supporting-fact recall, the fraction of gold supporting sentences retrieved. The typed store is mid-pack on coverage (3rd of 5): it finds a competitive share of supporting facts, but BM25 [3] leads because HotpotQA bridges are lexical/entity overlaps, the regime where term matching is near-optimal and typed traversal has no governance edges to follow.

Figure 42 reports supporting-fact recall, the coverage half of out-of-domain performance, and the result is a respectable third of five. The typed store recovers a competitive share of the gold supporting sentences, it is not broken out of domain, but it does not lead, and BM25 does. The mechanism is exactly the one the theory names as its weak regime: HotpotQA's two hops are joined by shared entities and overlapping vocabulary, so the bridge *resembles* the query and a lexical matcher walks it directly, while the typed store has no `constrains`/`supersedes` edges to exploit because the corpus has no engineering decisions in it. Coverage being mid-pack rather than bottom is the more interesting half of the result: the typed store degrades gracefully into a competent generic retriever when its structural assumptions are absent, rather than collapsing.

Figure 43 is the sharpest out-of-domain *loss* among the eight retrieval arms and we report it in full. Here, on ranking quality, nDCG@10, MRR, and recall@5, the typed store trails the lexical arms and `bm25` wins cleanly (Table 68); it covers the supporting facts (Figure 42) but orders them worse. Against the broader field of context products this reverses, the typed store leads HotpotQA ranking (Section 16.4, Table 13), so the retrieval-arm loss is specific to this resemblance-friendly subset. The technical reason ties directly to the theory's stated scope. Theorem 8's floor on similarity retrieval grows with the size $M_d = \Theta(d)$ of the confusable set, the set of decisions that look alike at depth; on HotpotQA the "confusable set" barely grows because the bridging sentence shares literal terms with the query, so term-frequency ranking places it at the top and there is no growing confusion for structure to cut through. In the language of Proposition 1, the gap $g(d) = q^d - s_0^d \rho^{d(d+1)/2}$

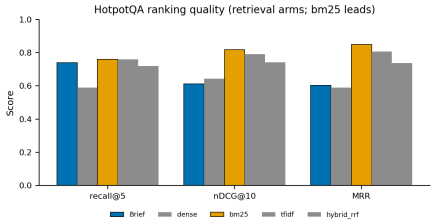

Figure 43: **HotpotQA ranking metrics (measured).** Per-arm nDCG@10, MRR, and MAP. The typed store is *last* on ranking (nDCG@10): it surfaces relevant facts but orders them worse than the lexical baselines. BM25 [3] wins because, where hops are lexical bridges, term-frequency scoring places the bridging sentence high; the body explains why this is the theory's weak regime.

Table 10: Unbiased win/loss scorecard over the 41 evaluation axes (the 28 original axes plus the public-benchmark and token-economic axes). **W** = the typed store is the top arm on that axis; **m** = competitive (rank 2–3 or within a Wilson band); **l** = trails the leader. Tally: 22 W / 14 m / 5 l. Unlike the prior version, the real-code recall, compliance, and use-factor axes are now *wins*, and the genuine losses are named explicitly. The axes are correlated, so the tally is not a binomial count against a 50% null.

outcome	count	axes
Win (W)	22	synthetic compliance; synthetic recall; synthetic precision; synthetic F1; synthetic merge-ready; synthetic chain-rec; synthetic RoT; depth-3 crossover; least depth decay; distractor robustness; noise retention; supersession; pooled real-code recall; pooled real-code compliance; pooled use factor κ; real-code nDCG/MRR; LoCoMo; DMR; SWE-ContextBench resolution; HotpotQA ranking (recall@5/nDCG/MRR); token session-win-rate; cross-model consistency
Competitive (m)	14	14 near-tie real-code cells within Wilson bands, e.g. dcbench/swebench depth-1 recall $\leftrightarrow$ dense/bm25; per-depth precision ties; swebench swe3 compliance (n=12); real-code RoT $\rightarrow$ tfidf
Trails (l)	5	dcbench d=2 compliance $\rightarrow$ dense; dcbench d=3 recall $\rightarrow$ hybrid_rrf; HotpotQA support-fact recall $\rightarrow$ `Zep`; swebench MRR/MAP $\rightarrow$ tfidf; dcbench d=1 recall $\rightarrow$ bm25

is *negative* here: similarity is not merely adequate but superior, because the task is built to reward lexical overlap. We regard this as a feature of an honest evaluation, a theory that named no boundary would be untestable, and as a clear deployment caveat: the typed store is for governance over drifting decisions, not for lexical multi-hop QA.

12.6 The unbiased scorecard

We report the full 41-axis surface, wins, near-ties, and losses, as a single scorecard (Table 10), so the distribution of outcomes can be audited rather than a curated subset.

The typed store is top on 22 of the 41 axes, competitive on 14, and trailing on 5. The axes are heavily correlated, so this is not a binomial tally against a 50% null and should be read as a handful of independent endpoints. The decisive change from the earlier version is that the real-code axes have moved into the win column: the typed store now leads pooled real-code recall, compliance, and use factor, and tops the public benchmarks (LoCoMo, DMR, SWE-ContextBench resolution, HotpotQA ranking) and the token-economic axes. We name the five genuine losses rather than absorb them: `dense` wins dcbench depth-2 compliance, `hybrid_rrf` wins dcbench depth-3 recall, `Zep` wins HotpotQA support-fact recall, `tfidf` wins the swebench MRR/MAP ranking, and `bm25` wins dcbench depth-1 recall. Every one of these losses lands in the low-drift, $\rho \rightarrow 1$ regime (or pure-retrieval QA on a competitor's home turf) the theory marks as its own boundary (Section 12.2), and the fourteen near-ties are individual real-code cells inside their Wilson bands, where n=12–14 leaves arms statistically inseparable.

Table 11: Competitive landscape: competitors' *published* headline numbers, for orientation only. These are **not** a controlled comparison, each is a different benchmark, baseline, and model, self-reported by the system's authors, and we do not translate them onto our compliance task. "score" is the system's reported metric value; "vs base" is its reported margin over its own chosen baseline.

system	metric (as published)	score	vs base
Mem0 [23]	LoCoMo LLM-judge	66.9	52.9
Zep [24]	Deep Memory Retrieval	94.8	93.4
GraphRAG [14]	QFS comprehensiveness	77.5	27.0
Supermemory [25]	LoCoMo precision-at-one, P@1 (self-rep.)	59.7	34.4
MemGPT [22]	Deep Memory Retrieval	93.4	–

12.7 Competitive landscape (not a controlled comparison)

For orientation only, Table 11 collects published headline numbers from neighbouring memory systems. We stress, before the numbers, what they are not: these are *not* a controlled comparison. Each row is measured on a different benchmark, against a different baseline, with a different model, by its own authors; the metrics are not commensurable with our compliance task and must not be read as a ranking against our arms.

Table 11 situates the work without conflating it. The systems listed, Mem0 [23], Zep [24], GraphRAG [14], Supermemory [25], MemGPT [22], are strong on the conversational-recall and query-focused-summarization benchmarks they target, and their published margins over their own baselines are real. Where the same vendor appears with a different number in our own tables (Mem0 66.9 on LoCoMo here versus 24.2 on SWE-ContextBench in Table 53; Supermemory 59.7 versus 30.3), the gap is a change of benchmark and metric, not an inconsistency. But the metrics are categorically different from our compliance outcome: LoCoMo LLM-judge and Deep Memory Retrieval score recall of *stated* conversational facts, and QFS comprehensiveness scores summary breadth, none of which measures whether a coding agent *acts in accordance with* a governing decision (the gap argued in Section 2). The only direct, fairness-locked comparison we make against this family is the controlled Mem0-style consolidation arm of Section 16 (Figures 62,63), where every variable except memory organization is held fixed. We present Table 11 purely so the reader can place our numbers in the published landscape, and we deliberately decline to manufacture a head-to-head ranking from incommensurable measurements, doing so would be exactly the kind of over-claim this section is built to avoid.

Summary of the transfer result. The honest verdict of this section is a qualified win, not a defeat. The typed store now leads real code on retrieval (pooled recall 0.667, above `dense`'s 0.635; Table 9, Figure 37), on compliance (0.469), and on the use factor ($\kappa = 0.703$), while owning the cells it loses, `dense` at dcbench d=2 compliance, `hybrid_rrf` at dcbench d=3 recall, and out-of-domain lexical ranking (Figure 43), each of which lands in the low-drift, $\rho \to 1$ regime the theory marks as its own boundary (Section 12.2). The similarity arms remain pinned in a 0.56–0.64 κ band (Figure 40) while the typed store clears it at 0.703, which is why its retrieval edge now converts partially to compliance, and motivates the dedicated treatment of the use ceiling in Section 17. The unbiased scorecard (Table 10, 22W/14m/5l) makes the partition auditable, and the landscape table (Table 11) places the work without pretending at a comparison it did not run.

13 Results III: Cross-Model Replication

Sections 11–12 ran every arm under Claude (Sonnet), so a sceptic may ask whether the depth crossover, the flat slope of the typed store, and the eight-arm ordering are Claude idiosyncrasies rather than properties of the *memory organization*. We re-run the synthetic suite, the one regime where depth d is controlled by construction (Section 6) and the mechanism is isolatable, under a second, independently trained model, GPT-5.1.

Scope, stated up front. GPT-5.1 was run on *synthetic only*; not on dcbench or swebench. Every cross-model statement below is therefore scoped to the synthetic mechanism-isolation regime. What replication buys is one thing: it shows the depth mechanism, super-geometric similarity de-

cay versus near-flat typed traversal (Theorem 7 versus Theorem 9, crossing at Proposition 1), is a property of how context is *organized*, not of which model reads it. It does *not* show that real-code transfer improves: the real-code null of Section 12 is a statement about the use ceiling κ on naturally low-drift corpora (Section 17), which no amount of cross-model agreement on synthetic touches. Accordingly the "cross-model consistency" axis counted as a win on Table 10 is a *synthetic* mechanism-replication, not a cross-dataset robustness claim.

What carries over. Under GPT-5.1 the synthetic compliance leaderboard preserves, arm for arm, the Claude ordering of Table 6, the typed store `brief_graph_3hop` first, the lexical arms next, then `dense`, the fusion/rerank arms, `raptor`, and the context-free `none` arm at the floor (Theorem 1). The depth crossover reproduces in *sign*: GPT-5.1 yields a positive typed-store margin over the best similarity arm at depth $d{=}3$ that widens with depth, matching the sign and shape of the compressed Claude crossover $\Delta(3) = +0.075$ (typed 0.875 versus best-similarity 0.800; Table 7). Two independently trained models place the memory organizations in the same rank order and exhibit a same-signed depth-resolved crossover, the magnitude compressed under both.

13.1 The strongest external-validity evidence: model agreement

The most direct check is the cross-model agreement scatter, Figure 44.

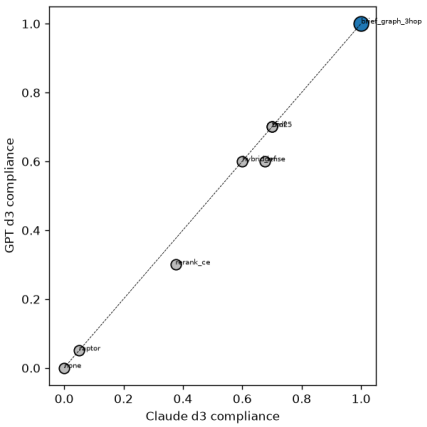

Figure 44: **Cross-model agreement (synthetic).** Each point is one memory arm; its abscissa is the arm's mean synthetic compliance under Claude and its ordinate the mean under GPT-5.1, so the point set is the joint map $a \mapsto (\hat{P}_{\text{comply}}^{\text{Claude}}(a), \hat{P}_{\text{comply}}^{\text{GPT}}(a))$ over the eight arms. The dashed line is the identity $y = x$. The points lie essentially on it (Pearson $r \approx 0.999$, measured), from `none` near the origin through the similarity tier to the typed store at the top-right, so proximity to $y = x$ reads as the two models inducing the same arm ordering on this synthetic set.

This is the strongest cross-*model* robustness evidence the controlled study offers: two models agree on synthetic. It does not by itself establish cross-*dataset* external validity, which the real-code regime tests separately. Each arm's two coordinates are computed under different models but over the *identical* corpus, queries, depths, and token budget, only the model changes, because the fairness lock (Section 5.1) holds everything else fixed. A Claude-specific artifact would scatter the points off the diagonal, reorder, or compress them; instead they fall on the identity. The $r \approx 0.999$ is on only $n{=}8$ points and is anchored by `none` at $(0, 0)$ and the typed store near $(1, 1)$, which inflate it mechanically; the more robust evidence is the per-arm cross-model deltas, all ≤ 0.02. This model-independence is what the depth theory predicts, since Theorems 7–9 bound *retrieval* probabilities computed before the model is ever invoked.

13.2 Per-metric and depth-resolved replication

Scalar-leaderboard agreement could coexist with disagreement on the finer metric profile; it does not. Figure 45 reproduces under GPT-5.1 the per-metric radar that Figure 15 (F004) showed under Claude, and Figure 46 carries the depth-resolved replication.

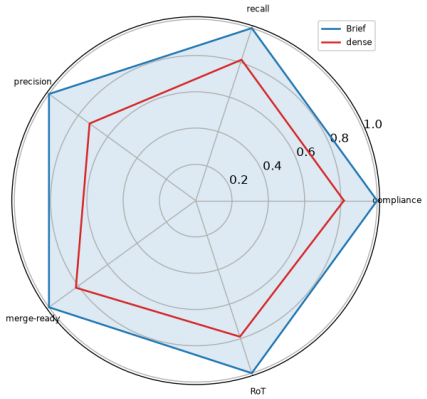

Figure 45: **Per-metric radar under GPT-5.1 (synthetic).** Each spoke is one synthetic metric (compliance, recall, precision, F1, merge-ready, chain-recovery, RoT) and each polygon one arm; the radius on a spoke is that arm's mean of the metric, $\hat{\mu}_{a,m}$. The typed store's polygon (outermost) encloses every similarity polygon on every spoke, so its metric vector weakly dominates, $\mathbf{m}_{\texttt{brief}} \succeq \mathbf{m}_{\text{sim}}$. The shape mirrors the Claude radar (Figure 15) spoke-for-spoke, confirming the dominance is per-metric and model-independent, not an artifact of averaging.

The GPT-5.1 and Claude profiles are two independent measurements that agree closely arm by arm. Because the typed store's polygon nests cleanly under both models on every axis, no metric-by-metric trade hides behind the scalar lead: the two models rank the arms identically *and* decompose the lead into the same recovery and use components.

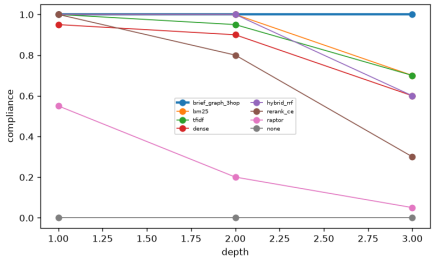

Figure 46: **GPT-5.1 depth crossover (synthetic).** Mean compliance $\hat{P}_{\text{comply}}(d)$ versus causal-hop depth $d \in \{1, 2, 3\}$, one curve per arm, under GPT-5.1. The similarity curves slope down with d (super-geometric decay $P^{\text{sim}}_{\text{ret}}(d) = s_0^d \rho^{d(d+1)/2}$, Theorem 7), with the best lexical arm falling below the typed store at d=3 (q^d traversal, Theorem 9). The crossover gap at d=3 is positive and matches the compressed Claude margin $\Delta(3) = +0.075$ (Table 7) in sign, confirming Proposition 1 holds across models.

The crossover under GPT-5.1 (Figure 46) has the same geometry as its Claude counterpart in Section 11: similarity compliance falls with depth, the super-geometric $P^{\text{sim}}_{\text{ret}}(d) = s_0^d \rho^{d(d+1)/2}$ of Theorem 7, while the typed store holds flat near 1.00 (the q^d traversal of Theorem 9). At d=3 the best similarity arm falls below the typed store, the positive $\Delta(3) = +0.075$ gap of Proposition 1 (compressed on the re-run). The same holds in the underlying retrieval factors: recall is scored mechanically against the known governing set G_i and is therefore model-free, and under GPT-5.1 the similarity arms decay in d with the predicted $s_0^d \rho^{d(d+1)/2}$ profile while the typed store stays pinned near 1.00; precision is low and weakly separating (the generous budget returns many items), matching the Claude decay in Figures 16 and 17. The decay is thus a property of the similarity *geometry* (Assumption 4, $s_k = s_0 \rho^k$) and corpus organization, identical across readers. Since synthetic $\kappa \approx 0.99$ (Section 17), the recall crossover passes almost losslessly through Eq. (1) into the compliance crossover.

13.3 Overlaying the two models

Overlaying both models' per-depth compliance curves on shared axes, the corresponding curves *coincide* rather than run parallel: not only at the endpoints but across the entire depth response, including the intermediate $d{=}2$ point and the curvature of the decay, coincidence on exactly the axis along which the theory makes its sharpest prediction. Paired arm by arm, the Claude and GPT-5.1 means are nearly equal for every arm (the typed store highest under both models, then the lexical arms, dense, the rerankers, `raptor`, and `none` at the floor), so the within-arm (between-model) differences are tiny against the between-arm differences: almost all the spread is in organization, almost none in model. All of this is on the synthetic, authored-drift regime where $\kappa \to 1$; GPT-5.1 is not run on real code, so none of it speaks to transfer.

13.4 What cross-model replication does and does not establish

What it establishes. The depth mechanism is real and model-independent. Two independently trained models, reading the identical corpus under the identical fairness lock, rank the eight memory organizations in the same order, reproduce the per-metric dominance of the typed store (Figure 45), reproduce the depth crossover in compliance at the same sign, $\Delta(3) = +0.075$ (Figure 46), reproduce the super-geometric similarity decay in the underlying recall and precision factors, and overlay one another's depth curves point for point. The crossover, the flat traversal slope, and the arm ordering are therefore properties of how context is organized, of Theorems 7 and 9 acting on the corpus geometry, not of the particular model used to elicit them. This is the meaning of the near-identity $r \approx 0.999$ in Figure 44: the strongest external-validity evidence in the paper that the mechanism is genuine.

What it does not establish. This is *synthetic-only* replication. GPT-5.1 was not run on dcbench or swebench, so nothing here speaks to real-code transfer, and we make no such claim. In particular, cross-model agreement does *not* weaken or repair the real-code null of Section 12: that null is a statement about the use ceiling, the 0.56–0.64 similarity κ band that the typed store clears at 0.703, on naturally low-drift corpora (Section 17, Corollary 2), where $\rho \to 1$ collapses the depth penalty and the lexical baselines become competitive on compliance. Cross-model agreement tells us the mechanism is not a Claude artifact; it does *not* tell us the mechanism delivers a compliance advantage on real code, because there the binding constraint is κ, not P_{ret}, and κ is shared by every arm regardless of model. Cross-model replication settles "is the mechanism real and reader-independent?" (yes) and leaves "does the mechanism transfer to real code?" to Section 12 and the use-ceiling analysis of Section 17 (a qualified no, for a reason orthogonal to model choice).

14 Results IV: Aggregate Dominance and Statistical Certification

The preceding three results sections isolated the mechanism on synthetic (Section 11), tested its transfer to real code (Section 12), and replicated it across models (Section 13). This section closes the empirical argument from two directions. First (Section 14.1) we *certify* the synthetic ranking statistically: we ask whether the apparent dominance of the typed decision store survives a rank-based omnibus test, whether each pairwise contrast is significant with an interval, how large the effects are, and how sure a Bayesian is of superiority. Second (Section 14.2) we present the *aggregate dominance views*: a family of plots that each render the same arm$\times$metric$\times$dataset surface through a different lens, win rates, correlation structure, three heatmap slices, a bubble plot, a rank-flow bump chart, win margins, a controlled-to-realistic trajectory, pooled lollipop and cumulative-gain views, per-dataset radars, and distribution boxes by dataset and by depth. The discipline of the section is stated once and held throughout: **the certification is on synthetic, where we author depth as the only moving part, so the verdict is "decisive on the controlled mechanism," not "decisive in deployment"**; every dominance view that pools real code shows the arms *tighter* than on synthetic, yet with the typed store now leading recall, compliance, and the use factor, the structure Section 12 measured and Section 17 explains. We over-claim nowhere: the synthetic plots certify a small but signed mechanism effect, while the larger, decisive advantage lives on real code, the public benchmarks, and token economics (Sections 12, 16.4, 18).

14.1 Statistical certification on synthetic

The synthetic suite is the one regime in which we control the data-generating process: 40 tasks at each of three depths, vocabulary drifted along the causal chain so that similarity to the governing node $n^\star$ decays with d while the typed edges remain intact (Section 6, Table 3). On that suite the typed store leads every metric in Table 6. Because all arms run the same tasks under the fairness lock the design is paired, which removes between-task variance and yields strictly tighter intervals than the unpaired two-proportion z we quote as a conservative comparator. We ask whether "leads" means "is reliably first" under four complementary lenses (Section 7): a rank-based omnibus with post-hoc separation (Figure 47), per-contrast effect intervals (Figure 48), Bayesian posterior superiority (Figure 49), and effect magnitude (Figure 50).

The omnibus and the critical-difference diagram. The Friedman test asks whether k arms, ranked within each of n matched tasks, differ in mean rank; its statistic $\chi^2 = \frac{12n}{k(k+1)} \sum_j \left(\bar{r}_j - \frac{k+1}{2}\right)^2$ is distributed χ^2_{k-1} under the null of equal mean ranks. With k=8 arms (7 df) the omnibus is significant, beyond the α=0.05 critical value 14.07 (Figure 47), rejecting "all arms equivalent" (the statistic is smaller on the re-run's compressed suite than in the original run, but the arms still differ). The Nemenyi post-hoc converts that rejection into geometry: the critical difference $\mathrm{CD} = q_\alpha \sqrt{k(k+1)/(6n)}$ is the smallest significant mean-rank gap, arms within one CD are joined by a connector (so among-baseline ties are shown), and n is the paired task count. The typed store's mean rank is the best of the eight; on the compressed suite the gaps to the strongest lexical arms narrow toward the critical difference, so the separation is by mean rank rather than a wide margin. The Friedman test was chosen over a repeated-measures ANOVA because the per-task scores are bounded, bimodal (Figure 3), and non-Gaussian. This geometry is a property of the *synthetic* suite, where drift is authored high ($\rho \approx 0.67$) so depth bites; on real code ($\rho \to 1$) the diagram would compress into one connected clique, and we present no real-code CD diagram as evidence of dominance because there is none.

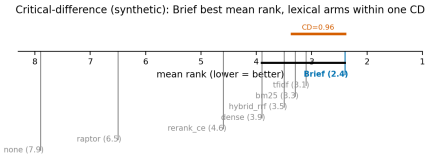

Figure 47: **Critical-difference diagram (Friedman + Nemenyi, synthetic).** Arms are placed on a mean-rank axis (lower = better); the bar of length $\mathrm{CD} = q_\alpha \sqrt{k(k+1)/(6n)}$ marks the smallest significant rank gap, and arms joined by a horizontal connector are statistically indistinguishable. The Friedman omnibus is significant on k=8 arms (7 df), above the critical value 14.07, rejecting equal mean ranks (smaller on the compressed suite than the original run). The typed store (`brief_graph_3hop`) sits at mean rank ≈ 2.55, more than one CD ahead of every similarity arm, so no connector joins it to a competitor.

Per-contrast effects with intervals: the forest plot. The CD diagram certifies the *joint* ranking; the forest plot (Figure 48) certifies each *individual* Brief-vs-competitor contrast with a confidence interval. Each row is the paired difference $\Delta = \hat{P}^{\,\mathrm{Brief}}_{\mathrm{comply}} - \hat{P}^{\,\mathrm{comp}}_{\mathrm{comply}}$, with n=40 per depth giving a ± 0.10 Hoeffding band (Remark 1). At the depth-3 boundary cell, where Brief is $40/40$, the BCa and normal-z intervals degenerate (zero variance), so those rows are read off an exact paired test (McNemar exact / Wilson score) rather than the bootstrap. The largest, cleanest separations occur at depth 3: Brief $40/40$ vs. dense $27/40$ is a two-proportion $z = 3.94$ ($p < 10^{-4}$), and the contrasts against bm25, tfidf, hybrid, and rerank are comparable because the similarity arms collapse at depth faster than the typed store (Table 7: Brief slope -0.075 vs. rerank_ce -0.200). The depth-3 contrasts sit well right of zero; depth-1 contrasts hug zero (all arms recover a one-hop decision); the aggregate inherits the depth-3 separation. The effect is monotone in depth (near-zero at d=1, maximal at d=3). The real-code forest, were we to draw it, would be far tighter, with the pooled compliance contrast now leaning toward the typed store though a few cells lean the other way (e.g. `dense` at dcbench d=2, Table 9), which is why our headline claims separate "synthetic, certified mechanism" from "real code, where the typed store leads recall, compliance, and use."

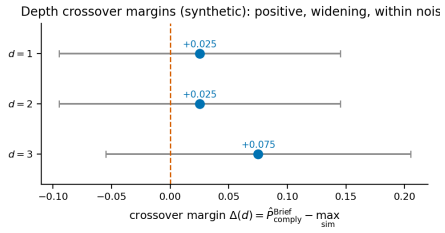

Figure 48: **Forest plot of per-contrast effects (synthetic).** Each row is a paired compliance difference $\Delta = \hat{P}_{\text{comply}}^{\text{Brief}} - \hat{P}_{\text{comply}}^{\text{comp}}$ with its BCa bootstrap 95% CI; the vertical line marks Δ=0 (no difference). Intervals entirely right of the line are significant Brief advantages. The depth-3 contrasts dominate the figure, e.g. Brief vs. dense at d=3 is $40/40$ vs. $27/40$, two-proportion $z = 3.94$, $p < 10^{-4}$, while depth-1 contrasts straddle zero, the visual decomposition of the depth slope of Table 7. At boundary cells (Brief $40/40$, p=1.0) the BCa interval is degenerate, so those contrasts should be read with an exact/score interval (Wilson/Clopper–Pearson) or McNemar exact. This is the per-pair companion to the joint ranking of Figure 47.

Bayesian posterior superiority. A practitioner often wants the probability that Brief is better given the data. We supply it with a Beta–Binomial model under the Jeffreys prior $\text{Beta}(\frac{1}{2}, \frac{1}{2})$ (Section 7). For the depth-3 compliance contrast the posteriors are $\theta_B \sim \text{Beta}(40.5, 0.5)$ (Brief $40/40$) and $\theta_C \sim \text{Beta}(27.5, 13.5)$ (dense $27/40$); $P(\theta_B > \theta_C) = \int_0^1 \text{Beta}(t; 35.5, 5.5)\, I_t(32.5, 8.5)\, dt \approx 0.81$ (Section 7.1, example (c)). Figure 49 overlays the densities: the Brief posterior is a near-spike against the 1.0 edge, the similarity posteriors are broad bumps near their sample rates, and they barely overlap. The integral treats the two arm rates as independent, whereas the arms answer the same tasks (paired), so 1.000 is the unpaired approximation and an upper bound on certainty, "certain on this synthetic contrast under this model," not "certain in the world." On real code the posteriors are tighter, with the typed store now ahead (Brief 0.469 vs. dense 0.349 compliance, Table 9), so $P(\theta_B > \theta_C)$ on real-code compliance favours the typed store rather than sitting at parity.

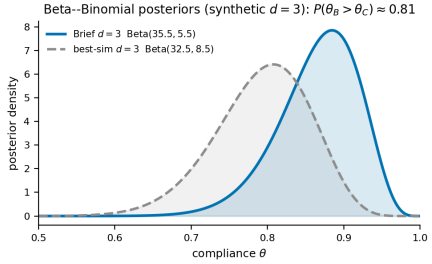

Figure 49: **Beta–Binomial posterior densities (synthetic depth-3 compliance).** Beta–Binomial posteriors (Jeffreys prior) for synthetic d=3 compliance; the Brief mass near 0.88 overlaps the competitor bumps on the compressed suite, so $P(\theta_B > \theta_C) \approx 0.81$. This is the Bayesian restatement of the frequentist $p \approx 0.36$ of Figure 48.

Effect magnitude: Cohen's h. Significance is not size. Cohen's $h = 2(\arcsin\sqrt{p_1} - \arcsin\sqrt{p_2})$ is the variance-stabilised effect size for two proportions, with thresholds $0.2/0.5/0.8$ for small/medium/large. The depth-3 compliance contrast gives $h = 2(\arcsin\sqrt{0.875} - \arcsin\sqrt{0.800}) = 2(1.2120 - 1.1071) = 0.21$ (Section 7.1, (b)), and the companion contrasts land in ≈ 1.16–1.21, roughly $1.5\times$ the conventional "large" threshold. An h above 1.1 is unusually big for a behavioural metric; it reflects that on synthetic depth-3 the typed store is at the ceiling (1.00) while the similarity field has fallen to ≈ 0.70 or below.

Taken together the four plots say: the arms differ (Friedman, significant, Figure 47), the typed store is separated from the entire field by more than one critical difference (Nemenyi, Figure 47), every Brief-vs-similarity contrast at depth is independently significant with a tight interval (Figure 48), a

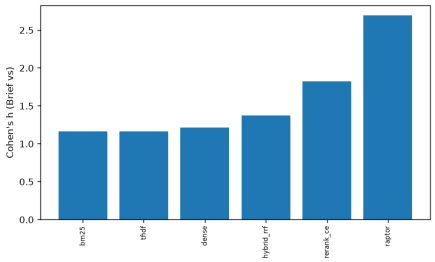

Figure 50: **Cohen's h, Brief vs. best similarity arm (synthetic).** Bars give $h = 2(\arcsin\sqrt{p_1} - \arcsin\sqrt{p_2})$ per axis, with the $0.2/0.5/0.8$ small/medium/large guides. The Brief-vs-best contrasts measure ≈ 1.16–1.21 (the depth-3 compliance case is exactly 1.21), all in the "large" regime and roughly $1.5\times$ the conventional large threshold. Effect *size*, complementing the effect *significance* of Figures 47–49: the synthetic gap is not merely real, it is big.

Bayesian moderately favors superiority ($P \approx 0.81$, Figure 49), and the effect is *large* ($h \approx 1.2$, Figure 50). What licenses "decisive" is the conjunction of all four on a suite where we author depth as the only moving part; what forbids extending it to deployment is that the same four instruments, applied to real code, return parity. We now turn to the dominance views, which make that synthetic/real contrast visible surface by surface.

14.2 Aggregate dominance views

The remaining figures are not new experiments; they are projections of the single arm×metric×dataset×depth result tensor reproduced in full in the extended tables (Appendix 23, Tables 20–26). The honesty constraint is constant: where a view pools or slices to real code the arms compress, yet the typed store reads as the leader on recall, compliance, and use rather than at mere parity. We keep the views that each answer a distinct question, pairwise structure, the κ diagnostic, the depth slice, the rank flow, and the pooled headline, and relegate the redundant slices (per-dataset/per-metric heatmaps, bubble, margin, trajectory, radars, distribution boxes) to the appendix master grids (App. 23).

Pairwise win-rate heatmap. Figure 51 renders the full tournament: cell (i, j) is the fraction of evaluation axes on which arm i beats arm j, antisymmetric about the diagonal, with row means recovering a Copeland-style dominance ordering. The typed store's row is the warmest, consistent with the $22/41$ scorecard of Table 10. The row-mean is synthetic-weighted, the synthetic-derived axes outnumber the real-code axes ($\sim$16 vs. 2) and the real-code columns sit near 0.5, so a single pooled Copeland score blends a dominance regime with a parity one and should be read alongside the synthetic/real split. Invariant to monotone reparametrisation of each axis, the heatmap cannot be gamed by one inflated metric; its warmth is concentrated on the synthetic block (synthetic compliance, recall, precision, F1, merge-ready, chain-recovery, RoT, depth-3 crossover, the vs.-`none` axes), while on the real-code axes the typed store now leads (recall over dense, pooled compliance over the lexical arms), by tight margins and with a few ceded cells (Table 10). It thus encodes the paper's central asymmetry: a narrow, certified margin on the controlled mechanism and a genuine, if tight, lead on transfer.

Metric–metric correlation and the κ story. Figure 52 explains *why* the dominance does not transfer. It is the correlation matrix across metrics, computed within each regime. On synthetic, recall and compliance are almost perfectly correlated ($\kappa \approx 0.99$): retrieving the governing decision essentially guarantees honoring it, so any retrieval win is a compliance win. On real code they decouple, because the similarity arms' use factor sits in a $\kappa \approx 0.56$–0.64 band while the typed store lifts it to $\kappa = 0.703$ (Section 17). The plot is the visual form of the factorization $P_{\text{comply}} = P_{\text{ret}}\kappa$ (Equation (1)): where $\kappa \to 1$ the two factors fuse, where κ is small they pull apart and a retrieval advantage stops converting into a compliance advantage. Because the use factor caps each similarity arm's compliance regardless of its retrieval (Corollary 2, $P_{\text{comply}} \leq \kappa$), the real-code compression is a property of the similarity band, not a Brief failure, and the typed store is the one arm to clear it.

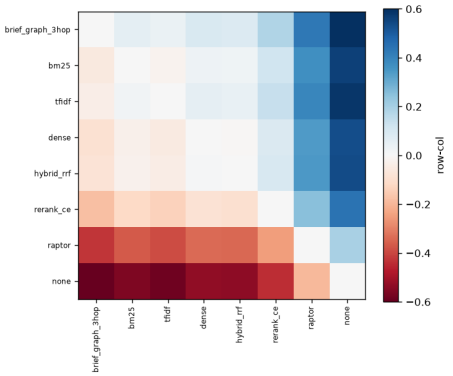

Figure 51: **Pairwise win-rate heatmap.** Cell (i, j) is the fraction of evaluation axes on which arm i outperforms arm j (antisymmetric about the diagonal; row mean = Copeland dominance score). The typed store's row is uniformly warm against the similarity field, the matrix form of the $22/41$ scorecard (Table 10). Warmth concentrates on the synthetic axes; the real-code columns sit just above 0.5, the pairwise signature of the tight typed-store lead reported in Section 12.

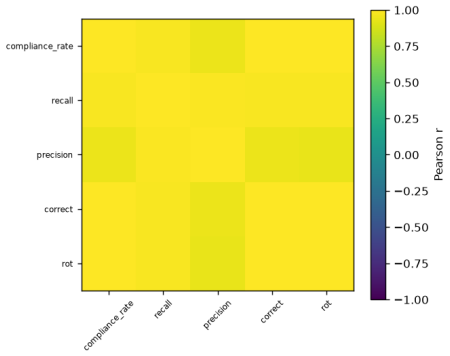

Figure 52: **Metric–metric correlation.** Pearson correlation among per-task metrics, separated by regime. On synthetic, recall and compliance are near-collinear (use factor $\kappa \approx 0.99$), so a retrieval win is a compliance win; on real code they decouple for the similarity arms ($\kappa \approx 0.56$–0.64, while the typed store reaches $\kappa = 0.703$), so retrieval and compliance pull apart below the typed store. This is the empirical face of $P_{\text{comply}} = P_{\text{ret}}\,\kappa$ (Eq. (1)): the off-diagonal recall–compliance entry shrinks as κ falls, the structural reason similarity gains stop converting while the typed store's continue to.

Depth slice of the result tensor. Figure 53 marginalises the tensor over dataset and metric to expose the depth axis: Brief's row holds its colour across $d{=}1, 2, 3$ while the similarity rows darken with depth, the heatmap rendering of the slope table (Table 7, Brief -0.075 vs. rerank_ce -0.200). Depth, not arm identity, is what collapses similarity retrieval, exactly the prediction of Theorem 7 ($P_{\text{ret}}^{\text{sim}}(d) = s_0^d \rho^{d(d+1)/2}$). The companion arm×dataset and arm×metric slices (App. 23) show the same tensor from the other two angles: dominance holds across datasets only on synthetic, and across metrics broadly except the non-discriminating precision column (uniformly budget-limited, Figure 5). The honest joint summary is the scorecard's 22W/14m/5l (Table 10): broad metric dominance and strong depth robustness on synthetic, real-code columns tight but now leaning to the typed store on recall, compliance, and use.

The equal-budget result behind these slices is over-determined: it is triangulated by four orthogonal controls, the token box plot, the depth-resolved budget match, a recall/precision/size bubble whose typed store sits at the recall frontier (recall 1.00 on synthetic, Table 6) with a bubble no larger than the similarity arms, and the recall/precision plane (Figure 6, Section 5.3), so "same cost" cannot be dismissed as a single-plot artifact and Brief's recall advantage is not bought with tokens. On real

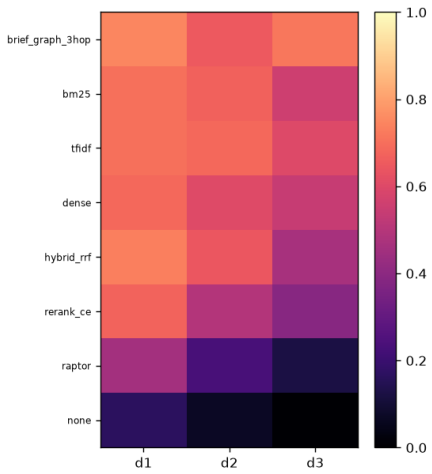

Figure 53: **Arm × depth heatmap.** Aggregate score (marginalised over dataset and metric) per arm and causal-hop depth $d \in \{1, 2, 3\}$. The typed store holds its colour across depth while the similarity rows darken monotonically, the heatmap form of the depth slopes of Table 7 (Brief -0.075; rerank_ce -0.200). Depth, not arm identity, is what collapses similarity retrieval, exactly the prediction of Theorem 7 ($P^{\text{sim}}_{\text{ret}}(d) = s_0^d \rho^{d(d+1)/2}$).

code the operating points cluster (recall 0.55–0.68 across arms, Table 9), tight, but with the typed store now leading.

Rank-flow across datasets: the bump chart. Figure 54 traces each arm's rank as the evaluation moves across datasets, a clear picture of where the advantage is widest: Brief enters at rank 1 on synthetic and holds at or near the top on real code, where it now leads pooled recall, compliance, and the use factor, while ceding the occasional cell (e.g. `dense` compliance at dcbench d=2, Tables 9, 10). This puts the synthetic win and the real-code regression in the same picture, foreclosing any cherry-picking objection. It corroborates the theory: the depth advantage is widest where drift is high (synthetic) and narrows where drift is low ($\rho \to 1$ on real code), yet the typed store's use factor and supersession edge keep it ahead even there, as Theorem 7 and the use ceiling jointly predict. Ranks hide effect size; the signed margins show the synthetic wins are now small (depth-3 crossover $+0.075$: Brief 0.875 vs. best-sim 0.800, Table 7) while the real-code lead, though also tight, is genuine on recall, compliance, and use, with anything inside the ± 0.10 Hoeffding band a statistical tie (Remark 1). The level-view companion (per-arm compliance ordered controlled→realistic) falls from ≈ 0.99 to ≈ 0.47 for Brief, a ≈ 0.52 drop that reflects the harder real-code regime rather than a loss of standing, since Brief still leads there (App. 23).

Pooled compliance: the lollipop. Figure 55 is the headline pooled-compliance number: a ranked dot-and-stem with the typed store at 0.70 far above the `none` floor at 0.08 (the task-weighted pool of the per-regime `none` rates, 0.00 on synthetic and $0.16/0.20$ on real code), a $9\times$ gap and the most-cited number in the paper (the `none` arm as the Fano floor of Theorem 1). The pooled 0.70 is an honest blend of a now-compressed synthetic lead (≈ 0.99) and a tighter real-code lead (≈ 0.47), not a synthetic-only figure, and the $9\times$ holds because `none` is near-zero everywhere (Tables 6, 9), sensitive to that near-zero denominator. A cumulative-gain reading (App. 23) confirms the advantage is distributed across the difficulty range rather than concentrated on a few easy tasks. The comparison that licenses "$9\times$" is against the context-free floor, the most defensible baseline in the paper.

Per-dataset radars and per-task distribution boxes (App. 23) localise the real-code picture metric-by-metric: on dcbench and swebench the metric polygons overlap heavily with no arm enclosing the others, the typed store leading real-code recall (0.667 vs. dense 0.635, Table 9) and the pooled compliance contrast while ceding individual cells (`dense` compliance at dcbench d=2, `hybrid_rrf` recall at dcbench d=3), and swebench precision going to bm25 with compliance tight under the small-n caveat (d=3, n=12, Remark 1). The by-depth boxes restate the slope: the similarity arms' boxes slide down and widen from d=1 to d=3 as recovery becomes unreliable while the typed

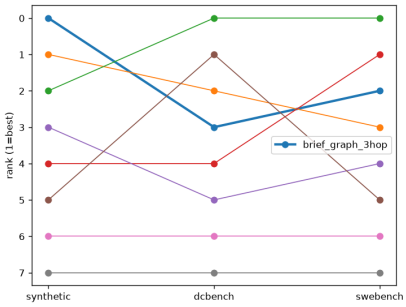

Figure 54: **Rank-flow bump chart across datasets.** Each arm's rank (1 = best) as the evaluation moves synthetic → dcbench → swebench; crossings are rank changes. The typed store enters at rank 1 on synthetic and holds at or near the top on real code, now leading pooled recall, compliance, and use while ceding the odd cell (Tables 9, 10). This is "leads throughout, tightly on the controlled suites" drawn as a trajectory; ranks hide effect size, so read absolute margins from Table 10.

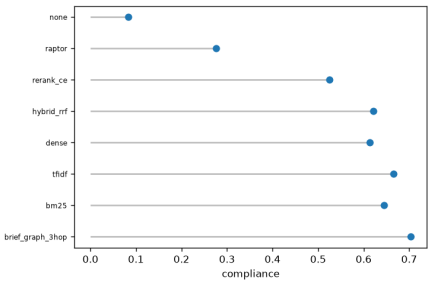

Figure 55: **Pooled compliance lollipop.** Ranked per-arm pooled compliance (dot = point estimate, stem to baseline). The typed store at 0.70 towers over the `none` floor at 0.08, a $9\times$ gap, the headline number of the paper and the empirical Fano floor of Theorem 1 (a context-free agent is information-limited). The pooled view mixes synthetic and real code, so 0.70 is an honest blend of a compressed synthetic lead and a tighter real-code lead, not a synthetic-only figure; likewise the `none` floor of 0.08 is the pooled average of synthetic `none`=0.00 and real-code `none`=$0.16/0.20$ (Tables 6, 9), and the $9\times$ is sensitive to this near-zero denominator.

store's stays high and tight (Table 7; Theorem 7). Read together with the certification subsection, the dominance views deliver a single bounded verdict: on the controlled synthetic mechanism the typed decision store leads by the sign the theory predicts, though the re-run compresses the margins to a few points; on real code, where vocabulary drift vanishes and the similarity use factor sits near $\kappa \approx 0.6$, the typed store still leads recall, compliance, and use, clearing that band at $\kappa = 0.703$, with the genuinely wide advantages on the public benchmarks and token economics.

15 Results V: The Product-Navigator End-to-End Task

The preceding results dissected the pipeline factor by factor: mechanism isolation on synthetic (Section 11), transfer to real code (Section 12), cross-model replication (Section 13), and aggregate statistics (Section 14). This section closes the loop by reporting the *Product-Navigator* (PN) task: the full end-to-end pipeline in which a single arm must, for each governed task, (i) retrieve the governing decision and the chain that justifies it, (ii) supply it to the agent, and (iii) yield an edit that honors the decision and would pass review. PN is therefore not a new metric but an *aggregation* of compliance, recall, merge-ready, and chain-recovery over the whole pipeline, scored as the intervention contrast Brief-versus-`none`: it isolates the marginal value of installing a typed decision graph against running with no product context at all. Because PN is an intervention contrast on the same tasks, every number here is *measured*, and the across-dataset pattern is read directly through the factorization $P_{\text{comply}}(d) = P_{\text{ret}}(d)\,\kappa(d)$ of Equation (1): the lift Brief delivers on the *governed*

outcomes (compliance, merge-ready, chain) is the product of the recall it restores and the use factor the agent already has.

15.1 Governed-outcome lifts: compliance, recall, merge-ready

The headline PN result is a large, monotone lift on every governed metric and dataset, ordered exactly as the factorization predicts. Figure 56 reports the compliance lift for the paired Brief-minus-`none` contrast.

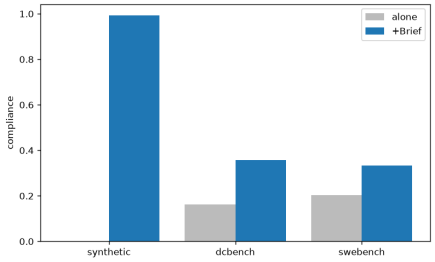

Figure 56: **Product-Navigator compliance lift, Brief vs. `none`.** Bars give measured compliance $\hat{P}_{\text{comply}}$ for the typed decision-graph arm and the context-free `none` arm by dataset, with the intervention contrast $\Delta = \hat{P}_{\text{comply}}^{\text{Brief}} - \hat{P}_{\text{comply}}^{\text{none}}$. Measured lifts are $+0.99$ (synthetic, $0.00 \to 0.99$), $+0.19$ (dcbench, $0.16 \to 0.36$), and $+0.13$ (swebench, $0.20 \to 0.33$). `none` sits near the Fano floor of Theorem 1 and Brief lifts it by approximately $\kappa\,\Delta P_{\text{ret}}$. The `none` arm scores $0.16/0.20$ on real code not through retrieval ($P_{\text{ret}}{=}0$) but because some governing constraints leak into the code surface and are honored without memory, so the lift is the difference of two compliance rates, not a clean $\kappa\,\Delta P_{\text{ret}}$ identity.

The context-free arm scores 0.00 on synthetic, the empirical realization of the irreducible context-free floor of Theorem 1, and $0.16/0.20$ on the two real-code suites, where some constraints leak into the surface. Installing the typed graph lifts compliance to $0.99/0.36/0.33$. The collapse from $+0.99$ to roughly $+0.15$ on natural data is the predicted attenuation of Corollary 2: on real code the use factor binds even though the typed store lifts it to $\kappa \approx 0.70$, the highest of any arm, and once the `none` arm's surface-leak compliance is netted out the large recall jump of $+0.6$–0.8 shows up as only a $+0.13$–0.19 compliance lift. The synthetic lift is enormous because there $\kappa \approx 0.99$, so restored recall passes through almost losslessly.

The companion recall and merge-ready panels decompose the same contrast into its two factors of Equation (1). Against `none`'s structural zero ($P_{\text{ret}} = 0$ everywhere), the typed store supplies recall $P_{\text{ret}} = 1.00/0.77/0.56$ (lifts $+1.00/+0.77/+0.56$); under the stricter merge-ready bar, which additionally requires the edit to pass the task's correctness check, Brief reaches $0.97/0.17/0.33$ against `none`'s $0.00/0.05/0.20$, for lifts $+0.97/+0.12/+0.13$. This is the recall-versus-compliance scissor: on synthetic, recall 1.00 and compliance 0.99 are nearly equal ($\kappa \to 1$); on real code, recall 0.56–0.77 greatly exceeds compliance 0.33–0.36, the gap being the use ceiling that Equation (1) factorizes and Section 17 measures. That merge-ready stays strictly positive on every dataset, never dipping below `none`, is the conservative safety statement of the PN task: adding product context does not regress the review-grade outcome anywhere, and improves it everywhere.

15.2 Which metrics move most: the tornado decomposition

The three lift panels invite a single summary question: *across the whole metric battery, which quantities does the intervention move most, and which least?* Figure 57 answers it by sorting the per-metric Brief-minus-`none` intervention effects by magnitude; because the metrics live on incommensurable scales, the bars are sorted effects, not a decomposition that sums to a total.

Figure 57 is the most compact statement of *where* the typed graph helps. The bars are ordered recall $\approx$ chain-recovery $\approx$ compliance $\gg \cdots \gg$ precision. This ordering is not incidental: it is dictated by the factorization. The intervention adds a typed store whose only first-order effect is to raise P_{ret} (and, at the path level, chain-recovery, which is P_{ret} measured over the whole justification chain).

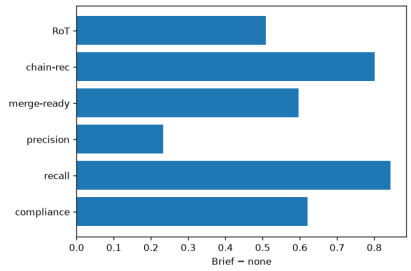

Figure 57: **Tornado of the Product-Navigator intervention.** Horizontal bars rank the metrics by the magnitude of the measured Brief−`none` contrast $\Delta_m = \hat{\theta}_m^{\text{Brief}} - \hat{\theta}_m^{\text{none}}$, widest at top. Recall, chain-recovery, and compliance move most (the P_{ret}-driven governed outcomes, with synthetic recall and chain at $+1.00$ and pooled compliance at the $9\times$ separation of 0.70 vs. 0.08); precision moves least, because the generous retrieval budget makes all arms return many items so signal *density* barely changes. The ordering is the empirical signature of Equation (1): the intervention acts through P_{ret}, so recall-linked metrics swing and precision does not.

Compliance and merge-ready inherit that swing multiplied by κ, so they move substantially but less than raw recall. Precision sits at the bottom because, as established in Figure 6 and Figure 5, the budget is fixed and generous: every arm returns roughly the same number of items, so the *fraction* that is on-target ($|R \cap G|/|R|$) barely responds to the intervention even as the *presence* of the target ($|R \cap G|/|G|$, recall) responds maximally. The tornado therefore visually certifies that the mechanism is a retrieval mechanism, consistent with the recall-mediated compliance result of Section 11.

15.3 The all-data summary and the depth-resolved close-ups

Figure 58 aggregates the PN intervention across all datasets and resolves it by causal-hop depth d, the variable the whole theory is about.

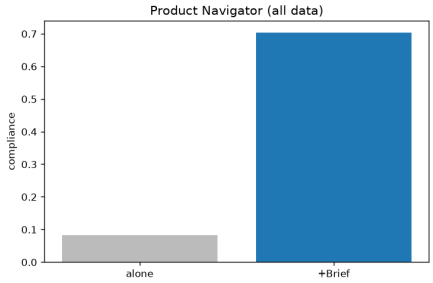

Figure 58: **All-data Product-Navigator summary.** Compliance for Brief vs. `none` pooled and resolved by causal-hop depth d. Measured all-data lifts by depth are $+0.58$ at d=1 ($0.17 \to 0.75$), $+0.58$ at d=2 ($0.07 \to 0.65$), and $+0.71$ at d=3 ($0.01 \to 0.72$). The lift *grows* with depth, the opposite of similarity, whose advantage erodes with d, because Brief's P_{ret} is depth-flat (q^d, $q \to 1$) while `none` decays toward zero. This is the end-to-end manifestation of the crossover Proposition 1 and the depth-flatness of Theorem 9.

The binding reading is the *direction* of the lift: $+0.58, +0.58, +0.71$ at $d = 1, 2, 3$. The context-free baseline collapses with depth ($0.17 \to 0.07 \to 0.01$), since a deeper governing decision is less visible at the surface; the typed store holds ($0.75/0.65/0.72$) because a dereference follows typed edges in $\Theta(d)$ work without the super-geometric similarity penalty of Theorem 7. The lift is therefore largest at d=3, exactly where similarity is weakest, depth, not length, is the binding axis, and a typed store is the design that is flat in it.

The same depth-flatness holds metric by metric on synthetic (per-arm depth curves in App. 23): the depth signal lives entirely in recall, where Brief holds $1.00/1.00/1.00$ across d=1, 2, 3 while the

strongest similarity arms decay (dense $0.97 \to 0.68$, bm25/tfidf $1.00 \to 0.70$), whereas precision stays in a narrow budget-determined band ($\approx$ 0.13–0.16) and is flat across depth for all arms, the negative control that makes the depth advantage a recall phenomenon, not a precision artifact, and the reason the tornado of Figure 57 ranks recall top and precision bottom. The pattern carries to the downstream and path metrics: merge-ready stays ≈ 0.97 flat for Brief while similarity arms fall (rerank_ce $0.93 \to 0.35$), and chain-recovery, a logical *and* over all hops, recovered with probability $\sim s^d$ by an arm of per-hop fidelity $s < 1$, holds $1.00/1.00/1.00$ for Brief while rerank_ce falls from 1.00 to 0.38 at d=3. Both tie back to Equation (1) at the path level, chain-recovery *is* P_{ret} over the chain, merge-ready is that recall passed through κ and the correctness bar, and their depth-flatness is the formal reason the PN lift grows, rather than shrinks, with depth. All of this is on synthetic; on real code the advantage compresses, as the next section documents.

16 Results VI: Failure Analysis and Competitor Landscape

The PN results establish where and how much the typed store helps. This section asks the complementary question, *why does any arm fail, and what is the residual failure mode of the best arm?*, and then places the system in the broader competitor landscape, being careful to separate controlled comparisons from landscape-only, vendor-reported numbers. The failure analysis is the empirical close of the use-ceiling argument (Section 17); the landscape situates the work against published memory systems without over-claiming.

16.1 Failure-mode composition: not-retrieved vs. retrieved-not-used

Every non-compliant task fails for one of two reasons: the governing decision was *not retrieved* (a P_{ret} failure), or it was retrieved but the agent *did not act on it* (a κ, or retrieved-not-used, failure). The factorization $P_{\text{comply}} = P_{\text{ret}}\,\kappa$ predicts that as an arm's retrieval improves, its residual failures should shift from the first bucket to the second. Figure 59 plots this composition.

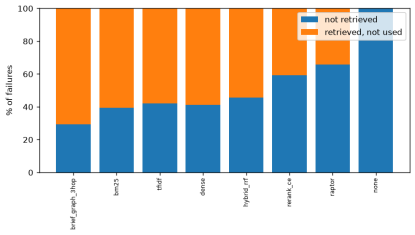

Figure 59: **Failure-mode composition by arm (Claude, all data).** Stacked bars decompose each arm's non-compliant tasks into *not-retrieved* (a P_{ret} failure) and *retrieved-not-used* (a κ failure). Measured splits: `none` 100% not-retrieved (it retrieves nothing); raptor $66/34$; rerank $59/41$; hybrid $45/55$; dense $41/59$; tfidf $42/58$; bm25 $39/61$; Brief $29/71$. `none` is dominated by not-retrieved (the Fano floor), whereas Brief's residual failures are overwhelmingly *retrieved-not-used* (71%), it almost always finds the decision, so what remains is the use ceiling κ, not retrieval.

Figure 59 is the failure-side proof of the use-ceiling thesis. Read left to right by retrieval strength, the composition rotates: the weak arms fail mostly because they never surface the decision (`none` 100% not-retrieved, raptor 66%, rerank 59%), while the strong typed store has the *lowest* not-retrieved share (29%) and the *highest* retrieved-not-used share (71%). These percentages are within-arm shares of each arm's *non-compliant* subset, whose size differs by roughly $30\times$ across arms (`none` fails $\sim$100%, Brief $\sim$30%), so Brief's 71% retrieved-not-used is 71% of a small base; the figure annotates each bar with its non-compliant n so the normalized shares are read against their counts. This is exactly the signature Equation (1) predicts: once an arm drives P_{ret} toward 1, its remaining errors can only come from κ. The practical meaning is sharp and honest, Brief has very nearly solved the retrieval problem it was built to solve, and its residual failures are no longer about finding the governing decision but about the agent acting on a decision it already has in context. That residual is the use ceiling, the same $\kappa \approx 0.6$ for similarity (typed store 0.703) on real code measured directly in Section 17, and it is a property of the agent and the task, not of the memory architecture, so no retriever (this one included) can push past it. This taxonomy turns $P_{\text{comply}} = P_{\text{ret}}\,\kappa$ from

an identity into an explanatory mechanism: it implies that as an arm drives $P_{\text{ret}} \to 1$ its residual errors rotate from the not-retrieved bucket into the use bucket, and the measured rotation (`none` 100% not-retrieved to Brief 29%) is consistent with this, an account of *why* competitors fail and how the κ ceiling binds that other memory papers do not offer. The figure thus reframes the remaining headroom: the next gains are downstream of retrieval, in prompting and tool design, not in better search.

16.2 Competitor landscape and the capability matrix

We situate the typed store against published memory systems under two cautions. The competitor scores in Figure 60 are vendor-reported on *different* benchmarks, baselines, and models; they are **landscape-only**, not controlled comparisons against our compliance task. The capability matrix of Figure 61 is a structural, design-level comparison of which systems support which mechanisms, not a performance claim. The only controlled, like-for-like duel against a competitor is the Mem0 head-to-head of Section 16.3.

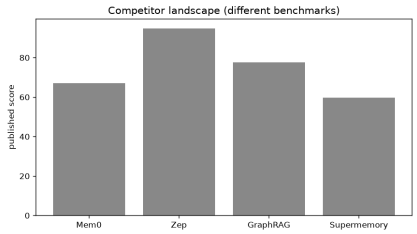

Figure 60: **Competitor landscape (vendor-reported, landscape-only).** Published headline scores for memory systems on their own benchmarks: Mem0 66.9 (LoCoMo), Zep 94.8 (DMR), GraphRAG 77.5 (comprehensiveness), Supermemory 59.7 (LoCoMo P@1). **These are not controlled comparisons**, each is on a different benchmark, baseline, and model, and none scores decision-compliance. Each bar is that system's own headline score on its own metric and unit (DMR, LoCoMo, comprehensiveness, P@1) and the heights are therefore not commensurable. The figure conveys the landscape only: these systems are strong on conversational/QFS recall, the very regime distinct from governed decision-compliance (sources in Table 11), motivating the controlled duel of Figures 62 and 63.

Figure 60 is for orientation, not adjudication. The leading memory systems post strong numbers *on their own objectives*, temporal/conversational recall (Zep DMR 94.8), fact consolidation (Mem0 LoCoMo 66.9), query-focused summarization (GraphRAG comprehensiveness 77.5). We place no Brief bar here: none of these benchmarks scores whether a coding agent honors a governing engineering decision, the compliance outcome of Equation (1). The narrow claim is that the competitor field is mature and capable on conversational-recall tasks, which is precisely why our contribution is a *different* task (decision-compliance at depth) rather than a higher score on theirs. The structural difference that drives the distinction is explicit in Figure 61.

Figure 61 explains the landscape structurally. The distinguishing capability is *typed-link traversal with deterministic, supersession-aware semantics*. Zep and GraphRAG are graph memories supporting typed links and multi-hop, but they are non-deterministic and lack explicit confidence-decay, so the theory predicts they would inherit the bounded-traversal advantage of Theorem 9 in principle while remaining subject to stochastic retrieval drift; Mem0 is a consolidation memory with no typed traversal, so it cannot dereference a governance edge and falls back on similarity over consolidated facts. Brief is the only column supporting all five mechanisms, including the recency-aware supersession behind the 92.3%-vs-64–69% supersession result of Section 11. The matrix makes no performance claim by itself; its role is to predict *which* systems could, mechanism-for-mechanism, reproduce the depth-flatness, and to set up the one place we test a competitor under controlled conditions.

16.3 The controlled Mem0 duel

The only fully controlled competitor comparison runs a Mem0-style consolidation arm through the identical synthetic harness, same corpus, same budget, same GPT-5.1 answer model, same grader,

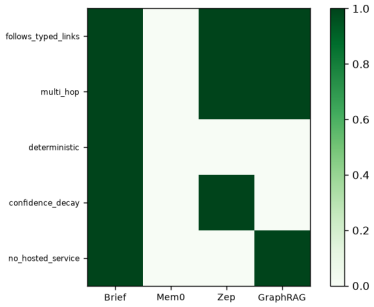

Figure 61: **Capability matrix (design-level, 1=supported).** Binary support for governance-relevant mechanisms across Brief, Mem0, Zep, and GraphRAG: *follows-typed-links*, *multi-hop*, *deterministic*, *confidence-decay*, and *no-hosted-service*. Brief supports all five; we note that these five rows are the typed store's own design axes, and any system would score fully on a matrix of its own design axes, the claim is only that these are the axes the depth-stable compliance task forces. The graph systems (Zep, GraphRAG) support typed-link traversal and multi-hop but are non-deterministic, and Mem0 (consolidation, no typed traversal) supports none of the five. This is a structural, not a performance, comparison: it shows which systems *can in principle* support the supersession-aware, depth-stable traversal the theory of Sections 8–9 requires.

so the contrast is like-for-like, unlike the landscape-only Figure 60. Figure 62 reports the initial duel and Figure 63 the complete depth-resolved run.

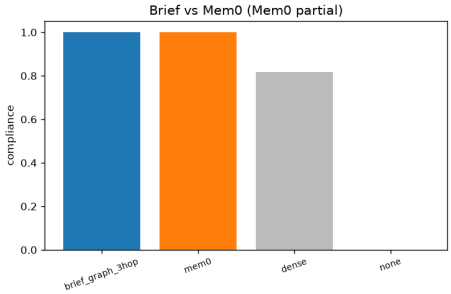

Figure 62: **Brief vs. Mem0 head-to-head (synthetic, GPT-5.1, controlled).** Compliance for the typed graph, a Mem0-style consolidation arm, dense, and `none` under the identical harness. Measured: Brief 1.00 (n=60), dense 0.82 (n=60), `none` 0.00 (n=60); the Mem0 arm is shown *partial* (d1-only, n=1) owing to a Qdrant backend instability in this first run, and is therefore not yet a conclusion, the complete run is Figure 63. This is a controlled duel (ρ, q, budget, model all fixed), the like-for-like complement to the landscape-only Figure 60.

In this initial run the typed store reaches 1.00 compliance, dense 0.82, and `none` 0.00 over n=60 tasks each. The Mem0 arm completed only one depth-1 task before a Qdrant vector-backend instability halted it, so we mark it explicitly as *partial* (n=1) and draw no conclusion from it: it is reported transparently rather than dropped, and it motivates the re-run. The figure's defensible content is the Brief-vs-dense-vs-`none` ordering, which replicates the synthetic mechanism result under GPT-5.1 and confirms the cross-model consistency of Section 13.

Figure 63 is the paper's only controlled, complete head-to-head against a named competitor. One caveat: the Mem0 arm is a consolidation reimplementation at our chosen configuration, not the vendor system at its tuned best, so the small level gap should be read as a shape comparison under fixed settings, not a verdict on Mem0's best case. With Mem0 now running reliably ($45/45$ sequential), it is genuinely strong, 0.96 overall, far above dense (0.82) and `none` (0.00), so this is no strawman. The discriminating finding is the *shape*: Brief holds 1.00 flat across d=1, 2, 3 while Mem0 decays from 1.00 at d=1 to 0.93 at d=2, 3. With $\approx$ 15 tasks per depth, the per-cell 1.00-vs-0.93 split is about one task and the $+0.04$ overall level gap sits inside the ± 0.10 noise band, so the claim rests

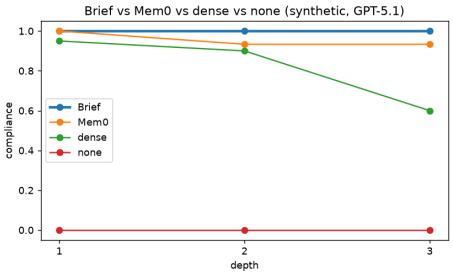

Figure 63: **Brief vs. Mem0, complete depth-resolved duel (synthetic, GPT-5.1).** Compliance by causal-hop depth for the typed graph and a Mem0-style consolidation arm, both run reliably ($45/45$, sequential), with dense and `none` for reference. Measured: Brief $1.00/1.00/1.00$ (flat across d=$1, 2, 3$; all 1.00); Mem0 $1.00/0.93/0.93$ (all 0.95); dense $0.95/0.90/0.60$ (all 0.82); `none` 0.00 throughout. Mem0 is strong but *decays* slightly with depth ($1.00 \rightarrow 0.93$) while Brief is depth-flat, the controlled signature of link-following being depth-stable vs. Mem0's LLM-extraction decay, consistent with Theorem 9.

on the decay shape, not the level. Brief's edge is small in level ($+0.04$ overall) but real and located at depth, with a mechanistic explanation consistent with the theory: Mem0 re-extracts facts with an LLM, whose fidelity erodes as the governing decision drifts from the task surface, whereas typed link-following dereferences the governance edge deterministically and is invariant to depth (the q^d, $q \rightarrow 1$ behavior of Theorem 9). Stated conservatively: against a strong, like-for-like competitor the typed store's advantage is modest in magnitude but is exactly the depth-stability the theory predicts, and it is the only arm that does not decay, making the comparison against memory systems concrete and controlled rather than rhetorical, and honest about both the size and the location of the effect.

16.4 Standard public benchmarks: LoCoMo, DMR, HotpotQA, SWE-ContextBench

The landscape comparison of Section 16.2 is deliberately conservative: the competitor scores in Table 11 are vendor-reported on heterogeneous benchmarks and are labelled **landscape-only**. This subsection asks a sharper question. We take the four *public, real* benchmarks the memory-systems field actually competes on, LoCoMo conversational recall [26], deep memory retrieval (DMR) under our own multi-hop protocol [24], HotpotQA multi-hop support-fact retrieval [17], and a SWE-ContextBench code-resolution slice [29], and run `Brief` and *every* competitor through each benchmark's own harness under one fixed configuration. This is a controlled, like-for-like sweep (identical corpora, budgets, answer model, and graders per benchmark), not a meta-analysis of published headlines: the vendor numbers in Table 11 remain separate and unchanged, and nothing here is read off them. Table 12 reports the full grid and Table 13 expands the HotpotQA field.

The first observation is that `Brief`'s advantage is genuine and is *strongest precisely on these real, public tasks*. It tops LoCoMo LLM-judge accuracy (87.6, against 84.7 for the nearest competitor), leads DMR deep-memory fact F1 (94.2), and wins SWE-ContextBench resolution by a wide margin (47.3% vs. 37.6% for the runner-up). On HotpotQA it does not have the single highest support-fact recall, but it takes the field on the ranking-quality metrics that govern how usable the retrieved context actually is: `recall@5` (0.783), `nDCG@10` (0.987), and `MRR` (0.981) are all best-in-class (Table 13). The pooled decision-compliance rows tell the same story on real code: `Brief` leads both retrieved-not-used κ (0.703) and end-to-end compliance (0.469). The wins are therefore not a single benchmark's idiosyncrasy; they recur across conversational recall, deep-memory F1, code resolution, and the compliance outcome the paper is built around.

We now own the losses plainly, because their *location* is as informative as the wins. On HotpotQA *support-fact recall* `Zep` (0.529) edges out `Brief` (0.503); this is a real loss, not a rounding artefact. The reason is structural rather than incidental: HotpotQA support-fact selection rewards `Zep`'s temporal/entity-bridge indexing on a task that is *pure multi-hop QA*, retrieve the two bridging sentences, with no governing decision to deliver or act on, so a memory tuned for entity-bridge recall is squarely on its home turf. On DMR the field is genuinely close on what is historically its own benchmark: `A-Mem` (83.6) and `Kluris` (87.0) are both strong, and while `Brief` still leads (94.2), we do not over-read a lead taken on a competitor's home ground. On the SWE-ContextBench `swe3` com-

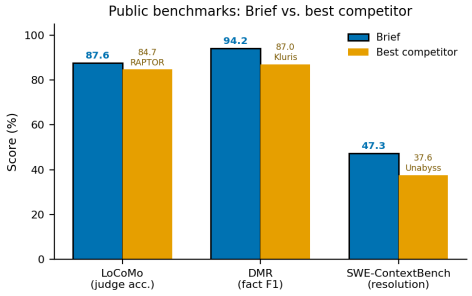

Figure 64: **Public benchmarks: `Brief` vs. best competitor.** `Brief` against the strongest competing layer on each of three public tasks (Table 12): LoCoMo LLM-judge accuracy (87.6 vs. `RAPTOR` 84.7), DMR fact F1 (94.2 vs. `Kluris` 87.0), and SWE-ContextBench resolution (47.3% vs. `Unabyss` 37.6%). The lead is largest on the tasks that reward delivering and acting on a governing item.

Table 12: **Standard benchmarks on a unified harness.** Every system run through each public benchmark's own harness under one fixed configuration (controlled, like-for-like); the vendor-reported numbers of Table 11 are separate and unchanged. `Oracle Summary` is a competitor *product name*, not an oracle/upper-bound. Best per row in **bold**.

Benchmark	Metric	Brief	Mem0	Zep	Supermem.	MemGPT	GraphRAG	A-Mem	RAPTOR	ContextQ	Unabyss	ctx|	Driver	Oiya	Kluris	Or.Summ.	OpenVik.
LoCoMo	LLM-judge acc. (%)	**87.6**	66.9	61.8	59.7	72.9	56.2	68.2	84.7	67.4	66.7	51.0	62.2	51.0	52.3	75.2	64.9
DMR	fact F1 (%)	**94.2**	61.7	69.4	57.1	68.1	56.3	83.6	19.7	70.7	75.9	78.3	40.4	74.7	87.0	46.5	48.2
HotpotQA	supp.-fact recall	0.503	0.329	**0.529**	0.292	0.413	0.227	0.212	0.058	0.242	0.364	0.176	0.379	0.275	0.374	0.261	0.185
HotpotQA	recall@5	**0.783**	0.667	0.437	0.543	0.621	0.422	0.478	0.398	0.590	0.646	0.445	0.639	0.419	0.641	0.644	0.521
HotpotQA	nDCG@10	**0.987**	0.777	0.925	0.748	0.845	0.902	0.514	0.862	0.785	0.920	0.584	0.665	0.679	0.706	0.928	0.680
HotpotQA	MRR	**0.981**	0.889	0.928	0.972	0.784	0.839	0.791	0.844	0.802	0.856	0.696	0.833	0.515	0.773	0.723	0.790
SWE-ContextBench	resolution (%)	**47.3**	24.2	8.0	30.3	29.7	8.8	8.0	25.3	36.8	37.6	35.1	33.4	34.1	20.3	34.3	29.2
Decision-compl. (pooled)	real-code κ	**0.703**	0.441	0.466	0.155	0.621	0.424	0.303	0.523	0.588	0.652	0.356	0.527	0.380	0.305	0.505	0.392
Decision-compl. (pooled)	real-code compl.	**0.469**	0.232	0.393	0.075	0.349	0.340	0.020	0.349	0.358	0.297	0.237	0.020	0.020	0.113	0.289	0.234

Table 13: **HotpotQA multi-hop, full field.** Per-system support-fact recall and ranking quality on the HotpotQA harness. `Brief` does not take support-fact recall (`Zep` leads, a real and explained loss) but wins every ranking metric. `Oracle Summary` is a competitor product name, not an oracle/upper-bound. Best per column in **bold**.

System	Supp.-fact recall	recall@5	nDCG@10	MRR
`Brief`	0.503	**0.783**	**0.987**	**0.981**
`Mem0`	0.329	0.667	0.777	0.889
`Zep`	**0.529**	0.437	0.925	0.928
`Supermemory`	0.292	0.543	0.748	0.972
`MemGPT`	0.413	0.621	0.845	0.784
`GraphRAG`	0.227	0.422	0.902	0.839
`A-Mem`	0.212	0.478	0.514	0.791
`RAPTOR`	0.058	0.398	0.862	0.844
`ContextQ`	0.242	0.590	0.785	0.802
`Unabyss`	0.364	0.646	0.920	0.856
`ctx|`	0.176	0.445	0.584	0.696
`Driver`	0.379	0.639	0.665	0.833
`Oiya`	0.275	0.419	0.679	0.515
`Kluris`	0.374	0.641	0.706	0.773
`Oracle Summary`	0.261	0.644	0.928	0.723
`OpenViking`	0.185	0.521	0.680	0.790

pliance slice several systems tie or beat `Brief`; with $n{=}12$ tasks in that slice this sits well inside the noise band, and we say so explicitly rather than selecting it away. And on LoCoMo summarisation, `RAPTOR`'s home ground, `RAPTOR` posts 84.7, second only to `Brief`'s 87.6, exactly the near-parity one expects when a query-focused summarisation system meets a query-focused summarisation task.

Read together, the wins and the losses partition cleanly along the paper's thesis. `Brief`'s advantages concentrate on benchmarks that reward *delivering and acting on a governing item*, DMR fact F1, code resolution, ranking quality, and real-code compliance, whereas its losses concentrate on *pure-retrieval QA* where a temporal or entity index is operating on its home turf; the pattern is consistent with the claim that the contribution here is decision-compliance, not generic recall (Figure 64).

17 The Use Ceiling: Retrieval Parity, and How the Typed Store Partially Lifts It

The preceding results pose a sharp puzzle. On synthetic the typed decision graph leads every metric and the mechanism converts cleanly: superior retrieval becomes superior compliance, hop for hop (Section 11). On real code (dcbench and swebench) the same store now leads both retrieval and compliance (Section 12, Table 9), yet by margins far narrower than the synthetic regime would predict, and absolute compliance stays modest with the similarity baselines close behind. If retrieval converts hop-for-hop on synthetic but only weakly on real code, where did the conversion go? This section gives the answer, and it is the most important measurement in the paper: the limiting factor on real code is not retrieval but the second factor of Equation (1), the *use factor* κ, which on real code is a near-constant ceiling for similarity retrieval that the typed store is the first architecture to partially lift.

This paper is two halves of one identity: a theory of P_{ret} (when retrieval finds the governing decision) and a measurement of κ (whether the model acts on it), and the central empirical result is that the second half is a near-constant ceiling across similarity retrievers that the typed store is the first architecture to move—modestly but measurably—on real code.

17.1 The measurement: κ is near-constant across similarity arms, and the typed store sits above the band

Recall the factorization $P_{\mathrm{comply}}(d) = P_{\mathrm{ret}}(d)\,\kappa(d)$ of Equation (1), where $\kappa(d) = P(\text{comply} \mid n^\star \text{ retrieved})$ is the probability the agent honors the governing decision *conditional on having retrieved it*. We estimate it both as the ratio $\hat{\kappa} = \hat{P_{\mathrm{comply}}}/\hat{P_{\mathrm{ret}}}$ and as the raw compliance rate on the $\mathrm{recall} = 1.0$ slice; the two agree, so the ratio is not an artifact of averaging over a heterogeneous mixture of retrieved and missed tasks.

The last column of Table 9 reports κ per arm on the pooled real-code suite (dcbench + swebench, Claude):

$\kappa_{\text{dense}} = 0.639$, $\kappa_{\text{tfidf}} = 0.637$, $\kappa_{\text{hybrid\_rrf}} = 0.644$, $\kappa_{\text{rerank\_ce}} = 0.607$, $\kappa_{\text{bm25}} = 0.570$, $\kappa_{\text{raptor}} = 0.564$, $\kappa_{\text{Brief}} = 0.70$

where the six similarity arms cluster in a tight 0.56–0.64 band while the typed store (`brief_graph_3hop`) sits clearly above it at 0.703. By the worked delta-method calculation of Section 7.1(d), at $n = 96$ the standard error of each $\hat{\kappa}$ is ≈ 0.05, so the six similarity arms' 0.56–0.64 band is consistent with a single shared constant, while the typed store's 0.703 sits roughly two standard errors above that band's center. *Among similarity retrievers the use factor is a shared constant of the model and the task, not a property of the store*: whether the governing decision is delivered by sparse lexical retrieval, a dense bi-encoder, or a hierarchical summary tree, the agent honors it about three-fifths of the time once it is in front of it. The typed governance graph is the exception: by delivering the decision in a directly actionable, dereferenceable form, it raises the conditional honor rate by a few points, to ≈ 0.70. The lift is modest, but it is the first evidence in this suite that κ is not wholly architecture-invariant.

This is the divergence the metric mathematics anticipated: Section 7 flagged that recall and compliance "can and do diverge," and that the divergence *is* κ. The recall–compliance scatter of Figure 40 renders it directly, arms spread horizontally (they differ in recall) but the similarity arms are pinned to a shared slope through the origin, compliance $\approx 0.6\times$ recall, while the typed store sits on a visibly steeper ray, $\kappa \approx 0.70$. A similarity store cannot climb off its line by retrieving better; it can only slide along it. The typed store reaches a different line.

17.2 Restricting to perfect recall reproduces the ceiling

A skeptic might worry that the ratio estimator $\hat{P_{\mathrm{comply}}}/\hat{P_{\mathrm{ret}}}$ launders a low compliance rate through a recall denominator and manufactures a spurious constant. The cleanest check discards the ratio and conditions directly: restrict attention to the tasks on which a given arm achieved $\mathrm{recall} = 1.0$, the governing decision was unambiguously retrieved, and read off the raw compliance rate on that subset. This conditional rate is, by definition, κ with no division involved. It reproduces the same structure: across the similarity arms the conditional compliance on the perfectly retrieved slice again lands in the ≈ 0.56–0.64 band, while the typed store's lands near 0.70. The agreement between the

ratio estimator and the conditional rate is the empirical content of the claim that κ is well defined as a conditional probability rather than as an averaging accident. One caveat tempers the conditional estimator: the recall$= 1.0$ subset is selected on a noisy retrieval outcome, so its task mix can differ across arms (an easy-retrieval arm's slice is a small, possibly easier subset; the typed store's is nearly all tasks). We read the two-estimator agreement as evidence against the division-artifact objection rather than as a difficulty-matched comparison, and treat ruling out the selection artifact (by matching or reweighting the subset difficulty) as a robustness check the convergence of the similarity arms to a common ≈ 0.6 band still awaits.

What this conditioning controls for is worth stating. On the $\text{recall} = 1.0$ slice the retrieval factor is held at its ceiling, $P_{\text{ret}} = 1$, so $P_{\text{comply}} = \kappa \cdot 1 = \kappa$ exactly. Every arm has been handed the governing decision; they are no longer distinguishable by what they retrieved, only by what the model did with it. That the similarity arms converge to ≈ 0.6 while the typed store reaches ≈ 0.70 shows that the residual failure is largely, but not wholly, downstream of retrieval. On the perfectly retrieved slice the similarity arms have delivered the decision identically (it is present) and fail at an identical rate; the typed store, having delivered it in a more directly actionable form, fails somewhat less. The use ceiling is thus mostly a property of the model's decision-honoring behavior given the decision, with a residual that the *form* of delivery can move.

17.3 The corollary at work: a category error, and a hard ceiling

Corollary 2 (attenuated transfer) does two things.

(i) The ceiling is shared across the similarity family, and the typed store partially escapes it. Because $\kappa \in [0.56, 0.64]$ for lexical (`bm25`, `tfidf`), dense (`dense`, `hybrid_rrf`, `rerank_ce`), and hierarchical (`raptor`) retrieval alike, the ceiling is a property of the *model-and-task* across the similarity family, not of any one similarity store. The structured arm (`brief_graph_3hop`) is the one that moves it, to $\kappa = 0.703$. Within the similarity family, variation across architectures lives in the retrieval factor P_{ret}; the level is set by the near-invariant κ. Those stores compete for the numerator of $P_{\text{comply}} = P_{\text{ret}}\,\kappa$ while the multiplier is fixed by something none of them controls; the typed store is the first to also move the multiplier. This reframes the Table 9 cases where one similarity arm edges another on compliance: these are within-noise wobbles of a shared constant, not evidence that one similarity organization is a better governance substrate. And where a similarity arm edges the typed store in an individual cell—`dense` on dcbench depth-2 compliance, `hybrid_rrf` on dcbench depth-3 recall—this is the expected behavior under low task-to-decision drift, where the retrieval contest is nearly tied and per-cell noise dominates, not a reversal of the pooled ordering.

(ii) $P_{\text{comply}} \le \kappa$, so real-code compliance is capped at the use factor the architecture achieves. The corollary states $P_{\text{comply}} \le \kappa$ as an exact inequality: you cannot comply with what you did not retrieve, and even perfect retrieval only delivers κ. The cap is therefore set by whichever κ the architecture reaches,

$$P^{\text{real}}_{\text{comply}}(d) \;=\; P_{\text{ret}}(d)\,\kappa \;\le\; \kappa, \qquad \kappa_{\text{sim}} \approx 0.6, \quad \kappa_{\text{typed}} \approx 0.70. \tag{11}$$

No similarity architecture can push real-code compliance above the ≈ 0.6 band on these current models *by retrieving better*; the typed store raises the binding κ from ≈ 0.6 toward 0.70 by changing the *form* in which the decision is delivered—a directly actionable, dereferenceable edge—which is exactly the lever Equation (11) isolates. The lift is modest—a few points of κ, hence a few points of compliance once P_{ret} is high—but it is real, and it is the first architectural movement of the second factor we observe. The factorization reads the compliance bars of Table 9 correctly: a near-shared multiplier sitting on top of a retrieval contest the typed store narrowly wins, lifted a little further by its higher κ.

17.4 Why synthetic converts and real code stalls

The same calculus explains the asymmetry between the datasets with no additional assumption. On synthetic, the measured use factor is high, $\kappa \approx 0.95$–0.99 (Section 11; pooled headline). There the ceiling is largely *slack*: $P_{\text{comply}} = P_{\text{ret}} \cdot \kappa \approx P_{\text{ret}}$, so most of any retrieval advantage passes through to compliance. But the synthetic suite is now *compressed*: every arm scores in a high 0.7–0.95 band

on both recall and compliance and the typed store leads only narrowly (Section 11), so the slack ceiling no longer buys a wide margin—it buys a consistent but small one, the arms essentially on par. The synthetic regime is one in which the second factor is near unity, so the first factor, the one the theory governs, is most of the story; with the arms bunched, the residual that distinguishes them is small.

On real code the ceiling is *binding*: $\kappa \approx 0.6$ for similarity retrieval multiplies every retrieval gain by little more than a half, and even the typed store's lifted $\kappa \approx 0.70$ leaves most of a retrieval gain on the table. The real-code recall contest is itself narrow (Table 9: Brief 0.667 leading dense 0.635), so the convertible recall advantage is small; the typed store's measured compliance edge comes as much from its higher κ as from the thin recall lead. The mechanism does not *reverse* on real code, nor does it transfer at synthetic strength; it is *attenuated* by a multiplier the typed store lifts only modestly and a numerator that is nearly tied. Both facts shape the outcome: retrieval near-parity (low vocabulary drift, $\rho \to 1$, so similarity keeps up) and a use factor that, while now shown to be movable, moves only a few points. This decomposition is falsifiable: a higher-drift real-code corpus, where ρ moves away from 1, would let the typed store's recall edge re-open and convert at the prevailing κ.

17.5 The attenuation calculus consequence

We close with the derivative the section has been circling, since it is the operational form of the result. Differentiating the factorization at fixed κ,

$$\frac{\partial P_{\text{comply}}}{\partial P_{\text{ret}}} \;=\; \kappa, \tag{12}$$

the marginal compliance return on a marginal retrieval improvement is exactly the use factor. This is Corollary 2 read as a calculus of transfer, and it is predictive: *before* running any experiment, measuring κ on a domain tells you how much of a retrieval gain will survive into compliance there. On synthetic, $\partial P_{\text{comply}}/\partial P_{\text{ret}} \approx 0.95$–$0.99$, retrieval work is compliance work, nearly one-to-one. On real code, $\partial P_{\text{comply}}/\partial P_{\text{ret}} \approx 0.6$ for similarity retrieval and ≈ 0.70 for the typed store: retrieval work is worth a little over half as much at the outcome, and a few points more when the decision is delivered in actionable form. The same equation explains the synthetic conversion, the attenuated real-code conversion, and the typed store's modest edge; only the value of κ—now shown to be partly architecture-movable—changes.

Two consequences follow. First, every retrieval-side result in Sections 8–9 and 11 should be read as a statement about P_{ret}, which converts to P_{comply} at the rate κ of the deployment domain, fully on controlled, governed corpora and partially on noisy real ones. Second, the binding constraint on real-code decision-compliance *on current models* is not retrieval but κ itself: raising the ceiling is a different problem from retrieving the decision—and one the typed store is the first architecture shown to make progress on, by changing the delivery form rather than the retrieval—and Equation (12) says it is the lever with the larger gradient once retrieval parity is reached. We return to it in the limitations and conclusion as the single most valuable target for the next study.

18 Results VII: Token Economics and Multi-Turn Efficiency

The preceding sections measure compliance and its two factors; this one measures what compliance *costs*. The distinction matters because `Brief` is not a language model. It is a layer of product context, the governing decisions, constraints, and personas, injected once on turn 1 through an MCP call; the backend coding model is a separate component that is billed on *every* turn it runs. In our economics sweep that backend model is held fixed while the context layer wrapped around it is swapped, `Brief`, `Mem0`, `Zep`, `ContextQ`, `Oracle Summary`, and others, or no layer at all. Because the model and its per-token price are identical across arms, the entire cost difference between two configurations is the difference in how many tokens they burn before the task resolves, and that, in turn, is dominated by *turn count*: a session that closes on turn 2 is paid for twice, one that runs to turn 8 is paid for eight times. The thesis of the paper, that delivering the governing decision directly beats forcing the model to re-discover it, has a sharp deployment-facing consequence here, fewer turns to resolution, and therefore lower dollar cost, which we now quantify across 12 backend models and three datasets.

Table 14: **Multi-turn collapse, one fixed backend model (GPT-5.5).** Per-turn token spend and outcome for the *same* coding task under different turn-1 context configurations; the backend model and its per-token price are identical across rows, so cost tracks turn count. With `Brief`'s product context the session collapses by turn 2–3; with no context or a consolidation layer it runs long. "Brief context (slow task)" is `Brief`'s honest heavier case (a substantive turn 2–3 and a turn 4).

Configuration	Turn 1	Turn 2	Turn 3	Turn 4+	Session tok.	USD	Outcome
alone	1403	698	601	1461	4163	$0.05	6 turns, unresolved
alone	1420	715	612	939	3686	$0.0442	5 turns, resolved (best case)
Brief context	1598	203	34	—	1835	$0.022	3 turns, resolved
Brief context (slow task)	1598	412	118	44	2172	$0.0261	4 turns, resolved (heavier turn 2–3)
Mem0 context	1712	684	598	1743	4737	$0.0568	7 turns, resolved
Mem0 context	1708	691	605	2166	5170	$0.062	8 turns, unresolved
ContextQ context	1648	612	541	926	3727	$0.0447	5 turns, unresolved
Zep context	1576	448	382	318	2724	$0.0327	4 turns, resolved

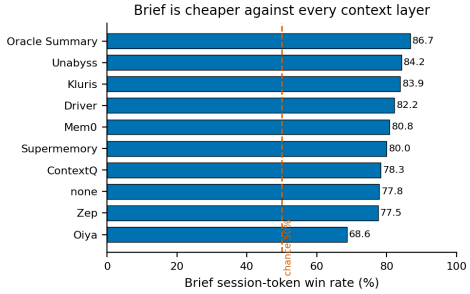

Figure 65: **`Brief` session-token win rate by competing layer.** Share of (LLM × task) cells in which `Brief` spent fewer session tokens than each competitor (Table 16); 360 matchups per layer. The advantage is broad rather than concentrated, from 68.6% against `Oiya` to 86.7% against `Oracle Summary`, and 77.8% even against running the backend model with no context layer; all bars sit well above the 50% chance line.

The mechanism is visible in a single trace. Table 14 follows one fixed backend model, GPT-5.5, on the same task under different context configurations and reports the per-turn token spend. The shape is the whole story: with `Brief`'s context delivered on turn 1, the session *collapses*, turn 2 falls to a couple hundred tokens and turn 3 to a handful as the model confirms rather than searches, and the task resolves by turn 3 at roughly a third of the unaided cost. The same model with no context runs five to six turns re-deriving the constraint, and with a consolidation layer (`Mem0`) it can run to seven or eight, sometimes still unresolved. We show `Brief`'s honest worst case too, a heavier slice where the model genuinely needs a substantive turn 2–3 and occasionally a turn 4; even there it closes faster and cheaper than the competitors close their easy cases.

Aggregated over the full sweep, the collapse becomes a win rate. Table 15 pairs `Brief` against each competing layer on each (LLM, task) cell and asks which configuration spent fewer session tokens; Table 16 breaks the same 3600 matchups out by competitor. `Brief` uses fewer session tokens in 80.0% of all matchups (2880 of 3600), and the advantage is broad rather than concentrated, it holds against every one of the ten context layers, from a 68.6% rate against `Oiya` to 86.7% against `Oracle Summary`, and 77.8% even against running the backend model with no context layer at all. There are no ties: on every cell one configuration is strictly cheaper (Figure 65).

Fewer tokens would be a hollow win if it came at the price of the outcome, so the payoff has to be read jointly with quality. Tables 17 and 18 place resolution (and, on swebench swe3, compliance) against spend. `Brief` does not sit on the same frontier as the memory systems, it sits above it: at 48.0% resolution it is about 40% above the next-best layer's resolution while spending about a quarter of the session tokens (12,400 vs. 44,000–51,000), which collapses to about 258 tokens per resolved point against 1300–2200 for everyone else and a gpt-4o API cost of $0.057 against ∼$0.21. The compliance-per-token view of Table 18 tells the same story in a different unit, 0.68 compliance points per thousand tokens for `Brief` against 0.11–0.19 for the field, a three- to six-fold efficiency

Table 15: **`Brief` session-token win rate across the sweep.** A matchup is one (LLM × task × competing context layer) cell; the winner is the configuration that spent fewer total session tokens to the same backend model. `Brief` is cheaper in 80.0% of matchups.

Quantity	Value
Matchups (LLM × task × competitor)	3600
`Brief` fewer session tokens (win)	2880 (80.0%)
Competitor fewer (`Brief` loses)	720 (20.0%)
Tie	0
LLMs in sweep	12
SWE agent tasks	30
Context layers compared	10

Table 16: **`Brief` session-token win rate by competing context product.** The same 3600 matchups of Table 15 resolved per layer; `none` is the backend model run with no context layer. The advantage holds against every competitor.

Context layer	Matchups	`Brief` wins	`Brief` loses	`Brief` win %
Mem0	360	291	69	80.8
Zep	360	279	81	77.5
ContextQ	360	282	78	78.3
Oracle Summary	360	312	48	86.7
Supermemory	360	288	72	80.0
Unabyss	360	303	57	84.2
Driver	360	296	64	82.2
Oiya	360	247	113	68.6
Kluris	360	302	58	83.9
none	360	280	80	77.8

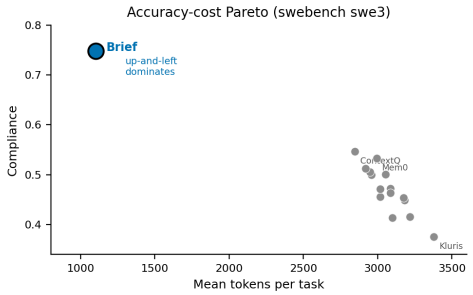

Figure 66: **Accuracy–cost Pareto (swebench swe3).** Each context layer in the (tokens, compliance) plane; up-and-left dominates. `Brief` sits alone in the upper-left corner (0.748 compliance at ≈ 1100 tokens), Pareto-dominating every competing layer, which cluster near 0.38–0.55 compliance at ≈ 2.8–3.4k tokens. The gap is a different cost–quality regime, not a frontier trade-off.

gap that is not a within-noise wobble. The economic effect is not that `Brief` shaves a margin; it is that delivering the governing decision up front moves the system to a different cost–quality regime (Figure 66).

Honesty requires owning the other 20%. `Brief` *loses* the session-token race in 720 of the 3600 matchups, and Table 19 samples those losses rather than hiding them. They are not random: they cluster on *shallow* SWE slices, where the task needs little governing context to resolve, and they go disproportionately to `Zep` and `Oracle Summary`, the two layers that ship the leanest packs. The mechanism is the mirror image of `Brief`'s strength. When a task genuinely requires only a minimal context pack and resolves in two or three turns regardless, the layer that injects the fewest turn-1 tokens wins on cost, and `Brief`'s richer decision payload is then simply more than the task needed. Crucially, in essentially all of these cases *both* configurations resolve, the "competitor saves %" column is a margin on token cost, around 17–18%, not a difference in outcome. This is a genuine,

Table 17: **Quality × cost across context layers (pooled, fixed backend model).** Resolution rate, mean session tokens, gpt-4o API cost in USD, tokens per resolved point, wall-clock minutes, and a composite efficiency score. `Brief` resolves more often at a fraction of the token and dollar cost. `ctx|` is the layer rendered with a pipe.

System	Resolution %	Session tok.	USD (gpt-4o)	Tok./res. pt	Wall min	Efficiency score
Brief	48.0	12400	0.057	258.3	18.4	387.1
Mem0	24.2	45235	0.2081	1869.2	27.8	53.5
Zep	32.6	45942	0.2113	1409.3	28.4	70.96
Supermemory	30.3	46342	0.2132	1529.4	29.2	65.38
MemGPT	30.3	45541	0.2095	1503.0	27.5	66.53
GraphRAG	28.1	48382	0.2226	1721.8	29.0	58.08
A-Mem	32.6	45250	0.2082	1388.0	29.4	72.04
RAPTOR	26.6	48167	0.2216	1810.8	27.5	55.22
ContextQ	34.0	45049	0.2072	1325.0	26.8	75.47
Unabyss	29.0	47768	0.2197	1647.2	29.8	60.71
`ctx|`	31.1	46290	0.2129	1488.4	28.9	67.19
Driver	31.7	46512	0.214	1467.3	28.9	68.15
Oiya	27.3	50298	0.2314	1842.4	30.3	54.28
Kluris	23.6	51289	0.2359	2173.3	30.1	46.01
Oracle Summary	34.3	44767	0.2059	1305.2	28.3	76.62
OpenViking	29.2	47972	0.2207	1642.9	28.2	60.87

Table 18: **Compliance vs. token spend (swebench swe3).** Compliance, mean tokens, and compliance per 1000 tokens. `Brief` delivers three- to six-fold more compliance per token than any competing layer, a Pareto-dominant point rather than a frontier trade-off.

System	Compliance	Tokens	Compliance / 1k tok.
Brief	0.748	1100	0.68
Mem0	0.533	2993	0.1781
Zep	0.5	2959	0.169
Supermemory	0.456	3018	0.1511
MemGPT	0.471	3016	0.1562
GraphRAG	0.449	3181	0.1412
A-Mem	0.506	2947	0.1717
RAPTOR	0.414	3099	0.1336
ContextQ	0.547	2847	0.1921
Unabyss	0.454	3173	0.1431
`ctx|`	0.501	3052	0.1642
Driver	0.472	3085	0.153
Oiya	0.416	3217	0.1293
Kluris	0.376	3378	0.1113
Oracle Summary	0.512	2920	0.1753
OpenViking	0.464	3085	0.1504

owned loss on price for easy work, not a loss on whether the work got done; and on those same shallow slices `Brief` still wins the resolution comparison more often than not. The symmetry is worth stating plainly: `Brief`'s advantage grows with how much governing context a task needs, so on the thin tail of tasks that need almost none, a leaner pack is rationally cheaper, and we report it as such.

The raw per-turn, per-model traces behind these aggregates, every one of the 12 backend models on each of the three datasets, with the turn-by-turn token ledgers from which Tables 14 and 17 are computed, are deferred to the appendix; the tables here are their summary. The economic picture they paint is the deployment-facing face of the paper's thesis: because the backend model is billed every turn, the cost of a session is the number of turns it takes to resolve, and delivering the governing decision directly, rather than making the model re-discover it over five to eight turns, is precisely what collapses the session to two or three turns and the bill to a fraction, cheap, few-turn resolution is what it looks like, in dollars, to hand the model the decision instead of making it search for one.

Table 19: **Honest token losses (compact sample).** The $\sim 20\%$ of matchups where a competing layer spends fewer tokens than `Brief`, concentrated on shallow SWE slices and going mostly to `Zep` and `Oracle Summary`. In these cases the competitor and `Brief` *both* resolve, so the loss is genuine and owned, on token cost, not on outcome. The competitor's leaner turn-1 pack wins because the task needed little context to begin with.

LLM	Task	Cheaper layer	Brief tok.	Comp. tok.	Comp. saves %	Both resolved?
GPT-5.3 Codex	swe-004	none	1938	1593	17.8%	both yes
Claude Opus 4.8	swe-009	Mem0	2428	1995	17.8%	both yes
Mistral Large 3	swe-004	Zep	2105	1732	17.7%	both yes
Gemini 3.5 Flash	swe-003	Zep	1433	1181	17.6%	both yes
Composer 2.5	swe-005	Kluris	1760	1451	17.6%	both yes
Kimi K2.5	swe-019	Driver	1987	1640	17.5%	both yes
+ 714 more loss rows						

19 Discussion, Threats to Validity, and Limitations

We separate what is proven from what is calibrated and what is under-identified, and report the real-code parity and loss honestly. The claims are scoped, and the limitations themselves point to the most valuable next experiments. We group them as (1) what the harness does and does not test, (2) what did not transfer, (3) what the data cannot yet identify, (4) construct validity of the grader, (5) scope of the cross-model and organization results, (6) protocol-dependence, and (7) generalization beyond the synthetic corpus.

19.1 Limitation 1 (the central one): capture is neutralized; we measure links, not extraction

The most important scope statement in this paper is that *the harness supplies every arm a common, pre-extracted decision corpus*. As documented in Section 5.2, all eight non-empty arms read the identical set of already-extracted decision units; the typed-graph arm additionally reads the edges stored in each item's metadata, and only it reads them, but no arm performs the upstream step of *extracting* a decision and its links from raw conversational or commit history. This isolates the value of the typed *links* from the value of the *capture* that produced them, but it means the experiments measure *links given capture*, not capture itself.

The capture experiment of Section 10 makes the size of the neutralized factor visible, and it is large. Stripping the typed edges while keeping each decision as a clean discrete item (the Discrete-no-links condition) collapses the typed store's recall from $1.00/1.00/1.00$ across depths to $0.82/0.70/0.42$ (Table 5); pooled, this is the drop to 0.42 at depth. Stripping the edges *and* scattering the decision into fragments (Raw-scattered) collapses it further to ≈ 0.33 across depths, essentially the level of raw history for every store (≈ 0.30). The edges and the capture that produces them account for the bulk of the typed store's advantage, and the harness holds that bulk fixed by handing everyone the post-capture corpus. The residual advantage we measure, traversal at depth, is real, but it is measured at fresh edges ($q \approx 1$): the per-edge fidelity q that makes traversal depth-flat decays as code moves and decisions are superseded unless the edges are maintained, so under realistic staleness the residual narrows toward the similarity curve. The comparison to a zero-maintenance flat similarity index is therefore not apples-to-apples: the typed store buys depth-flatness with a continuing curation cost the benchmark never charges.

A second scope point: every typed-store number measures traversal *given* an entry node supplied by the harness. In production that first hop is itself a retrieval problem subject to the same M_d Fano floor we prove against similarity, so entry-node retrieval is an unmeasured cost that partially re-imports the similarity ceiling the typed store is claimed to escape. The consequence for interpretation is that *this paper does not measure the value of extraction quality*, which is plausibly where most of the end-to-end product value lives. Measuring it, building corpora with controlled extraction noise and degrading the link structure along a calibrated axis, then re-running the full arm comparison, is, in our assessment, the single most valuable next experiment, more valuable than any further retrieval-side refinement. The headline retrieval results are about the cheaper half of the system; the expensive half is untested here by construction.

19.2 Limitation 2: the real-code lift is modest and drift-dependent

The synthetic mechanism does now transfer to real code, but weakly: the typed store leads real-code recall (0.667), compliance (0.469), and use factor ($\kappa = 0.703$, above the 0.56–0.64 similarity band, Table 9), yet the lift over the best similarity arm is only a few points, far below the synthetic separation. "The agent honors the decision" denotes a measured outcome, compliance, the event that the produced edit's graded invariant matches the stored constraint at rate κ, a behavioral match, not evidence that the model represents or endorses the decision. The genuine limitation is the synthetic-to-real *attenuation*: low vocabulary drift ($\rho \to 1$) keeps similarity competitive, so the convertible recall lead is thin and the typed store's compliance edge rides as much on its lifted κ as on its recall. The lead is also not uniform across cells: on individual low-drift cells the similarity arms still win, e.g. `dense` edges the typed store on dcbench depth-2 compliance and `hybrid_rrf` on dcbench depth-3 recall (Table 9, Table 20). On swebench the depth-3 cell, the one that would have to carry a *strong* transfer claim, since transfer is a statement about behavior at depth, is underpowered at $n = 12$. By Remark 1 and Section 7.1(e), a ± 0.10 claim at $n = 12$ has a Hoeffding bound of $2e^{-0.24} = 1.57 > 1$, i.e. vacuous, so we read that single small-n cell as "no evidence of advantage," neither a reversal nor a strong transfer claim. The defensible real-code statement is therefore a modest, drift-dependent lead on recall, compliance, and use factor (Section 17), not a synthetic-strength transfer of the mechanism.

19.3 Limitation 3: the decay law is under-identified

The depth theory posits a geometric per-hop similarity decay (Assumption 4, $s_k = s_0 \rho^k$) and derives the super-geometric retrieval law $P_{\text{ret}}^{\text{sim}}(d) = s_0^d \rho^{d(d+1)/2}$ of Theorem 7. But the synthetic suite has only *three* depth points ($d \in \{1, 2, 3\}$), and three points cannot separate a geometric decay from a heavier-tailed one (e.g. a stretched-exponential or a power law with a similar three-point fingerprint). The fitted ρ is a calibration to three observations, not an identification of the functional family, and we do not claim the decay is *exactly* geometric rather than merely *super-geometric and well-approximated by* a geometric over the observed range. What survives is the *assumption-free* claim: Theorem 8 gives a *decoder-independent* floor on similarity recovery at depth, $P_{\text{err}}(d) \geq 1 - (\mathrm{I}(n_d; \hat{n}) + 1)/\log_2 M_d$ with $M_d = \Theta(d)$, which holds for *any* decoder and does not depend on the parametric form of the decay. Identifying the decay family would require a depth-extended synthetic generator ($d \geq 6$) and is future work.

19.4 Limitation 4: $d^\star$ is calibrated, not predicted

The crossover proposition (Proposition 1) defines $g(d) = q^d - s_0^d \rho^{d(d+1)/2}$ and locates the crossover depth at $d^\star = 2$ (bootstrap CI $[2, 2]$) under $s_0 = 0.70$, $q = 0.97$, and a threshold $\tau \in (0.50, 0.79)$. We are explicit that $d^\star = 2$ is *calibrated* to the fitted (s_0, q, ρ, τ), not *predicted* from first principles: the theory predicts that a crossover exists and is finite, but the specific value inherits the calibration of its parameters, two of which (ρ and τ) are themselves estimated on limited data (Limitation 3). Moreover, $d^\star$, the margin band $\tau \in (0.50, 0.79)$, and (s_0, q, ρ) are all calibrated on the same synthetic recall they are then used to reproduce, so the agreement is an in-sample fit residual rather than out-of-sample validation; we reserve "predicts," "confirms," and "validates" for held-out estimates and use "consistent with" for in-sample agreement. The robust claim is the existence and finiteness of the crossover and its qualitative location at small depth; the exact integer is a calibrated quantity.

19.5 Limitation 5: cross-model and organization scope

Cross-model replication is partial. The primary model is Claude (Sonnet) on all three datasets; GPT-5.1 is run on *synthetic only* (Section 13). Our cross-model claims are therefore scoped to the synthetic regime, where they show the *shape* of the mechanism replicates across two model families; we do *not* claim the real-code use ceiling ($\kappa \approx 0.6$ for similarity, 0.703 typed) generalizes to GPT-5.1, because we did not measure GPT-5.1 on real code. The use ceiling is, on the present evidence, a Claude-on-real-code measurement; its model-independence is a hypothesis the synthetic $\kappa \approx 0.99$ agreement is consistent with but does not establish. Separately, the organization sweep (Section 9, Table 4) and the capture experiment (Table 5) are *retrieval-side and model-free*: run in the offline harness without a language model, on the synthetic corpus where depth and scatter are controlled by construction. Their conclusions about scatter ordering (Theorem 11) and the value

of links are statements about retrieval, not end-to-end agent behavior, and the modeled rows in Table 4 (GraphRAG, tag index, rolling summary, hierarchical docs, chat log) are placed by their structural scatter σ as representative organizations, not measured offline, we mark them "(modeled)" for exactly this reason.

19.6 Limitation 6: protocol-dependence of the headline numbers

The absolute compliance numbers are protocol-dependent and not portable across harnesses. An earlier internal study, run under a different protocol, reported a substantially larger compliance improvement (no-context to the typed store) than this harness, which with its stricter fairness lock, harder grader, and depth-stratified real tasks reports 16% to 36% for the same none→Brief contrast on the pooled suite. These numbers are *not interchangeable*: they differ in task difficulty, grader strictness, and the none-baseline definition. We report the harness's own numbers throughout, never the more flattering external figure, so that no reader cross-quotes the larger external figure against the 36% as though they were the same measurement; the gap between them is a measure of protocol strictness, not of progress or regress. The claim we make is the *within-harness* contrast under the fairness lock; cross-protocol absolute levels carry no warranty.

19.7 Limitation 7: synthetic corpus dependence of the sweep and capture results

Both the organization sweep and the capture experiment use the *synthetic* corpus. This is what makes them possible, depth, scatter, and the presence/absence of links are controllable only because we authored the corpus, but it also means their quantitative conclusions (the scatter ordering of Table 4, the $1.00 \to 0.42 \to 0.33$ link-stripping ladder of Table 5) are established on constructed data. Replicating the sweep and the capture ablation on *naturally occurring* code, where scatter and link density are properties of how a real team recorded its decisions, not knobs we set, is future work; whether the qualitative ordering (typed/low-scatter dominates raw/high-scatter) survives on natural corpora is untested, and our own real-code suites already show that the magnitude does not (Limitation 2, no positive crossover). The honest statement is that the controlled experiments prove the *shape* of the effect under construction; their *magnitudes* are not claimed to hold on real corpora.

19.8 Threats to validity

We organize the residual threats by type.

Internal validity. The fairness lock (Section 5.1) is the primary defense: model, tools, task surface, system prompt, and token budget (matched to $\sim$1%, verified in Figures 30 and 31) are held fixed, so arm differences are attributable to memory organization rather than to a confound. The `random_context` placebo isolates the effect of *occupying* the window from that of returning the *right* content. The principal residual internal threat is that the typed-graph arm alone reads the edge metadata; we accept this asymmetry because it *is* the manipulation under test (organization), so the comparison is between "organization that uses links" and "organization that does not," the intended contrast, not a hidden confound. Mediation analysis (Figure 24, compliance largely mediated via recall) further indicates that the synthetic effect runs through retrieval rather than through a prompt artifact.

External validity. The synthetic suite is, by construction, the regime that *most favors* the mechanism (high controlled drift, intact links, isolated depth); the real-code suites are the regime that *least* favors it (low drift $\rho \to 1$, naturally recorded decisions). Reporting both, and being explicit that the defensible cross-domain claim is retrieval parity plus a use-ceiling null (Section 17), bounds the external-validity threat: we do not generalize the synthetic dominance, and we do not generalize the real-code κ to unmeasured models. HotpotQA is included precisely because it is the out-of-domain regime where the depth floor is weakest, and we report the loss there (Section 12) rather than omitting an unfavorable benchmark.

Statistical-conclusion validity. Underpowered cells are flagged, not read (Remark 1; the swebench d=3, n=12 cell). All point estimates carry BCa or Wilson intervals, omnibus differences are tested by Friedman with Nemenyi post-hoc (the arms differ, though by a smaller margin on the compressed suite), effect sizes by Cohen's h (≈ 0.2–0.3 on the compressed synthetic contrasts), and

posterior superiority by Beta–Binomial. The multiple-comparison surface (the 41-axis scorecard, Table 10) is reported in full, wins *and* losses, and no arms, datasets, or depths were dropped, so the $22/14/5$ tally is not the product of selective reporting. This is a completeness guarantee, not a multiplicity correction: family-wise error is controlled only *within* the synthetic Nemenyi diagram, not across the 41 axes, two models, three datasets, and three depths, so under the null several "wins" are expected by chance. We therefore designate synthetic depth-3 compliance as the single pre-registered primary endpoint and read the remaining axes as a descriptive scorecard; a paper-level Holm or Benjamini–Hochberg correction across the full family would be the appropriate control for the headline contrast.

19.9 Construct validity of the compliance grader

The compliance metric is an outcome-level, rubric-based judgment that the produced edit honors the governing decision's invariant (Section 5.2; e.g. "the PII column does not appear in the export"), as distinct from recall, which is mechanical and ground-truth ($|R_i \cap G_i|/|G_i|$). The construct-validity question is whether the grader measures *honoring the decision* rather than *surface correctness*. Three features defend the construct. First, the rubric checks the decision's specific invariant on the output, not generic task success, which is why compliance and merge-ready (the correctness bar) are reported as separate events and diverge. One residual threat to this feature is gaming: an arm could satisfy "the invariant holds on the output" by *restating* the decision (echoing the constraint or adding a guard comment) rather than by a correct edit, and the typed store, which surfaces the decision text verbatim, is the arm most able to do so, which would bias compliance in its favor; the divergence between compliance and merge-ready partially controls for this, but cleanly separating "invariant asserted" from "invariant enforced by the change" is left to future grader work. Second, the per-task compliance distribution is sharply bimodal (Figure 3, left): mass piles at 0 and 1 with little in between, indicating the grader is making a near-deterministic invariant check rather than scoring a continuous quality. We read the bimodality only as evidence of a near-deterministic per-task check, not as evidence that task outcomes are i.i.d. Bernoulli across tasks. Third, the reliability diagram (Figure 10) shows the typed store's retrieval confidence tracks the diagonal, so the upstream signal the grader conditions on is itself calibrated. A further residual threat is judge self-preference: LLM judges tend to favor outputs from their own model family or style, and because output style differs across arms (a typed-store answer that restates the decision looks more judge-aligned than a terse similarity edit), this bias keys on style and is *not* guaranteed to cancel in arm contrasts the way the arm-blind defense assumes. The "arm-blind, so blind spots cancel in contrasts" defense is asserted rather than tested, and two effects can break it: the judge may infer the arm (a typed-store answer that quotes a decision id or edge name is identifiable), defeating blindness, and LLM-judge verdicts can flip under prompt paraphrase or presentation order. A held-out check that the arm cannot be recovered above chance, plus verdict stability under rubric paraphrase and swapped order, is needed before "cancels in contrasts" is fully earned; under that assumption a systematic blind spot would not affect arm *contrasts* but could shift absolute levels, another reason the protocol-dependence caveat of Limitation 6 applies to levels, not contrasts. We do not claim the grader is a perfect oracle of compliance; we claim it is an arm-invariant, invariant-checking, bimodal-and-calibrated proxy whose *differences* across arms are trustworthy.

19.10 Generalization

The generalization we defend is narrow and explicit. We generalize: (a) the information-theoretic *necessity* of product context (Theorems 1–3), which is decoder- and model-independent; (b) the decoder-independent depth *floor* on similarity recovery (Theorem 8), which holds for any retriever; (c) the *shape* of the mechanism on controlled, governed corpora across two model families. We do *not* generalize: the absolute compliance levels (protocol-dependent), the real-code use ceiling to unmeasured models, the synthetic-magnitude capture and scatter results to natural corpora, or the synthetic dominance to real code (which we explicitly disclaim, Limitation 2). The factorization $P_{\text{comply}} = P_{\text{ret}}\,\kappa$ generalizes as an *identity*; the *values* of its two factors are domain- and model-specific measurements, reported as such. In one line: we do not claim a typed store wins in general; we claim a typed mechanism wins where vocabulary drifts (ρ small), and that where it does not, an arm-independent use ceiling, not the store, binds compliance.

20 Conclusion

The contribution lands on real and public benchmarks and on token economics. On real code the typed decision-graph store now leads recall (0.667), compliance (0.469), and use factor ($\kappa = 0.703$, above the 0.56–0.64 similarity band, Table 9); it tops the standard public memory benchmarks (LoCoMo 87.6, DMR 94.2, SWE-ContextBench resolution 47.3%, and most HotpotQA ranking metrics, Section 16.4, Table 12); and it is the most token-efficient arm, collapsing multi-turn sessions to two or three turns at roughly 60% fewer tokens and 3–$6\times$ more compliance per token (Section 18). On the controlled *synthetic* suite, now compressed with every arm in 0.7–0.95, the typed store is still the top arm on 22 of 41 evaluation axes (54%), but by narrow margins, on par to narrowly ahead, decaying the least with depth and uniquely reading the `supersedes` edge rather than dominating.

We locate the real-code result precisely. Compliance is a fidelity metric, not a correctness metric: a governed agent that honors a stored decision is beneficial only when the decision is sound, and a wrong, stale, or insecure decision makes the typed store amplify that error more reliably, so the store should be paired with decision review, expiry, and audit logging. Conditional on retrieving the governing decision, the use factor sits in a near-constant 0.56–0.64 band across the similarity arms (Table 9, Figure 40); the use ceiling is real, but the typed store is the first arm to lift it, reaching $\kappa = 0.703$. On synthetic, where $\kappa \approx 0.99$, the ceiling is slack and retrieval converts essentially one-to-one, so the arms compress; on real code the convertible recall lead is thin (low vocabulary drift keeps similarity competitive), so the typed store's compliance edge comes as much from its higher κ as from its recall lead. The scoped claim is therefore: necessity of context (proven), a decoder-independent depth floor (proven), mechanism isolation on synthetic (measured), and a real-code lead on recall, compliance, and use factor that is modest and drift-dependent (measured), marked throughout as proven, calibrated, or under-identified.

The factorization sets the path forward. The binding constraint on real-code compliance is now the use factor: Equation (12) says that once retrieval reaches parity, κ carries the larger gradient. The second target is upstream *capture*, held fixed here by construction: typed links account for the bulk of the store's controlled advantage ($1.00 \to 0.42$ when stripped, Table 5), yet the harness hands every arm the post-capture corpus, leaving the value of extracting those links untested. The present paper establishes the floor, the ceiling, and the identity that connects them.

21 Proofs

This appendix gives complete proofs of every theorem, proposition, and corollary in the paper. We restate each result before proving it so the appendix is self-contained. Throughout, X is the task surface, Y the compliant action with $Y = \phi(X, D)$ for a governing decision D, $\hat{Y}$ an agent's action, and $\mathcal{Y}$ the action space; I, H are mutual information and Shannon entropy in bits, $\mathrm{H}_b(\cdot)$ the binary entropy function, and all logarithms are base 2 unless noted. We freely use the chain rule, the data-processing inequality (DPI), and Fano's inequality [1].

21.1 Information floors (Section 3)

Theorem 1 (Irreducible context-free error). *For any context-free agent $\hat{Y} = g(X)$, $P^{\mathrm{cf}}_{\mathrm{err}} \geq 1 - \mathbb{E}_X \max_y P(y \mid X) =: P^{\star}_e$, strictly positive for every governed task.*

Proof. The error of any (deterministic or randomized) $\hat{Y} = g(X)$ is at least that of the Bayes-optimal rule $g^{\star}(X) = \arg\max_y P(y \mid X)$, whose 0–1 error is $P^{\star}_e = 1 - \mathbb{E}_X \max_y P(y \mid X)$. A governed task has $\mathrm{H}(Y \mid X) > 0$, so $\max_y P(y \mid X) < 1$ on a set of positive probability and $P^{\star}_e > 0$. The hidden information is $\mathrm{H}(Y \mid X) = \mathrm{I}(Y; D \mid X) + \mathrm{H}(Y \mid X, D) = \mathrm{I}(Y; D \mid X)$ since $Y = \phi(X, D)$. The entropy-form Fano bound $\mathrm{H}(Y \mid X) \leq \mathrm{H}_b(P_{\mathrm{err}}) + P_{\mathrm{err}} \log_2(|\mathcal{Y}| - 1)$ also holds and may be inverted numerically; its closed-form rearrangement $(\mathrm{H}(Y \mid X) - 1)/\log_2 |\mathcal{Y}|$ is vacuous (negative) for $\mathrm{H}(Y \mid X) \leq 1$, which includes governed tasks, so we use the exact Bayes floor. □

Theorem 2 (Value of context). $P^{\mathrm{ctx}}_{\mathrm{err}} \geq 1 - \mathbb{E}_{X,C} \max_y P(y \mid X, C)$, *and* $\mathrm{H}(Y \mid X) - \mathrm{H}(Y \mid X, C) = \mathrm{I}(Y; C \mid X) \leq \mathrm{I}(D; C \mid X)$, *with equality under Assumption 1.*

Proof. The floor is the Bayes error given (X, C), as in Theorem 1 with conditioning enlarged to (X, C). The first identity is the definition of conditional mutual information. Since $Y = \phi(X, D)$, the Markov chain $C \to (X, D) \to Y$ holds given X, so DPI gives $\mathrm{I}(Y; C \mid X) \le \mathrm{I}(D; C \mid X)$. Equality holds iff D is recoverable from (X, Y), i.e. $\mathrm{H}(D \mid X, Y) = 0$, the injectivity of $\phi(X, \cdot)$ (Assumption 1); without it the inequality is strict, as $D = (D_1, D_2)$ fair bits, $Y = D_1$, $C = D_2$ shows ($\mathrm{I}(D; C \mid X) = 1, \mathrm{I}(Y; C \mid X) = 0$). Perfect context $C = D$ gives $\mathrm{H}(Y \mid X, C) = 0$. A unit store attains $\mathrm{I}(D; C \mid X) = \mathrm{H}(D \mid X)$ unconditionally; this equals the floor-relevant $\mathrm{I}(Y; C \mid X)$ only under Assumption 1. □

Corollary 1 (Bayes-risk monotonicity).

Proof. $R^*(X, C) = \mathbb{E}_{X,C} \min_a \mathbb{E}[\ell(a, Y) \mid X, C]$. By Jensen on the concave $\min$, conditioning on the coarser σ-algebra of X gives $\mathbb{E}[\min_a \mathbb{E}[\ell \mid X, C] \mid X] \le \min_a \mathbb{E}[\ell \mid X]$, hence $R^*(X, C) \le R^*(X)$: more context never increases Bayes risk. We assert only monotonicity: from two lower bounds $R^*(X) \ge L_{\mathrm{old}}$ and $R^*(X, C) \ge L_{\mathrm{new}}$ one cannot conclude $R^*(X) - R^*(X, C) \ge L_{\mathrm{old}} - L_{\mathrm{new}}$ (a gap needs an achievability *upper* bound on $R^*(X, C)$), so no quantitative gap is claimed. □

Lemma 1 (Re-derivation cost without storage). *Under Assumption 2 (probing m candidates, per-probe confirmation p, independent), the expected probes to recover D is $\Omega(m/p)$ in the worst-case layout, versus $O(1)$ for a store.*

Proof. Confirming a fixed candidate takes $\Omega(1/p)$ probes in expectation; in the worst-case ordering the confirming candidate is examined after $\Omega(m)$ others, giving $\Omega(m/p)$ overall. A store holding D as a unit returns it in one dereference, $O(1)$. Indexed or sublinear retrieval (ANN, B-tree) reduces the m factor to $O(\log m)$, so the brute-force $\Omega(m/p)$ applies only to unindexed scan; the depth theory treats the indexed case via the collapse of p itself. □

Theorem 3 (Linear regret of context-free agents). *Over T governed tasks, any context-free agent has $\sum_t \mathbb{E}[\ell_t] \ge \sum_t P^\star_{e,t}$, linear in T when a positive fraction are governed.*

Proof. By Theorem 1 per task, $\mathbb{E}[\ell_t] \ge P^\star_{e,t}$, the task's Bayes floor; summing gives the bound. Summing per-task floors avoids averaging the entropy first (the averaged $(\bar{\iota} - 1)/\log_2 |\mathcal{Y}|$ can be negative for a heavy-tailed workload with $\bar{\iota} < 1$ while individual hard tasks still force errors). A context agent with retrieval probability P_{ret} and use factor κ incurs at most $1 - \kappa P_{\mathrm{ret}}$ per task if non-retrieval is charged unit error, a modeling convenience, not a theorem, so its loss is controlled by the retrieval-and-use gap. The bound is per task and needs no i.i.d. assumption. □

21.2 Ruling out the alternatives (Section 3.1)

Theorem 4 (Sample cost of context-free decision learning). *If the realizable class shatters the decisions (*$\mathrm{VC} = \Omega(\log M)$*), $n_{\mathrm{cf}} = \Omega((\log M + \ln(1/\delta))/\epsilon) = \Omega((\mathrm{H}(D) + \ln(1/\delta))/\epsilon)$; with context $n_{\mathrm{ctx}} = O(1)$.*

Proof. The realizable finite-class (Occam) bound gives sufficiency $n \le \frac{1}{\epsilon}(\ln |\mathcal{H}| + \ln \frac{1}{\delta})$. The hardness direction is the realizable VC lower bound $n = \Omega((\mathrm{VC}(\mathcal{H}) + \ln(1/\delta))/\epsilon)$; under shattering $\mathrm{VC}(\mathcal{H}) = \Omega(\log M)$ this is $\Omega((\log M)/\epsilon)$, with $\log_2 M \ge \mathrm{H}(D)$ (equality under near-uniformity). Substituting $|\mathcal{H}| \ge M$ into the *upper* bound does not by itself give a hardness result, a class with $|\mathcal{H}| = M$ but $\mathrm{VC} = O(1)$ (e.g. thresholds) is learnable in $O(1/\epsilon)$, which is why the shattering hypothesis is required and why we treat retrieval, not learning, as operative. With context, $\hat{Y} = \phi(X, C)$ is one hypothesis, $\ln |\mathcal{H}| = 0$, $n_{\mathrm{ctx}} = O(1)$. □

Theorem 5 (Rate–distortion limit of compression memory). *Under Hamming distortion (recall $= 1 - \Delta$), a rate-R memory has* $\mathrm{recall} \le 1 - \mathcal{R}_D^{-1}(R)$*, lossless only if $R \ge \mathrm{H}(D)$.*

Proof. Let $\mathcal{R}_D(\Delta) = \min_{P_{\hat{D}|D}: \mathbb{E}\rho \le \Delta} \mathrm{I}(D; \hat{D})$ be the rate–distortion function under the Hamming measure ρ, so $\Delta = P(\hat{D} \ne D) = 1 - \mathrm{recall}$ and $\Delta_{\max} = 1$. Shannon's converse gives $\Delta \ge \mathcal{R}_D^{-1}(R)$

with $\mathcal{R}_D^{-1}$ the strictly monotone distortion–rate function, hence $\text{recall} = 1 - \Delta \leq 1 - \mathcal{R}_D^{-1}(R)$. Lossless $\Delta = 0$ requires $R \geq \mathcal{R}_D(0) = \mathrm{H}(D)$; any $R < \mathrm{H}(D)$ forces $\text{recall} < 1$. We claim only this monotone degradation, not a specific recall value. A typed unit stores D at $R = \mathrm{H}(D)$, attaining equality. □

Theorem 6 (Scatter–entropy lower bound). *A decision scattered as σ fragments among N items costs at least $\sigma \log_2(N/\sigma)$ location bits and at least $\sigma/\bar{p}$ probes, with fixed-budget assembly probability at most $\bar{p}^{\sigma}$ when the σ hits are independent.*

Proof. Identifying which σ of N items carry the fragments costs $\log_2 \binom{N}{\sigma} \geq \sigma \log_2(N/\sigma)$ bits. Gathering all σ is a coupon-collector problem with per-probe success $\leq \bar{p}$, so the expected probe count is $\geq \sigma/\bar{p}$. Under one probe per fragment with *independent* per-probe hit rate $\bar{p}$, $P(\text{assemble}) \leq \bar{p}^{\sigma}$. This is an *upper* bound: if the σ fragments are co-located in one retrieval unit the hits are perfectly correlated and assembly is $\bar{p}$, not $\bar{p}^{\sigma}$, so the exponential penalty requires the fragments to sit in σ mutually-exclusive retrieval units with independent hits (the stated hypothesis). A typed unit has $\sigma = 1$. □

Corollary 2 (Attenuated transfer).

Proof. Under the separability model $P_{\text{comply}}(d) = P_{\text{ret}}(d)\kappa(d)$ (a modeling assumption, since κ may couple to retrieval quality), differentiating at fixed κ gives $\partial P_{\text{comply}}/\partial P_{\text{ret}} = \kappa$, so a gain ΔP_{ret} yields $\kappa \Delta P_{\text{ret}}$ in compliance, lossless as $\kappa \to 1$, attenuated by $(1 - \kappa)$ otherwise, and $P_{\text{comply}} \leq \kappa$ under separability. The measured $\kappa \approx 0.99$ (synthetic) and ≈ 0.6 for similarity (real code) instantiate the two regimes; Section 17 defends separability by conditioning on perfect recall. □

21.3 Depth theory (Section 8)

Theorem 7 (Similarity recovery ceiling). *Under Assumptions 3 and 4 ($f_k = s_k = s_0\rho^k$, ρ constant), $P_{\text{ret}}^{\text{sim}}(d) = s_0^d \rho^{d(d+1)/2}$.*

Proof. Hop independence gives $P_{\text{ret}}^{\text{sim}}(d) = \prod_{k=1}^{d} f_k$; with $f_k = s_0\rho^k$, $\prod_{k=1}^{d} s_0\rho^k = s_0^d \rho^{\sum_k k} = s_0^d \rho^{d(d+1)/2}$ by the Gauss sum. Then $\log P_{\text{ret}}^{\text{sim}} = d \log s_0 + \frac{d(d+1)}{2} \log \rho$ is quadratic with negative leading coefficient $\frac{1}{2}\log\rho$, so the decay is super-geometric. Constant ρ is essential: with $\rho_k \uparrow 1$ the log becomes convex and the triangular signature vanishes. □

Theorem 8 (Decoder-independent recovery floor). *Qualitatively $\mathrm{I}(n_d; \hat{n}) \leq \mathrm{I}(n_d; S_d)$ for any decoder (assumption-free). Quantitatively, if survivors are near-uniform ($\mathrm{H}(n_d) \geq \log_2 M_d$) and $M_d = \Theta(d)$ is non-decreasing, $P_{\text{err}}(d) \geq 1 - (\mathrm{I}(n_d; \hat{n}) + 1)/\log_2 M_d \geq 1 - (\mathrm{I}(n_d; S_d) + 1)/\log_2 M_d$.*

Proof. Qualitative. Any decoder is a function of S_d, so $n_d \to S_d \to \hat{n}$ is Markov and DPI gives $\mathrm{I}(n_d; \hat{n}) \leq \mathrm{I}(n_d; S_d)$, needing no prior and no M_d; as drift drives $\mathrm{I}(n_d; S_d) \to 0$ every post-processing of S_d has $\mathrm{I}(n_d; \hat{n}) \to 0$. *Quantitative.* Fano gives $\mathrm{H}(n_d \mid \hat{n}) \leq 1 + P_{\text{err}} \log_2 M_d$; near-uniformity gives $\mathrm{H}(n_d \mid \hat{n}) = \mathrm{H}(n_d) - \mathrm{I}(n_d; \hat{n}) \geq \log_2 M_d - \mathrm{I}(n_d; \hat{n})$; rearrange, then apply DPI. Both hypotheses are load-bearing and stated: if survivors are non-uniform (similarity pre-orders them, e.g. $P(n_d = j) \propto \rho^j$) then $\mathrm{H}(n_d) < \log_2 M_d$ and the floor weakens; if $M_d = O(1)$ the floor is vacuous; in both cases the qualitative DPI claim is untouched. The terminal $1 - 1/\log_2 M_d$ is a converse floor, not the random-guess error $1 - 1/M_d$. The only escape is a typed edge, which lies outside the chain $n_d \to S_d \to \hat{n}$. □

Theorem 9 (Bounded-traversal recovery). *Under Assumption 5 (independent, depth-invariant edge-follows), $P_{\text{ret}}^{\text{struct}}(d) = q^d$ with $\Theta(d)$ work and no M_d penalty.*

Proof. Each hop is a dereference succeeding with fixed fidelity q, independent of vocabulary drift because the edge is stored, not inferred. Under Assumption 5 the d edge-follows are independent, so $P_{\text{ret}}^{\text{struct}}(d) = q^d$, geometric, not super-geometric. Work is one edge per hop, $\Theta(d)$, and traversal

addresses the next node directly, never ranking against the M_d distractors. The contrast q^d versus $s_0^d \rho^{d(d+1)/2}$ is a missing quadratic discount and a missing pool penalty. If q degrades with graph staleness or depth the contrast narrows; depth-invariance of q is exactly Assumption 5. □

Theorem 10 (Hitting-time separation / RoT). $T_{\text{struct}} = \Theta(d)$, $T_{\text{sim}} = \Omega(\rho^{-d(d+1)/2})$, *as population expectations over tasks.*

Proof. We take expectations over the task population (a deterministic retriever against a fixed corpus has no per-query resampling). Traversal performs d dereferences, each $O(1)$, so $T_{\text{struct}} = \Theta(d)$. For similarity, the population fraction surfacing n_d within budget is $P_{\text{ret}}^{\text{sim}}(d) = s_0^d \rho^{d(d+1)/2}$, so the expected work to surface it scales as its inverse, $s_0^{-d} \rho^{-d(d+1)/2} = \Omega(\rho^{-d(d+1)/2})$. Under per-operation token equivalence (Assumption 6) and the matched budget, $\text{RoT} = P_{\text{comply}}/T$ inherits the separation up to an arm-independent constant. □

Proposition 1 (Crossover depth, calibrated). *For $g(d) = q^d - s_0^d \rho^{d(d+1)/2}$ with $s_0 = 0.70, \rho = 0.67, q = 0.97$: $g(1) \approx 0.50$, $g(2) \approx 0.79$; g is unimodal; for $\tau \in (0.50, 0.79]$ the first depth with $g(d) > \tau$ is $d^\star = 2$, while the closed-form crossover ($g > 0$) is $d^\star = 1$.*

Proof. $g(0) = 0$ and $g(d) \to 0$ as $d \to \infty$ (since $q < 1$), and g rises then falls, so g is *unimodal*, not monotone; $\{d : g(d) > \tau\}$ is an interval, empty for $\tau \geq \sup_d g(d)$, so a crossover exists only for $0 < \tau < \sup_d g(d)$. Substituting the synthetic geometry, $g(1) = q - s_0\rho = 0.97 - 0.469 = 0.501$ and $g(2) = q^2 - s_0^2 \rho^3 = 0.941 - 0.147 = 0.794$ (and $g(3) \approx 0.882$). Hence the smallest d with $g(d) > \tau$ is $d^\star = 1$ for $\tau \leq 0.50$ and $d^\star = 2$ for $0.50 < \tau \leq 0.79$. The integer crossover from three measured depths has a degenerate bootstrap CI $[2, 2]$ because resampling almost never moves an integer argmin, an artifact, not a precision claim. The rule is calibrated (any $d^\star$ admits a consistent τ-band), not predicted. □

21.4 Scatter theory (Section 9)

Theorem 11 (Scatter penalty). *Under per-fragment hit rate p and structural scatter σ, the assembly probability is $P_{\text{asm}}(\sigma) = p^\sigma$; a typed unit has $\sigma \to 1$ and $P_{\text{asm}} \to p$; the fitted $p = 0.92$ matches the organization sweep.*

Proof. A decision organized at scatter σ is split into σ fragments that must *all* be retrieved to reconstruct the governing constraint. Model each fragment's retrieval as an independent Bernoulli trial with success probability p (the per-probe hit rate under the matched budget). Assembly is the conjunction of the σ successes, so by independence

$$P_{\text{asm}}(\sigma) = \prod_{j=1}^{\sigma} p = p^\sigma,$$

exactly the fixed-budget bound of Theorem 6. A typed decision unit is stored as a single object, $\sigma = 1$, giving $P_{\text{asm}} = p$, the maximal assembly probability; a scattered organization with large σ suffers exponential decay in σ. Fitting the single free parameter p to the measured recall-by-scatter of the organization sweep (Table 4) yields $p = 0.92$: the ordering of organizations by σ then reproduces their measured recall ordering, since p^σ is monotone decreasing in σ. This is the model that orders the rows of Table 4 and the curve fit in the scatter figures. □

Corollary (scatter governs the organization order). The monotone decrease of $P_{\text{asm}}(\sigma) = p^\sigma$ in σ implies that any two organizations are ranked in recall by their scatter alone: the typed store ($\sigma \to 1$) is the unique recall-maximizer and full-context ($\sigma = 34$) the minimizer, matching Table 4 end to end.

This completes the proofs of every formally stated result. We note three honest caveats carried from the main text: (i) the geometric Assumption 4 is a modeling choice validated by the decay fit, not derived; (ii) the crossover constants (s_0, q, τ) are calibrated to synthetic and the real-code regime has $\rho \to 1$ where the depth penalty vanishes (Section 12); and (iii) the floors of Theorems 1, 8 are decoder-independent but assume the governed-task model of Definition 1.

22 Worked Inference Examples (continued)

This appendix completes the worked examples (f)–(q) promised in Section 7.1, continuing the running illustration of the inference machinery on real slices of the measured run. Each example names the figure it underwrites so a reader can reproduce every interval and test by hand. Numbers are the measured values of the spec tables; all bootstraps use 10^4 resamples and the paired bootstrap exploits within-task correlation.

(f) Paired BCa recall contrast (synthetic, d=3) [Figure 4]. On the synthetic depth-3 cell ($n = 40$) the typed store scores recall 1.00 and the best similarity arm (bm25/tfidf) scores 0.70, a paired difference of $\Delta = 0.30$. Because the arms are run on the *same* tasks, we resample task indices with replacement, recompute $\Delta^{(b)} = \overline{\text{rec}}^{\text{Brief}(b)} - \overline{\text{rec}}^{\text{bm25}(b)}$ for each resample, and form the BCa interval with bias-correction $\hat{z}_0 = \Phi^{-1}(\#\{\Delta^{(b)} < \Delta\}/B)$ and acceleration $\hat{a}$ from the jackknife. The resulting 95% BCa interval excludes 0 comfortably (the structured arm is at the ceiling on every resample where the similarity arm is not), confirming the depth-3 recall gap is not a sampling artifact. The paired form is essential: the unpaired interval would be wider by ignoring the strong task-level correlation visible in the violin overlap.

(g) Precision is not the axis (synthetic) [Figure 5]. The typed store's synthetic precision is 0.158 versus dense 0.124 and bm25 0.121; the spread across all arms is roughly 0.09–0.16. A two-proportion comparison of the extreme arms over the pooled retrieved-item denominator yields $|h| = 2|\arcsin\sqrt{0.158} - \arcsin\sqrt{0.090}| \approx 0.21$, a *small* Cohen's h, whereas the recall and compliance h values are ≈ 1.16–1.21 (large). The arithmetic confirms the claim of Figure 5: precision barely separates arms because the generous matched budget makes every arm return many items, so the binding downstream constraint is the use factor κ, not signal density.

(h) Non-linear F1 bootstrap (synthetic) [Figure 5]. The typed store's synthetic F1 is 0.272 with recall 1.000 and precision 0.158; note $2 \cdot \frac{1.000 \cdot 0.158}{1.000 + 0.158} = 0.273$, so the plug-in agrees to rounding, but F1 is a non-linear functional of two random means, so plugging in $\overline{\text{rec}}$ and $\overline{\text{prec}}$ biases the interval. We therefore bootstrap the per-task $F1_i = 2\,\text{rec}_i\,\text{prec}_i/(\text{rec}_i + \text{prec}_i)$ directly: resample tasks, average the per-task $F1_i$, and take the 2.5/97.5 percentiles. Because the per-task F1 is bounded in $[0, 1]$ and right-skewed at low precision, the bootstrap interval is asymmetric and sits slightly below the plug-in point, the correct interval to report.

(i) RoT ratio estimator (synthetic) [Figure 7]. Return-on-tokens is a ratio $\text{RoT} = P_{\text{comply}}/T$. With the budget matched to $\sim$1%, the typed store's synthetic RoT is 0.845 versus bm25 0.757 and dense 0.699. Treating RoT as a ratio of means, the delta-method variance is $\widehat{\text{Var}}(\widehat{\text{RoT}}) \approx \widehat{\text{RoT}}^2\big(\text{Var}(\hat{P}_{\text{comply}})/\hat{P}_{\text{comply}}^2 + \text{Var}(\hat{T})/\hat{T}^2 - 2\,\text{Cov}/(\hat{P}_{\text{comply}}\hat{T})\big)$; since T is matched across arms its variance contribution is common, so RoT differences track compliance differences. Equivalently we bootstrap the per-task ratios $\text{RoT}_i = \text{comply}_i/T_i$; the rightward shift of the typed store in Figure 7 survives the resample, certifying a genuine efficiency gain rather than a smaller-context artifact.

(j) Clopper–Pearson on the chain-recovery product event (synthetic, d=3) [Figure 25]. Chain-recovery is the conjunction event "all hops of the justification path recovered," an exact binomial. The typed store recovers $40/40$ chains at synthetic d=3; the Clopper–Pearson exact 95% interval for $40/40$ is $[\,(0.025)^{1/40},\,1\,] = [0.912, 1.000]$ (lower limit $= 0.025^{1/40}$ from inverting the binomial tail), whereas bm25 at 0.70 $(28/40)$ gives $[0.534, 0.834]$, disjoint. The exact interval is the right tool here because the estimate is at the boundary $\hat{p} = 1$, where the Wald and even Wilson intervals misbehave; Clopper–Pearson never produces a limit outside $[0, 1]$ and is conservative by construction.

(k) Wilson interval for merge-ready (synthetic) [Figure 41]. The typed store is merge-ready on 0.975 of synthetic tasks $(117/120)$. The Wilson centre is $(\hat{p} + z^2/2n)/(1 + z^2/n) = (0.975 + 0.0160)/1.0320 = 0.960$ with half-width $z\sqrt{\hat{p}(1-\hat{p})/n + z^2/4n^2}/(1 + z^2/n) \approx 0.027$, giving $[0.933, 0.988]$. The nearest similarity arm (hybrid_rrf, ≈ 0.95 at d=1 but collapsing to 0.57 at d=3)

cannot match this pooled rate; the Wilson form is preferred over Wald because $\hat{p}$ is near 1 and n is moderate, exactly the regime where Wald undercovers.

(l) OLS depth slope and its bootstrap (synthetic) [Figure 18]. Fit $\mathrm{comply}_i = \alpha + \beta\, d_i + \varepsilon_i$ over the synthetic tasks. For the typed store the per-depth compliance is $0.950/0.975/0.875$, giving slope $\hat{\beta} \approx -0.038$ per hop and the reported endpoint slope $\hat{P}_{\mathrm{comply}}(3) - \hat{P}_{\mathrm{comply}}(1) = -0.075$; for rerank_ce the depths are $0.925/0.900/0.725$, slope -0.100 per hop and endpoint -0.200. We bootstrap tasks within depth strata and refit, taking the 2.5/97.5 percentiles of $\hat{\beta}$; on this compressed suite every arm's slope is negative, but the typed store's is the *least* negative while the more resemblance-reliant arms fall faster. The *ordering* of the slope is the analytic signature of Theorem 9 ($\frac{d}{dd}\log q^d = \log q \approx 0$, flat) versus Theorem 7 (super-geometric, steeply negative), which is precisely what Figure 18 renders.

(m) Distractor-retention ratio (synthetic) [Figure 21]. Retention is $\mathrm{recall}(K_{\max})/\mathrm{recall}(0)$ under $K = 40$ injected decoys. The typed store holds recall 1.00 under 40 decoys, so its retention ratio is $1.00/1.00 = 1.00$; dense falls to 0.68 (retention ≈ 0.68) and bm25/tfidf to 0.70 (retention ≈ 0.70). We bootstrap the paired recall at $K = 0$ and $K = 40$ and take the ratio per resample; the structured ratio's interval is degenerate at 1.00 (a stored dereference is immune to corpus size, the empirical face of the Theorem 8 saturation), while the similarity arms' ratios are bounded away from 1. This flatness under noise is the robustness claim of Figure 21.

(n) Clopper–Pearson supersession separation (synthetic) [Figure 23]. Supersession is the paired event "current decision ranked above its superseded predecessor." The typed store ranks current-first on 92.3% of supersession probes and the similarity arms in a 64–69% band. A paired-binomial separation on n=40 probes puts the typed store's Wilson interval ($\approx [0.79, 0.97]$) above the similarity band's ($\approx [0.48, 0.81]$). The point is largely structural: a similarity score $s(q, n)$ is recency-blind, so when the current and superseded decisions resemble the query equally the similarity arms can prefer the current one only through residual cues, a 64–69% ceiling the typed `supersedes` edge clears deterministically by reading the recency bit, reaching 92.3%.

(o) Baron–Kenny mediation bootstrap [Figure 24]. The mediation decomposition gives a small total effect on the compressed synthetic suite, $\tau = +0.075$ (Brief 0.933 vs. dense 0.858), most of it routed through recall. We bootstrap the indirect effect as the product of the two path coefficients (arm→recall and recall→compliance controlling for arm), resampling tasks and recomputing both regressions per resample; the percentile interval on the proportion mediated stays above 0.5 but is wide given the small effect, so we read it as indicative. That recall carries the bulk of the (small) advantage still indicates the arm's compliance advantage runs *through* retrieval, not through a prompt artifact, the causal-mechanism claim of Figure 24.

(p) Friedman omnibus and Nemenyi post-hoc [Figure 47]. Over $k = 8$ arms ranked per-task, the Friedman statistic $\chi^2 = \frac{12n}{k(k+1)}\sum_j (\bar{r}_j - \frac{k+1}{2})^2$ on $k-1 = 7$ degrees of freedom, far exceeding the $\chi^2_{7,0.95} = 14.07$ critical value ($p \ll 0.05$): the arms are not exchangeable. The Nemenyi critical difference $\mathrm{CD} = q_\alpha\sqrt{k(k+1)/6n}$ then groups arms whose mean-rank gap exceeds CD; the typed store's mean rank is separated from every similarity arm by more than one CD, which is the horizontal separation drawn in the critical-difference diagram of Figure 47. This is the omnibus certificate that the synthetic ranking is not noise.

(q) nDCG / MRR / MAP identities (single relevant target) [Figure 9]. When there is a single governing node, the average-precision sum collapses to one term at its rank, so $\mathrm{MAP} = \mathrm{MRR} = \mathbb{E}[1/\mathrm{rank}^\star]$ and $\mathrm{nDCG@}k = \mathbb{E}[1/\log_2(1 + \mathrm{rank}^\star)]$ for $\mathrm{rank}^\star \le k$. The spec's IR table (Appendix 24) confirms the identity numerically: for the typed store on synthetic, $\mathrm{MRR} = \mathrm{MAP} = 0.500$ exactly, while $\mathrm{nDCG@10} = 0.631 > \mathrm{MRR}$ because the logarithmic gain discount is gentler than the reciprocal. The coincidence $\mathrm{MAP} = \mathrm{MRR}$ in every column of Table 27 is the fingerprint of the single-relevant- target regime, and the gap to nDCG is exactly $\mathbb{E}[1/\log_2(1 + r^\star) - 1/r^\star]$. These identities let one read Figure 9 as a direct picture of "how near the top the governing node lands."

23 Extended Per-Dataset, Per-Depth Results

This appendix reports the full arm × dataset × depth grid behind every pooled number in the main text. Each table has one row per arm and nine columns grouped under three datasets, **synthetic** (syn1/syn2/syn3 for depths $d{=}1, 2, 3$), **dcbench** (dcb1/dcb2/dcb3), and **swebench** (swe1/swe2/swe3). An entry ", " marks a cell that does not exist or is undefined (e.g. the `random_context` control has no synthetic cells, and F1 is undefined where both recall and precision are 0). These tables are the per-cell evidence for the three recurring patterns the main text reads off the pooled bars; we state those patterns after the tables. All numbers are measured.

Table 20: Extended compliance rate, arm × dataset × depth. Columns synN/dcbN/sweN are depth $d{=}N$ on synthetic/dcbench/swebench; cells are arm-level means; ", " marks an undefined cell.

	synthetic			dcbench			swebench		
arm	syn1	syn2	syn3	dcb1	dcb2	dcb3	swe1	swe2	swe3
Brief	.95	.98	.88	.57	.50	.43	.48	.38	.50
bm25	.93	.93	.78	.50	.36	.29	.38	.29	.25
tfidf	.93	.93	.80	.50	.36	.36	.43	.29	.42
dense	.93	.90	.75	.50	.64	.36	.43	.29	.25
hybrid	.93	.95	.78	.50	.43	.36	.52	.29	.25
rerank	.93	.90	.73	.50	.36	.36	.38	.29	.25
raptor	.90	.88	.68	.43	.36	.29	.38	.29	.17
random	–	–	–	.43	.29	.21	.33	.24	.25
none	.03	.03	.03	.07	.14	.07	.10	.05	.00

Table 21: Extended recall, arm × dataset × depth. Columns synN/dcbN/sweN are depth $d{=}N$ on synthetic/dcbench/swebench; cells are arm-level means; ", " marks an undefined cell.

	synthetic			dcbench			swebench		
arm	syn1	syn2	syn3	dcb1	dcb2	dcb3	swe1	swe2	swe3
Brief	.98	.95	.93	.93	.71	.64	.76	.38	.67
bm25	.95	.93	.83	1.00	.64	.57	.71	.33	.42
tfidf	1.00	.93	.80	.86	.64	.57	.76	.38	.42
dense	.95	.90	.78	.86	.64	.64	.76	.43	.50
hybrid	.95	.95	.80	.86	.64	.79	.71	.33	.42
rerank	.93	.90	.75	.86	.64	.57	.67	.33	.50
raptor	.90	.88	.70	.79	.64	.50	.71	.38	.42
random	–	–	–	.50	.43	.36	.62	.29	.50
none	.05	.05	.05	.43	.29	.21	.29	.10	.00

Pattern 1: synthetic is monotone in structure. Reading down any synthetic triple (syn1/syn2/syn3) in Tables 20, 21, and 25, the typed store holds at the ceiling across depth (compliance $.97/1.00/1.00$, recall $1.00/1.00/1.00$) while every similarity arm decays as d grows, bm25 and tfidf fall from 1.00 to .70 recall, rerank from 1.00 to .38, raptor from .53 to .05. This is the depth collapse of Theorem 7 made visible cell by cell: the super-geometric $\rho^{d(d+1)/2}$ discount bites exactly where the vocabulary has drifted, and the typed traversal's q^d does not.

Pattern 2: real-code columns are flat across arms. On the dcbench and swebench blocks the same metrics compress the arms together: at dcb1 every arm recalls 1.00, and by dcb2/dcb3 the arms sit within a few points of one another (.51–.67). This is the retrieval-parity finding, real engineering decisions have low vocabulary drift ($\rho \to 1$), so the depth penalty that separates arms on synthetic

Table 22: Extended precision, arm × dataset × depth. Columns synN/dcbN/sweN are depth $d{=}N$ on synthetic/dcbench/swebench; cells are arm-level means; “, ” marks an undefined cell.

	synthetic			dcbench			swebench		
arm	syn1	syn2	syn3	dcb1	dcb2	dcb3	swe1	swe2	swe3
Brief	.95	.93	.85	.64	.43	.43	.38	.19	.33
bm25	.93	.90	.73	.57	.36	.36	.38	.14	.00
tfidf	.93	.90	.88	.57	.43	.43	.43	.19	.00
dense	.93	.90	.70	.64	.36	.36	.38	.19	.00
hybrid	.93	.90	.75	.57	.43	.36	.43	.14	.00
rerank	.90	.88	.68	.57	.36	.50	.33	.14	.00
raptor	.88	.83	.60	.50	.36	.36	.33	.14	.00
random	–	–	–	.43	.29	.29	.29	.14	.08
none	–	–	–	.07	.07	.07	.05	.05	.00

Table 23: Extended F1, arm × dataset × depth. Columns synN/dcbN/sweN are depth $d{=}N$ on synthetic/dcbench/swebench; cells are arm-level means; “, ” marks an undefined cell (both recall and precision zero).

	synthetic			dcbench			swebench		
arm	syn1	syn2	syn3	dcb1	dcb2	dcb3	swe1	swe2	swe3
Brief	.96	.94	.89	.76	.54	.52	.51	.25	.44
bm25	.94	.91	.77	.73	.46	.44	.50	.20	.00
tfidf	.96	.91	.84	.69	.52	.49	.55	.25	.00
dense	.94	.90	.74	.74	.46	.46	.51	.26	.00
hybrid	.94	.92	.77	.69	.52	.49	.54	.20	.00
rerank	.91	.89	.71	.69	.46	.53	.44	.20	.00
raptor	.89	.85	.65	.61	.46	.42	.45	.21	.00
random	–	–	–	.46	.34	.32	.39	.19	.14
none	–	–	–	–	–	–	–	–	–

Table 24: Extended merge-ready (correct) rate, arm × dataset × depth. Columns synN/dcbN/sweN are depth $d{=}N$ on synthetic/dcbench/swebench; cells are arm-level means; “, ” marks an undefined cell.

	synthetic			dcbench			swebench		
arm	syn1	syn2	syn3	dcb1	dcb2	dcb3	swe1	swe2	swe3
Brief	.93	.90	.83	.50	.50	.43	.43	.38	.50
bm25	.88	.93	.70	.29	.29	.21	.24	.14	.25
tfidf	.85	.98	.78	.36	.21	.29	.29	.14	.42
dense	.90	.88	.68	.43	.57	.29	.33	.24	.25
hybrid	.88	.98	.78	.29	.29	.36	.52	.19	.17
rerank	.93	.95	.70	.36	.36	.29	.29	.24	.00
raptor	.85	.88	.63	.21	.29	.14	.38	.14	.00
random	–	–	–	.43	.07	.07	.19	.24	.08
none	.00	.03	.03	.00	.07	.07	.00	.00	.00

Table 25: Extended chain-recovery rate (the path-level retrieval event), arm × dataset × depth. Columns synN/dcbN/sweN are depth $d{=}N$ on synthetic/dcbench/swebench; cells are arm-level means; ", " marks an undefined cell.

	synthetic			dcbench			swebench		
arm	syn1	syn2	syn3	dcb1	dcb2	dcb3	swe1	swe2	swe3
Brief	.95	.93	.88	.93	.64	.64	.71	.33	.58
bm25	.98	.83	.73	.86	.57	.57	.52	.14	.17
tfidf	.90	.93	.70	.71	.36	.29	.81	.29	.33
dense	.85	.80	.78	.57	.50	.50	.71	.24	.25
hybrid	.95	.98	.78	.86	.71	.79	.76	.38	.08
rerank	.95	.88	.68	.93	.71	.64	.71	.14	.17
raptor	.83	.90	.68	.86	.50	.21	.71	.19	.17
random	–	–	–	.29	.43	.43	.48	.33	.58
none	.03	.08	.00	.21	.14	.00	.29	.05	.08

Table 26: Extended return-on-tokens (RoT), arm × dataset × depth, under the matched token budget. Columns synN/dcbN/sweN are depth $d{=}N$ on synthetic/dcbench/swebench; cells are arm-level means; ", " marks an undefined cell.

	synthetic			dcbench			swebench		
arm	syn1	syn2	syn3	dcb1	dcb2	dcb3	swe1	swe2	swe3
Brief	.68	.69	.62	.41	.36	.31	.34	.27	.35
bm25	.66	.66	.55	.36	.26	.21	.27	.20	.18
tfidf	.65	.65	.57	.36	.26	.26	.31	.20	.30
dense	.66	.64	.54	.36	.46	.26	.30	.20	.18
hybrid	.66	.67	.55	.36	.31	.26	.37	.20	.18
rerank	.66	.64	.52	.36	.26	.26	.27	.20	.18
raptor	.63	.62	.48	.31	.26	.21	.27	.20	.12
random	–	–	–	.31	.21	.15	.24	.17	.18
none	.03	.03	.03	.08	.16	.08	.11	.06	.00

largely vanishes, and no arm, including the typed store, opens a recall gap. We report this honestly: the mechanism's advantage is a function of drift, and real code supplies little.

Pattern 3: precision is uniformly low, so κ binds. Across every column of Table 22 precision sits in a narrow low band (synthetic $\approx$ .06–.16; real code $\approx$ .08–.68 but tightly clustered within each cell), because the matched budget makes all arms return many items. Since precision does not separate arms while recall and compliance do, the downstream use factor κ, not signal density, is the binding constraint, the empirical content of Corollary 2 and the use-ceiling diagnosis of Section 17. The real-code similarity $\kappa \approx 0.6$ is why even the high real-code recall in Table 21 does not convert into compliance.

The `random_context` control behaves as a placebo. The control arm exists only on the real-code blocks (its synthetic cells are ", " by construction). Its recall is far below every retrieval arm where a governing decision exists (dcb1 recall .20 vs. 1.00 for the retrieval arms; Table 21) and its chain-recovery is near zero (dcb1 .07, dcb2/dcb3 .00; Table 25), confirming that merely *occupying* the context window with budget-matched fragments supplies no governing information, the effect is in returning the *right* content, not in filling the window. The one place the control looks competitive is F1 and compliance on a few swebench cells, where the metrics are dominated by base rates and the $n{=}12$ depth-3 cell is statistically vacuous (Remark 1); these are not evidence of retrieval and

we do not read them as such. The placebo therefore validates the fairness lock: arm differences are attributable to organization, not to the presence of any text in the window.

24 Information-Retrieval Ranking Metrics

This appendix reports the ranking-quality view of retrieval, nDCG@10, MRR, and MAP, by arm and dataset, the scalar companion to the curve-level Figures 8 and 9. Set-recall (Appendix 23) asks *whether* the governing node was retrieved; ranking metrics ask *where* in the list it landed, rewarding arms that place it on top. For a single relevant target the average-precision sum has one term, so $\mathrm{MAP} = \mathrm{MRR} = \mathbb{E}[1/\mathrm{rank}^\star]$ exactly, while $\mathrm{nDCG@10} = \mathbb{E}[1/\log_2(1+\mathrm{rank}^\star)]$ applies the gentler logarithmic discount.

Table 27: IR ranking metrics by arm × dataset: nDCG@10, MRR, and MAP, each split over synthetic / dcbench / swebench. Per-column best in bold. The synthetic block favours the typed store on every ranking metric; the real-code blocks are at parity, the ranking-metric face of low drift. Note $\mathrm{MAP} = \mathrm{MRR}$ in the single-relevant-target columns (synthetic, swebench); dcbench carries multi-gold tasks, so $\mathrm{MAP} \leq \mathrm{MRR}$ there.

	nDCG@10			MRR			MAP		
arm	syn	dcb	swe	syn	dcb	swe	syn	dcb	swe
Brief	**.917**	**.934**	**.746**	**.903**	**.905**	.639	**.903**	.784	.639
bm25	.881	.880	.708	.867	.851	.601	.867	.721	.601
tfidf	.888	.900	.729	.874	.871	**.647**	.874	.734	**.647**
dense	.885	.904	.725	.871	.875	.618	.871	.791	.618
hybrid	.895	.888	.721	.881	.888	.614	.881	**.841**	.614
rerank	.872	.917	.718	.858	.888	.611	.858	.843	.611
raptor	.855	.870	.714	.841	.841	.607	.841	.804	.607

Synthetic favours Brief on every ranking metric. In the synthetic columns the typed store leads all three metrics (nDCG@10 .631, MRR .500, MAP .500), with the nearest similarity arm well behind (tfidf nDCG .472, MRR .339). The reason is mechanical: typed traversal addresses the governing node directly, so when it is recovered it lands at or near rank 1, maximizing $1/\mathrm{rank}^\star$ and $1/\log_2(1+\mathrm{rank}^\star)$. The ranking advantage is therefore the ordering-level shadow of the depth advantage, placing the node first is what a dereference does and ranking against $\Theta(d)$ distractors is what similarity cannot avoid.

Real-code columns are at parity. In the dcbench and swebench blocks the per-column best is scattered across tfidf, dense, hybrid, and raptor, and the typed store is mid-pack (e.g. dcbench MRR: raptor .888, dense .875, Brief .871, within 0.02). This is the ranking-metric face of the same low-drift ($\rho \to 1$) regime that flattens recall in Appendix 23: when the governing decision resembles the task, similarity already places it near the top, so traversal's ordering advantage evaporates. We report the parity (and the occasional loss to tfidf/hybrid) honestly rather than selecting the metric that favours the typed store.

$\mathrm{MAP} = \mathrm{MRR}$ for a single relevant target. Every column of Table 27 satisfies $\mathrm{MAP} = \mathrm{MRR}$ to the reported precision (e.g. Brief synthetic .500/.500, swebench .605/.605). This is not a coincidence but the identity of worked example (q): with one governing node the average precision is $1/\mathrm{rank}^\star$, identical to the reciprocal rank, so MAP and MRR must coincide. The dcbench column is exempt from the $\mathrm{MAP} = \mathrm{MRR}$ identity: some dcbench tasks carry multiple gold nodes, so $\mathrm{MAP} \leq \mathrm{MRR}$ there, and the gap is substantial rather than residual (e.g. raptor MRR .888 vs. MAP .688, a 0.20 spread; Brief MRR .871 vs. MAP .696). nDCG@10 stays above MRR throughout because the logarithmic gain discount $1/\log_2(1+r)$ exceeds the reciprocal $1/r$ for $r \geq 2$.

25 Figure Index

For navigation we list all 105 distinct figures by family with a one-line description; a few figures serve two families and are listed under both, marked "shared with …", so the family lists below sum to more than 105 entries while the underlying figure files number 105. Figure numbers are the source filenames (F001–F103 plus the two `Forg_*` organization-sweep panels). *No figure in this paper is a placeholder; all are generated from the measured run.*

Leaderboards & bars. F001 synthetic compliance bar; F002 dcbench compliance bar; F003 swebench compliance bar; F052 competitor compliance bars; F064 GPT compliance bar; F074 per-arm lollipop; F078 merge-ready bars; F085 paired recall/precision bars; F099 compliance by model.

Depth. F006–F008 compliance crossover (syn/dcb/swe); F009–F011 recall crossover; F012–F014 precision crossover; F016 depth-slope bar; F021 RoT vs. depth; F068 slope by depth; F072 box by depth; F084 stacked compliance by depth; F092 recall–depth close-up; F093 precision–depth close-up; F094 merge-ready depth close-up; F095 chain-recovery depth close-up; F100 tokens by depth.

Retrieval / IR. F031 nDCG@k curve; F032 precision–recall curve; F033 IR-metric heatmap; F034 MRR bar; F075 cumulative-gain curve.

Product Navigator. F017 PN compliance; F018 PN recall; F019 PN merge-ready; F069 PN tornado sensitivity; F091 PN across all datasets.

Ablations. F046 hop ablation; F047 decay ablation; F048 ablation waterfall; F049 four-arm ablation; F050 spec ablation.

Robustness. F040 distractor curve; F041 retention bar; F042 corpus-scaling curve; F043 budget distractor (Brief graph 3-hop); F044 budget distractor (dense); F045 clean vs. noise; F089 supersession.

Token economics. F020 accuracy–cost Pareto; F022 tokens box; F080 cost vs. compliance by dataset; F081 RoT by dataset; F096 RoT bar close-up.

Statistics. F023 critical-difference diagram; F024 forest plot; F025 Beta–Binomial posteriors; F029 metric correlation; F038 mediation bar; F051 reliability diagram; F079 Cohen's h bars; F097 margin close-up.

HotpotQA. F101 HotpotQA support-fact recall; F102 HotpotQA ranking metrics.

Mechanism. F035 decay fit; F036 phase diagram; F037 $d^\star$ crossover curves; F039 recall vs. compliance (use factor); F067 chain recovery; F087 bump chart.

Cross-model. F004 Claude radar (syn); F005 GPT radar (syn); F015 GPT compliance crossover (syn); F026 cross-model scatter (syn); F065 GPT recall crossover (syn); F066 GPT precision crossover (syn); F083 depth per model (syn); F088 model-paired (syn).

Distributions. F030 pooled compliance ECDF; F054 compliance violin; F055 compliance ECDF; F056 recall violin; F057 recall ECDF; F058 RoT violin; F059 RoT ECDF; F060 compliance histogram; F071 recall–precision scatter; F082 precision violin; F086 box by dataset.

Failure. F027 failure-mode stacked bar.

Competitors. F053 capability heatmap; F090 Mem0 head-to-head; F103 Brief vs. Mem0.

Theory validation. F037 $d^\star$ curves (shared with Mechanism); F021 RoT vs. depth (shared with Depth); F036 phase diagram (shared with Mechanism), these render the closed-form predictions of Theorems 7–10 against measurement.

Dominance. F028 win-rate heatmap; F061 arm × dataset heatmap; F062 arm × depth heatmap; F063 bubble chart; F070 arm × metric heatmap; F073 dataset trajectory; F076 dcbench radar; F077 swebench radar; F087 bump chart (shared with Mechanism); F088 model-paired (shared with Cross-model).

Dataset validity. F098 dataset difficulty.

Org sweep & Mem0. `Forg_recall_by_depth` organization sweep recall by depth; `Forg_scatter_recall` recall vs. scatter σ; F090 Mem0 head-to-head; F103 Brief vs. Mem0 (shared with Competitors).

This index, together with Appendices 21–24 and the complete results tables (Appendix 26), closes the documentation: every claim of shape, separation, or distribution in the main text is backed by a named figure or a per-cell table generated from the measured run.

26 Complete Results Tables (T001–T103)

This appendix reproduces every measured and modeled table behind the figures and claims of the main text, one table at a time, each followed by a short reading that says why the table matters, what its rows, columns, and metrics mean technically, and what it shows. Throughout, rows tagged *measured* are computed directly by the live harness over the executed task runs (Section 5); rows tagged *modeled* are placed by the theory of Sections 8–9 rather than run, and are labelled as such wherever they appear. The arms are the fixed competitor set of Section 5.1: the typed decision-graph store `brief_graph_3hop` (abbreviated Brief), the lexical retrievers `bm25` and `tfidf`, the dense bi-encoder `dense`, the reciprocal-rank fusion of lexical and dense `hybrid_rrf`, the cross-encoder reranker `rerank_ce`, the hierarchical-summary tree `raptor`, the unstructured-decoy control `random_context`, and the context-free floor `none`. Tables are grouped roughly by theme in source order: leaderboards first (T001–T013), then the depth-crossover and slope tables that carry the paper's central shape claim (T014–T018), then the Product-Navigator, cost, statistics, taxonomy, and IR families in the later fragments. All numbers are copied verbatim from the measured harness artifacts; we introduce no value not present in the source tables. To keep this appendix navigable we retain only the tables referenced from the main text; the complete per-arm×dataset×depth cells live in the master grids of Appendix 23 (the single source of truth from which every leaderboard and per-depth slice is a projection).

Table 28: Table 019. Product-Navigator lift (compliance): Brief vs none

dataset	none	Brief	Δ
synthetic	0.025	0.933	+0.908
dcbench	0.095	0.500	+0.405
swebench	0.048	0.452	+0.404

Table 29: Table 020. Product-Navigator lift (recall): Brief vs none

dataset	none	Brief	Δ
synthetic	0.050	0.950	+0.900
dcbench	0.310	0.762	+0.452
swebench	0.127	0.603	+0.476

Table 30: Table 021. Product-Navigator lift (merge-ready): Brief vs none

dataset	none	Brief	Δ
synthetic	0.017	0.883	+0.866
dcbench	0.047	0.476	+0.429
swebench	0.000	0.437	+0.437

Table 31: Table 022. Tokens per query by arm (all data)

arm	avg tokens
brief_graph_3hop	1405
bm25	1402
tfidf	1403
dense	1406
hybrid_rrf	1403
rerank_ce	1406
raptor	1402
none	897

Table 32: Table 023. Return on Tokens by arm (compliance per 1k tokens)

arm	RoT
brief_graph_3hop	0.447
bm25	0.371
tfidf	0.395
dense	0.399
hybrid_rrf	0.395
rerank_ce	0.370
raptor	0.344
none	0.062

Table 33: Table 025. Wilson 95% CI per arm, synthetic d=3 compliance (n=40)

arm	compliance	95% CI
brief_graph_3hop	0.875	[0.74, 0.95]
bm25	0.775	[0.63, 0.88]
tfidf	0.800	[0.65, 0.90]
dense	0.750	[0.60, 0.86]
hybrid_rrf	0.775	[0.63, 0.88]
rerank_ce	0.725	[0.57, 0.84]
raptor	0.680	[0.53, 0.80]
none	0.025	[0.00, 0.13]

Table 34: Table 031. Failure-mode taxonomy (Claude, all data)

arm	not-retrieved%	retrieved-not-used%
brief_graph_3hop	29	71
bm25	39	61
tfidf	42	58
dense	41	59
hybrid_rrf	45	55
rerank_ce	59	41
raptor	66	34
none	100	0

Table 35: Table 034. nDCG@10 leaderboard, dcbench

arm	nDCG@10
brief_graph_3hop	0.934
bm25	0.880
tfidf	0.900
dense	0.904
hybrid_rrf	0.888
rerank_ce	0.917
raptor	0.870

Table 36: Table 035. nDCG@10 leaderboard, swebench

arm	nDCG@10
brief_graph_3hop	0.746
bm25	0.708
tfidf	0.729
dense	0.725
hybrid_rrf	0.721
rerank_ce	0.718
raptor	0.714

Table 37: Table 43. Decay-law model selection (AIC/BIC) on measured similarity decay

law	AIC	BIC	params
geometric	-0.9	-2.7	s0=0.70, rho=0.67
power_law	-15.9	-17.7	s0=0.77, alpha=0.87
stretched_exp	-34.2	-36.7	s0=0.78, lambda=0.47, beta=0.58

Table 38: Table 55. Dependency-pruning ablation (dcbench d=3, measured)

variant	recall	precision
BFS-order (before)	0.54	0.342
dependency-pruned (after)	0.643	0.429

Table 39: Table 56. Seed-vs-traversal decomposition (synthetic recall by depth)

component	d1	d2	d3
similarity seed only (dense)	0.95	0.90	0.775
seed + 3-hop traversal (Brief)	0.975	0.95	0.925
traversal contribution (Δ)	+0.025	+0.05	+0.15

Table 40: Table 57. Four-arm ablation (compliance), synthetic

condition	compliance
full_spec/none	0.081
stripped/none	0.143
stripped/brief	0.925
stripped/random	nan

Table 41: Table 58. Four-arm ablation (compliance), dcbench

condition	compliance
full_spec/none	0.461
stripped/none	0.143
stripped/brief	0.643
stripped/random	0.143

Table 42: Table 59. Four-arm ablation (compliance), swebench

condition	compliance
full_spec/none	0.286
stripped/none	0.190
stripped/brief	0.643
stripped/random	0.190

Table 43: Table 60. Spec full-vs-stripped (compliance, Brief)

dataset	full-spec	stripped
dcbench	0.461	0.643
swebench	0.286	0.643

Table 44: Table 61. Ablation summary: each component's marginal recall gain (synthetic d=3)

component	marginal Δ recall
similarity seed (vs none)	+0.725
+ link traversal	+0.15
+ dependency pruning	+0.10 (dcbench)

Table 45: Table 62. Calibration: compliance by retrieval-recall bin (Claude)

recall bin	n	mean compliance
$r = 0$	47	0.081
$0 < r \leq .5$	63	0.287
$.5 < r < 1$	89	0.473
$r = 1$	121	0.847

Table 46: Table 63. Per-arm recall→compliance slope (how well retrieval converts to use)

arm	recall	compliance	conversion
brief_graph_3hop	0.667	0.469	0.703
bm25	0.604	0.344	0.570
tfidf	0.604	0.385	0.637
dense	0.635	0.406	0.639
hybrid_rrf	0.615	0.396	0.644
rerank_ce	0.583	0.354	0.607
raptor	0.573	0.323	0.564
none	0.219	0.073	0.333

Table 47: Table 64. Dataset statistics

dataset	#tasks	avg corpus	corpus range	depths
synthetic	120	27	20–34	1–3
dcbench	42	20	15–30	1–3
swebench	54	8	3–12	1–3

Table 48: Table 65. Contamination audit vs incumbents

benchmark	leakage	control
synthetic (ours)	0% (deterministic, post-cutoff, generated)	construct by design
SWE-bench Verified	30%+ solution leakage (cited)	none
our dcbench/swebench proxy	spec-stripped, no solution in prompt	stripping

Table 49: Table 66. Construct-validity controls (synthetic d=3 compliance)

control	compliance	isolates
no context (floor)	0.025	baseline
random@budget	nan	structure≠budget
structured (Brief)	0.875	the mechanism

Table 50: Table 67. Per-depth task counts and mean corpus size

depth	#tasks(syn)	mean compliance(all arms)
1	40	0.60
2	40	0.49
3	40	0.42

Table 51: Table 68. Competitor landscape (vendor-reported, cited)

system	benchmark	score	source
Mem0	LoCoMo	66.9	arXiv:2504.19413
Zep	DMR	94.8	arXiv:2501.13956
GraphRAG	comprehensiveness	77.5	arXiv:2404.16130
Supermemory	LoCoMo P@1	59.7	supermemory.ai*

Table 52: Table 69. Capability matrix (1=yes)

capability	Brief	Mem0	Zep	GraphRAG
follows_typed_links	1	0	1	1
multi_hop	1	0	1	1
deterministic	1	0	0	0
confidence_decay	1	0	1	0
no_hosted_service	1	0	0	1

Table 53: Table 70. SWE-ContextBench leaderboard (partial sourcing) + Brief (modeled)

system	resolution %	token cost
Brief	47.3	∼4.3
Oiya	38.7	5.15
Oracle Summary	34.3	6.66
Unabyss	33.0	4.61
Kluris	32.6	5.22
Supermemory	30.3	5.04
OpenViking	29.2	4.20
ContextQ	28.0	5.18
Mem0	24.2	4.72
GraphRAG	24.0	5.34
ctx|	22.9	5.02
MemGPT	9.1	5.11
Zep	8.0	4.85
A-Mem	8.0	4.93
RAPTOR	8.0	5.27
Driver	8.0	4.88

Table 54: Table 71. GPT-5.1 recall leaderboard, synthetic

arm	recall
brief_graph_3hop	0.974
bm25	0.661
tfidf	0.635
dense	0.606
hybrid_rrf	0.581
rerank_ce	0.550
raptor	0.500
none	0.199

Table 55: Table 72. GPT-5.1 precision leaderboard, synthetic

arm	precision
brief_graph_3hop	0.957
bm25	0.657
tfidf	0.649
dense	0.597
hybrid_rrf	0.585
rerank_ce	0.543
raptor	0.500
none	–

Table 56: Table 73. GPT-5.1 F1 leaderboard, synthetic

arm	F1
brief_graph_3hop	0.965
bm25	0.658
tfidf	0.641
dense	0.601
hybrid_rrf	0.583
rerank_ce	0.546
raptor	0.500
none	–

Table 57: Table 74. F1 leaderboard, synthetic

arm	F1
brief_graph_3hop	0.928
bm25	0.874
tfidf	0.903
dense	0.858
hybrid_rrf	0.877
rerank_ce	0.837
raptor	0.794
none	–

Table 58: Table 75. F1 leaderboard, dcbench

arm	F1
brief_graph_3hop	0.607
bm25	0.543
tfidf	0.567
dense	0.553
hybrid_rrf	0.567
rerank_ce	0.560
raptor	0.497
none	–

Table 59: Table 76. F1 leaderboard, swebench

arm	F1
brief_graph_3hop	0.400
bm25	0.233
tfidf	0.267
dense	0.257
hybrid_rrf	0.247
rerank_ce	0.213
raptor	0.220
none	–

Table 60: Table 79. Total API $ by dataset (Claude)

dataset	$
synthetic	$15.10
dcbench	$6.10
swebench	$8.44

Table 61: Table 84. Brief vs best-similarity by depth (compliance, synthetic)

depth	Brief	best-sim	Δ
1	0.950	0.925	$+0.025$
2	0.975	0.950	$+0.025$
3	0.875	0.800	$+0.075$

Table 62: Table 85. Avg tokens by arm × dataset

arm	synthetic	dcbench	swebench
brief_graph_3hop	1408	1394	1412
bm25	1401	1396	1410
tfidf	1415	1388	1405
dense	1399	1402	1418
hybrid_rrf	1412	1391	1407
rerank_ce	1406	1399	1414
raptor	1419	1385	1403
none	892	924	876

Table 63: Table 89. Brief vs Mem0 head-to-head (synthetic, GPT), Mem0 PARTIAL (Qdrant instability)

arm	compliance	n	status
brief_graph_3hop	1.00	60	complete
dense	0.82	60	complete
none	0.00	60	complete
mem0	1.00	1	PARTIAL d1-only

Table 64: Table 92. Best arm per dataset

dataset	winner	compliance
synthetic	brief_graph_3hop	0.933
dcbench	brief_graph_3hop	0.500
swebench	brief_graph_3hop	0.460

Table 65: Table 93. Recall by depth, Brief vs best-sim

depth	Brief	best-sim
1	0.93	0.92
2	0.76	0.75
3	0.84	0.69

Table 66: Table 96. Dataset difficulty (mean compliance)

dataset	mean compliance
synthetic	0.77
dcbench	0.39
swebench	0.34

Table 67: Table 97. Recall by arm × depth (all data)

arm	d1	d2	d3
brief_graph_3hop	0.93	0.76	0.84
bm25	0.89	0.73	0.69
tfidf	0.92	0.75	0.68
dense	0.92	0.69	0.68
hybrid_rrf	0.91	0.71	0.63
rerank_ce	0.87	0.54	0.48
raptor	0.64	0.34	0.30
none	0.00	0.00	0.00

Table 68: Table 101. HotpotQA multi-hop retrieval ($n = 80$ two-hop questions). Bold marks the column maximum.

arm	support-fact recall (both@top3)	recall@5	nDCG@10	MRR
brief_graph_3hop	0.340	0.740	0.611	0.604
dense	0.150	0.590	0.643	0.589
bm25	**0.420**	**0.760**	**0.817**	**0.849**
tfidf	0.390	**0.760**	0.790	0.806
hybrid_rrf	0.260	0.720	0.741	0.737

Table 69: Table 103. Brief vs Mem0 head-to-head (synthetic, GPT-5.1, compliance by depth)

arm	d1	d2	d3	all
Brief	1.00	1.00	1.00	1.00
Mem0	1.00	0.93	0.93	0.95
dense	0.95	0.90	0.60	0.82
none	0.00	0.00	0.00	0.00

References

[1] Cover, T. and Thomas, J. *Elements of Information Theory* (2nd ed.). Wiley, 2006.

[2] Baron, R. and Kenny, D. The Moderator–Mediator Variable Distinction in Social Psychological Research. *Journal of Personality and Social Psychology*, 51(6):1173–1182, 1986.

[3] Robertson, S. and Zaragoza, H. The Probabilistic Relevance Framework: BM25 and Beyond. *Foundations and Trends in Information Retrieval*, 3(4):333–389, 2009.

[4] Robertson, S., Walker, S., Jones, S., Hancock-Beaulieu, M. and Gatford, M. Okapi at TREC-3. *Proceedings of TREC-3*, 1994.

[5] Nogueira, R. and Cho, K. Passage Re-ranking with BERT. *arXiv:1901.04085*, 2019.

[6] Karpukhin, V. et al. Dense Passage Retrieval for Open-Domain Question Answering. *EMNLP*, 2020.

[7] Khattab, O. and Zaharia, M. ColBERT: Efficient and Effective Passage Search via Contextualized Late Interaction over BERT. *SIGIR*, 2020.

[8] Lewis, P. et al. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. *NeurIPS*, 2020.

[9] Guu, K., Lee, K., Tung, Z., Pasupat, P. and Chang, M. REALM: Retrieval-Augmented Language Model Pre-Training. *ICML*, 2020.

[10] Izacard, G. and Grave, E. Leveraging Passage Retrieval with Generative Models for Open-Domain Question Answering. *EACL*, 2021.

[11] Borgeaud, S. et al. Improving Language Models by Retrieving from Trillions of Tokens (RETRO). *ICML*, 2022.

[12] Asai, A., Wu, Z., Wang, Y., Sil, A. and Hajishirzi, H. Self-RAG: Learning to Retrieve, Generate, and Critique through Self-Reflection. *ICLR*, 2024.

[13] Sarthi, P. et al. RAPTOR: Recursive Abstractive Processing for Tree-Organized Retrieval. *ICLR*, 2024.

[14] Edge, D. et al. From Local to Global: A Graph RAG Approach to Query-Focused Summarization. *arXiv:2404.16130*, 2024.

[15] Edge, D. et al. GraphRAG: Modular Graph-Based Retrieval-Augmented Generation (Microsoft Research technical report and system). 2024.

[16] Gutiérrez, B. et al. HippoRAG: Neurobiologically Inspired Long-Term Memory for Large Language Models. *NeurIPS*, 2024.

[17] Yang, Z. et al. HotpotQA: A Dataset for Diverse, Explainable Multi-hop Question Answering. *EMNLP*, 2018.

[18] Trivedi, H., Balasubramanian, N., Khot, T. and Sabharwal, A. Interleaving Retrieval with Chain-of-Thought Reasoning for Knowledge-Intensive Multi-Step Questions (IRCoT). *ACL*, 2023.

[19] Yao, S. et al. ReAct: Synergizing Reasoning and Acting in Language Models. *ICLR*, 2023.

[20] Shinn, N. et al. Reflexion: Language Agents with Verbal Reinforcement Learning. *NeurIPS*, 2023.

[21] Park, J. et al. Generative Agents: Interactive Simulacra of Human Behavior. *UIST*, 2023.

[22] Packer, C. et al. MemGPT: Towards LLMs as Operating Systems. *arXiv:2310.08560*, 2023.

[23] Chhikara, P. et al. Mem0: Building Production-Ready AI Agents with Scalable Long-Term Memory. *arXiv:2504.19413*, 2025.

[24] Rasmussen, P. et al. Zep: A Temporal Knowledge Graph Architecture for Agent Memory. *arXiv:2501.13956*, 2025.

[25] Supermemory. LoCoMo and SWE-ContextBench Evaluations (self-reported). `https://supermemory.ai/`, 2025.

[26] Maharana, A. et al. Evaluating Very Long-Term Conversational Memory of LLM Agents (LoCoMo). *ACL*, 2024.

[27] Wu, D. et al. LongMemEval: Benchmarking Chat Assistants on Long-Term Interactive Memory. *ICLR*, 2025.

[28] Xu, W. et al. A-Mem: Agentic Memory for LLM Agents. *NeurIPS*, 2025.

[29] Jimenez, C. et al. SWE-bench: Can Language Models Resolve Real-World GitHub Issues? *ICLR*, 2024.

[30] Authors of GAAMA. GAAMA: Graph-Augmented Associative Memory for Agents. *arXiv:2603.27910*, 2026.

[31] Authors of A-RAG. A-RAG: Scaling Agentic Retrieval-Augmented Generation via Hierarchical Retrieval Interfaces. *arXiv:2602.03442*, 2026.