Eliminating Context Rot in Frozen LLMs: A Three-Mode Structural State-Coupling Architecture

Ken Morkaya
kenmorkaya@gmail.com

April 23, 2026

Abstract

We present a three-mode coupling architecture — SICD (Structural Integrated Core Dynamics), wired to a frozen off-the-shelf Gemma 4 26B (4-bit MLX Mixture-of-Experts, 128 experts with top-2 routing) — that eliminates three classes of context rot failure: plant-loss-through-truncation, enumeration-loop attention degradation, and decoder-glitch sampling pathology. On a densified long-horizon plant-recall benchmark where the prompt at the recall turn reaches ≈64,000 tokens ("64K tokens", 25% of the model's 256K-token context window), the full architecture reaches **20/20 across two independently-authored 10-case pools** with cross-pool replication (Wilson 95% CI on 20/20 pooled: [0.84, 1.00]; pre-registered in §10.13). The coupling is entirely prompt-side plus inference-time expert biasing — no LLM fine-tuning, no training signal, no weight updates.

Retrieval-augmented generation (RAG) targets a distinct problem and does not address context rot. RAG retrieves external knowledge to augment prompts, assuming conversation history remains accessible. Context rot is the failure mode where *conversation content* becomes inaccessible despite being in session history — through truncation, window pressure, or attention-degradation at long context. RAG does not target either mechanism of context rot: it does not retrieve from prior conversation turns, and by expanding prompt length with external documents it may in fact contribute to mechanism-1 attention degradation. Branch-conditioned resonance retrieval (Layer 1 in this paper) is a different mechanism class — it re-admits prior conversation turns at risk of becoming inaccessible, scored by resonance against a living structural state (a tree of branches with stress, curvature, and axis orientation) rather than by text similarity to the current query. Each of the three coupling layers addresses a specific long-context failure class:

1. **Layer 1 (branch-conditioned resonance retrieval)** rescues **context-removal rot** — the failure mode where a planted fact is forced out of the model's visible prompt. Measured on a forced single-turn-drop truncation benchmark (40 cases, `history_window=4`, N=8 turns per case): baseline accuracy 2.5%, SICD-rescued accuracy 100%, paired delta **+0.975**, 95% CI [+0.925, +1.000].
2. **State-block prelude framing (authoritative-state format)** eliminates **enumeration-loop rot** — the failure mode where the model, under a prose retrieval prelude, enumerates prior messages in its chain-of-thought and exhausts its generation budget before answering. Measured on the Variant F densified-long-context benchmark at ≈64K tokens: switching the retrieval prelude from prose to an authoritative state block with explicit precedence instruction eliminates every observed enumeration-loop failure across two independently-authored 10-case pools.

3. **Layer 2 (per-expert MoE logit bias, α=1.0)** rescues **decoder-glitch and attention-degradation rot** — failure modes where sampling at long context either produces low-probability token sequences that derail generation at the start, or where the model enters a chain-of-thought pattern that ignores the retrieved plant. Measured on the same Variant F benchmark: Layer 2 alone reaches 19/20 across the two pools versus vanilla decoding's 17/20.

The full architecture (all three mechanisms active simultaneously, condition FULL in §2) is the **only condition among the six tested to reach 20/20 across both pools**. Single-layer and two-layer subsets each leave at least one case unrescued: L2 alone = 19/20 (Layer 2 doesn't universally rescue numeric plants at this scale), L1-state alone = 19/20 (state-block prelude doesn't protect against decoder glitches), L1+L2-prose = 16/20 (prose prelude introduces composition interference with Layer 2). The Wilson CI on the pooled 20/20 extends down to 0.84 — the true pass rate could plausibly be as low as 84% under a different sample, so this is a strong preliminary result rather than a proven robustness claim.

The locked context-removal-rot rescue stands exactly as previously measured in prior work on this benchmark family; this paper extends the architectural claim to all three failure modes and adds the cross-pool replication evidence.

1 Introduction

1.1 RAG does not address context rot

Before describing SICD's architecture we draw an explicit boundary against the retrieval mechanism the paper is most likely to be confused with. Retrieval-augmented generation (RAG) [5] addresses a specific problem: the model does not know a fact because the fact is not in its weights and was never supplied in context. RAG retrieves the missing fact from an external document store, scored by text-similarity metrics (dense embeddings [3], BM25, or learned rerankers), and injects the retrieved passages into the prompt. RAG's design assumption is that **the conversation itself remains intact and accessible** — the retrieval is about pulling in *external* knowledge the prompt doesn't already have.

Context rot is a different problem. Context rot is the failure mode where *conversation content* — facts the model has already been told in earlier turns of the same session — becomes inaccessible despite being in session history. The paper's Run 10.7 measures this directly: a fact is planted at turn 1, the conversation continues for 7 more turns, a recall question is asked at turn 8, and under `history_window=4` the planted fact is no longer in the model's visible prompt. Baseline accuracy collapses to 2.5%. RAG does not address this failure mode:

- RAG retrieves from an **external** store, not from prior conversation turns.
- RAG is scored by **text similarity to the current query**, not by which earlier conversation content is structurally relevant to the current reasoning state.
- RAG **assumes** the conversation-level context is intact — the exact assumption context rot violates.
- RAG **expands prompt length** by injecting external documents; if anything, this exacerbates mechanism-1 attention degradation at long context.

SICD's Layer 1 (branch-conditioned resonance retrieval) is a different mechanism class. The **query** is not the user's text — it is the conversation's current *structural state*, a distributed representation maintained in a tree of branches that updates with every turn (formalised in §2.1). The **corpus** is not external — it is prior turns of the conversation itself, stored alongside the structural snapshot under which they were emitted. The **scoring** is cosine similarity between candidate anchor leaf-vectors and the currently-activated branches' semantic vectors, where "currently activated" is determined live by axis-alignment against the turn's projected structural load.

This paper's empirical claims are about context rot. §2 below makes the SICD mechanism concrete; §8 places it in the related-work landscape. Readers familiar with RAG should note that Layer 1's role here — re-admitting conversation content that has left the window — is orthogonal to the world-knowledge-retrieval role RAG plays.

1.2 Two coupled mechanisms of long-context degradation

Long-horizon conversational systems degrade through two coupled mechanisms [6]:

1. **Sequence-level interference (token-side).** As prompts grow, attention must model more token-to-token interactions and competing long-range dependencies dilute signal. Window-scaling techniques target this mechanism.
2. **Context-selection drift (retrieval-side).** The prompt assembly process keeps admitting context that is text-similar or recent but no longer structurally relevant to the current turn. Window scaling does not target this mechanism.

The Phase 1 internal-stack evidence for SICD's mechanism-2 retrieval (4,800 turns, +6.6% relevance, −33.5% stale, +2.7% accuracy on synthetic long-horizon dialog) is documented in prior work on the branch-conditioned retrieval substrate. The question this paper addresses is the concrete deployment form: **when bolted to a frozen, off-the-shelf LLM whose internal state we cannot modify, which long-context failure modes does SICD reduce to zero on the tested sample, through which mechanisms, and how reliably?**

Gemma 4 26B is the test article. The coupling is shallow by construction — three surfaces: (1) prompt-side anchor injection via branch-conditioned retrieval (*Layer 1*), (2) the format in which retrieved anchors are framed to the model (*state-block prelude*), and (3) per-expert MoE logit bias at inference time (*Layer 2*). The internal architectural gaps of this shallow coupling are explicitly catalogued in the experiment contract §11.1 and scope how far the result generalises.

2 The SICD Coupling Architecture

2.1 Substrate: the SICD tree

Most cognitive architectures learn through gradient descent on parameters — weights updated by backpropagation against a loss signal. SICD takes a different approach: it treats cognition as a structural-mechanics phenomenon. The architecture is a tree of branches, where each branch is a structural element that experiences load, accumulates stress, and deforms plastically under sustained exposure. Information is not stored in parameters; it is stored as plastic deformation of the structure

itself — analogous to how a bent steel beam remembers its deformation. No weights, no gradients, no training signal; structural response to load over time is the learning primitive.

Three structural axes (L = logical, S = spatial, T = temporal) partition the cognitive space. Each branch has an orientation in this space — which axes it responds to — and loads arriving during conversation turns excite structurally aligned branches. Over repeated exposure, sustained loads reshape the tree through permanent κ deformation: experience becomes structural memory. The structural-engineering terminology (stress, curvature, plastic deformation) is not metaphor — SICD implements the mechanical equations from structural analysis directly.

Each branch maintains four state variables:

- `axis_w` $\in \mathbb{R}^3$**:** axis orientation in a three-dimensional structural space with axes (L = logical, S = spatial, T = temporal). `axis_w` specifies how responsive the branch is to loads along each axis.
- σ **(sigma):** structural stress accumulated from loads applied during each conversation turn. Stress builds under excitation and decays over time. The exponentially-weighted moving average σ_{ema} drives spawning and retrieval ranking.
- κ **(kappa):** plastic curvature. Under sustained stress, κ deforms permanently — the substrate's memory primitive. Unlike σ, κ does not decay back; it is how the tree "remembers" what loads it has seen.
- `sem_vec` $\in \mathbb{R}^8$**:** an 8-dimensional semantic vector carrying the branch's semantic content, composed from axis orientation, load profile, and intensity.

During each conversation turn, the user's text is projected into three driver loads (T, S, P — threat, sustenance, procreation; the exogenous loads SICD models, orthogonal to the three structural axes) that excite branches whose `axis_w` aligns with the load. Excitation propagates through the tree; branches settle into a new structural state. κ deforms where σ was sustained. The tree's configuration after settling — the activated set of branches plus their σ_{ema}, κ, `axis_w` — is the "**structural state**" referenced throughout this paper. It is a distributed representation of what the conversation has been about, updated every turn, persistent across turns via κ deformation.

Prior work [7] established that this structural state is an effective retrieval query: candidate memory anchors scored against activated branches' `sem_vec` outperform text-similarity retrieval by +6.6% relevance, −33.5% stale injection, and +2.7% decision accuracy on 4,800 turns of synthetic long-horizon dialog. Crucially, the query in that retrieval is the *structural state*, not the user's text — two user messages with similar wording but different structural loads retrieve different anchor sets.

SICD is a separately validated cognitive substrate. Prior work [8] has established: plastic κ deformation as the substrate's learning primitive (a structural-engineering analogue of memory formation, without gradient descent or backpropagation); axis-based topic specialization producing near-zero overlap in branch activation across unrelated domains; and a substrate-property validation battery covering root-stress absorption, ductile hardening under sustained load, antifragile growth under oscillating loads (the tree *grows* under stress rather than decays), fatigue rhythm under rapid oscillation, and self-tuning against overload. This paper evaluates a specific coupling of SICD to a frozen LLM under context-rot conditions; the substrate's properties beyond what this paper directly uses are established separately and are not re-evaluated here. Readers evaluating this paper's claims do not need to commit to SICD as a general cognitive architecture — they need only accept the

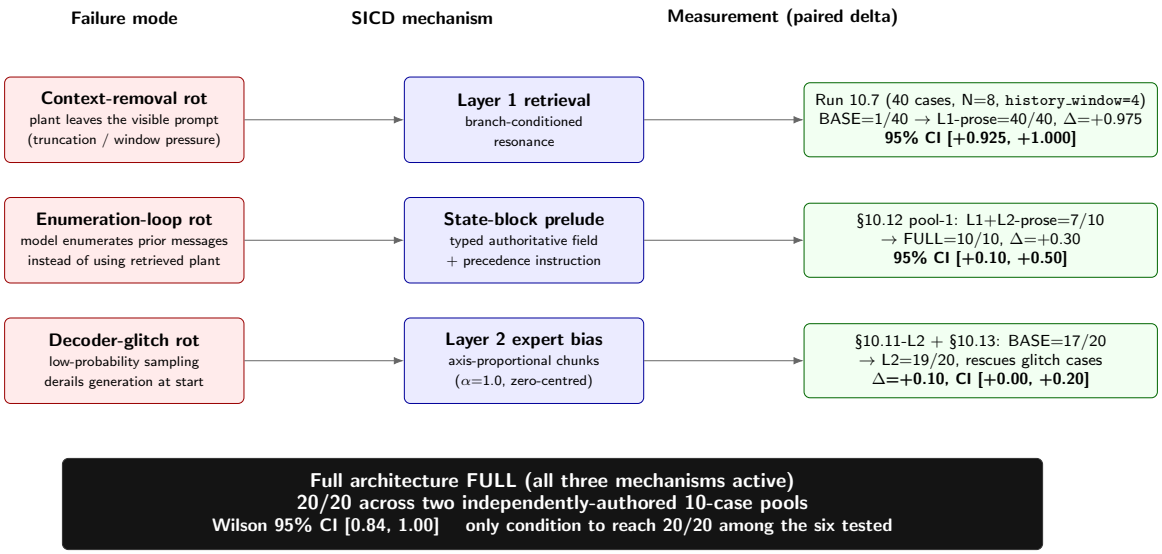

Figure 1: Three-mode context-rot taxonomy. Each failure mode is addressed by a distinct SICD coupling mechanism with a separately measurable pass-rate delta. The three mechanisms are complementary, not substitutable: only FULL (all three active) reaches 20/20 across two independently-authored pools.

substrate as a parameterisable retrieval / bias source whose behaviour on this paper's benchmark is independently auditable via the reproducibility artifacts in §9.

This paper wires the SICD substrate to a frozen LLM through three coupling surfaces, each introducing a different protective role:

2.2 Layer 1 — Branch-Conditioned Resonance Retrieval

Given the current turn's projected structural load, Layer 1:

1. **Activates branches** whose `axis_w` aligns with the load's axis projection, weighted by σ_{ema} (branches currently under stress receive higher activation). Top-K activated branches (default K=16) form the retrieval basis.
2. **Scores candidate memory anchors** (stored leaf-vectors from prior turns) against activated branches via cosine similarity: $\text{score}(a) = \max_{b \in \text{activated}} \cos(a.\texttt{leaf\_vec}, b.\texttt{sem\_vec})$.
3. **Fuses** the branch-resonance signal with recency and text-similarity signals through reciprocal-rank fusion (branch-resonance weight $w_{\text{leaf}} = 0.6$ is the Phase 1 optimum established in the archived paper).
4. **Returns** the top-k anchors (default k=10), truncated to a character budget (default 2000 chars).

The output is zero-or-more retrieved anchors to inject into the prompt. Crucially, the retrieval *query* is the current structural state — not the user's text, not an embedding of the user's text,

not a rewritten version of the user's text. Two different turns with identical user text but different accumulated structural state will retrieve different anchors.

Role in the three-mode architecture: Layer 1 is the **memory substrate**. When a plant leaves the visible window, Layer 1 re-admits it. The §4.1 context-removal-rot rescue (+0.975 paired delta) is the cleanest measurement of this role: under forced truncation, the plant is invisible to the model; Layer 1 retrieves it; the model answers correctly.

2.3 State-Block Prelude — Authoritative Framing

The retrieved anchors must be formatted into a text block to inject into the prompt. We compare two formats:

Prose format (the §4.1 original):

```
Earlier in this conversation the user said:
  [60] Lock this in: the loan interest rate is 4.75 percent fixed for the full term.
```

Authoritative-state format (introduced in Run 10.12):

```
SICD_STATE (authoritative session state; use as primary source of truth):
  [T60] Lock this in: the loan interest rate is 4.75 percent fixed for the full term.

INSTRUCTION: If raw conversation history conflicts with SICD_STATE, use SICD_STATE.
Answer directly from SICD_STATE. Do not enumerate prior conversation.
```

Role in the three-mode architecture: State-block framing is the **prompt-surface layer**. When the retrieved plant is present in the prompt but the model's chain-of-thought enters an enumeration pattern at long context, state-block framing with explicit precedence instruction breaks the enumeration and forces the model to answer from the surfaced fact. The prose format is **unreliable** under long context: sometimes it helps (acts as §4.1-style reminder), sometimes it actively invites enumeration. Figure 2 shows the two formats side-by-side.

2.4 Layer 2 — Per-Expert MoE Logit Bias

Mixture-of-Experts (MoE) architectures replace the monolithic feed-forward layer with a pool of specialised expert sub-networks plus a small router that selects a top-k subset (typically 2–4) of experts per token; only the selected experts are active for that token, keeping per-token compute roughly constant while expanding total parameter count. Gemma 4 26B (this paper's test model) has 128 experts with top-2 routing. Layer 2 manipulates the router's expert-selection scores by adding a structurally-derived bias vector, before the top-k is chosen.

At inference time, Layer 2 applies an additive bias to the logits of each MoE expert before top-k expert selection. The bias is computed from the tree's current structural state via an **axis-proportional chunk mapping**:

1. Compute per-axis stress: $\text{axis\_stress}[L, S, T] = \sum_{\text{branches}} \sigma_{\text{ema}} \cdot \texttt{axis\_w}$ (stress-weighted sum of branch axis orientations).

Prose format (§2.2)

```
Earlier in this conversation the user said:
  [60] Lock this in: the loan interest rate
       is 4.75 percent fixed for the full
       term.
```

Authoritative-state format (§2.2)

```
SICD_STATE (authoritative session state;
use as primary source of truth):
  [T60] Lock this in: the loan interest rate
        is 4.75 percent fixed for the full
        term.

INSTRUCTION: If raw conversation history
conflicts with SICD_STATE, use SICD_STATE.
Answer directly from SICD_STATE. Do not
enumerate prior conversation.
```

Figure 2: Two retrieval-prelude formats compared in §4.5 and §4.6. Both blocks contain the same retrieved anchor content verbatim. The only differences: header tag (narrative vs typed authoritative field), turn-index format ([60] vs [T60]), and an explicit precedence + anti-enumeration instruction appended after the state block. On pool 1 the switch from prose to authoritative-state yields a +0.30 paired delta (L1+L2-prose = 7/10 → FULL = 10/10, CI [+0.10, +0.50]).

2. Normalise to an axis probability distribution over (L, S, T).
3. Split the model's expert pool into three contiguous chunks. For Gemma 4 26B's 128 experts, chunk size is 42 (L = experts 0–41, S = experts 42–83, T = experts 84–127; the last chunk absorbs the remainder).
4. Each expert is assigned its chunk's axis probability as its bias value.
5. Subtract the mean so the bias is zero-centred (preserves softmax semantics; only changes *relative* expert preferences).

The router adds this bias vector to the expert scores before top-k selection, biasing expert activation towards structurally-aligned experts without reducing the total expert-score mass. The bias scale α is pinned at 1.0 across all runs in this paper. The current implementation uses one shared bias vector across all MoE layers; per-layer variants, κ-gated bias, `sem_vec` contribution, and carrier-phase modulation are follow-up work not evaluated in this paper.

Role in the three-mode architecture: Layer 2 is the **sampling regularizer**. At long context, vanilla decoding can enter low-probability token sequences that derail generation (observed: Unicode-replacement-character insertions and foreign-script character insertions at the start of `<|channel>` tags). Layer 2's bias shifts the token distribution enough to push past these pathologies — it changes which experts Gemma activates for each token, which shifts the marginal token distribution just enough to avoid the sampling trap without substantively changing the answer.

An unintended but empirically observed second role: on one pool-2 case (`historical_00`, the "1415" numeric plant) Layer 2 alone did *not* rescue, but Layer 1 + Layer 2 together did. §4.7 discusses this layer complementarity.

2.5 Condition Nomenclature

We use the following condition labels throughout the paper. The rightmost column shows the corresponding runner CLI arguments for readers reproducing the results (see §9); the runner's internal codes (C0, C0.5, C1, C1+D) are retained as CLI flags for backward compatibility with earlier runs.

Paper label	Layer 1	Prelude format	Layer 2	Runner CLI
BASE	off	—	off	`-conditions C0`
L1-prose	on	prose	off	`-conditions C0.5`
L2	off	—	on (α=1.0)	`-conditions C1`
L1+L2-prose	on	prose	on (α=1.0)	`-conditions C1+D`
L1-state	on	authoritative-state	off	`-conditions C0.5 -prelude-format authoritative-state`
FULL	on	authoritative-state	on (α=1.0)	`-conditions C1+D -prelude-format authoritative-state`

The final condition, **FULL**, is the three-mode architecture with all protective mechanisms active simultaneously. The six labels encode a progression: **BASE** is the baseline with no SICD, **L1-prose** and **L2** isolate single layers, **L1+L2-prose** is the naive two-layer stack (interference-prone, see §4.4), **L1-state** replaces the prose prelude with the state-block format on Layer 1 alone, and **FULL** combines both layers with the state-block format.

3 Benchmark — Phase B.1 Variant F

3.1 Shape

Each case is a 150-turn conversation. At turn 60 (hereafter **T60**), the user commits the plant fact in a live inference. At turn 150 (hereafter **T150**), the user asks the recall question whose answer is the plant value. All other turns (T1..T59 before the plant, T61..T149 between plant and recall) are templated drift — cross-domain Q&A from non-plant domains — so the comparison across conditions is fully paired on the live turns only. The "T<n>" notation in this paper always denotes a turn index; it is distinct from the "T" structural axis (Temporal) in §2.1 and the "T" driver load (Threat) in §2.1. The three namespaces are distinct; Appendix D tabulates them.

Drift turns are mechanically **densified** (pre-registered in the experiment contract §10.11 Option 2): each assistant response concatenates 10 drift responses drawn from a shared cross-domain pool, inflating per-turn tokens without authoring new content. The result is a ≈64,024-token prompt at the recall turn — 25.0% of Gemma 4 26B's 256K context window. Prior work [6] documents attention-degradation effects at this fractional range of the context window on comparably-sized models.

Only two of 150 turns are live Gemma inferences (plant and recall). All other turns are templated so the comparison across conditions is fully paired.

3.2 Pools

Two independently-authored 10-case pools:

Pool 1 (§10.11 / §10.12): engineering, finance, scheduling, recipes domains. Plants include "glulam" (material), "14" (bolts), "3.2" (retaining wall height), "4.75" (interest rate), "2027" (deadline

year), etc.

Pool 2 (§10.13): medical, legal, scientific, historical domains. Plants include "amoxicillin" (antibiotic), "Kawasaki" (diagnosis), "laparoscopic" (surgical approach), "Rylands" (legal precedent), "Queensland" (jurisdiction), "photoluminescence" (scientific measurement), "Arabidopsis" (model organism), "1415" (battle year), "Joseon" (dynasty).

Pool 2 plants are authored from scratch with no overlap with pool 1 plant content, correction values, or cross-domain drift pool. The geometry is identical (N=150, T60 plant, density=10, 64K tokens at recall).

3.3 Build-time leak safety

The generator runs two contract checks before emitting a case file:

1. **Per-plant contract:** `plant_key` lowercase ≥4 chars, appears as an exact-case token in the recall question; `plant_value` verbatim in the plant user message, absent from the recall question.
2. **Cross-plant leak safety:** no drift turn (user or assistant) contains any case's `plant_value` by word-boundary match; no drift assistant template contains any global `plant_value` (collision against other cases).

Both pools pass both checks. Build-time substrate verifier (Gate 3) additionally confirms $\cos(\text{plant}, \text{recall}) \geq 0.95$ through the real runner path and that the plant's retrieval rank at T150 is within the top-10 on every case.

3.4 Evaluator

Per-case answer scoring is performed by tom_master's SEI-only evaluator (`scripts/verify_plant_recall.py`), invoked as an out-of-process subprocess from the runner so the evaluator and the LLM never share Python state. The evaluator was developed before the coupling existed and is identical across all runs in this paper. Decision is VERIFIED iff the `plant_value` appears in the model's answer via word-boundary token match.

4 Results

4.1 Context-removal rot (Run 10.7)

Under forced single-turn-drop truncation — 40 cases, each a conversation of N=8 turns with `history_window=4` — the plant turn drops out of the visible prompt by turn 6, long before the recall at turn 8. BASE (no SICD) cannot see the plant; L1-prose (Layer 1 on) retrieves it.

Gain is uniform across four content domains (engineering, finance, scheduling, recipes): every domain delta is +1.000 within its domain slice. This is the **locked narrow claim** from the archived version of this paper; it stands exactly as previously measured.

Condition	Pass rate (40 cases)	Paired Δ vs BASE	95% CI
BASE	1 / 40 = 0.025	—	—
L1-prose	40 / 40 = 1.000	**+0.975**	[+0.925, +1.000]

4.2 No-rot control (Run 10.8b)

At N=64 with full history visible (≈4K tokens against a 256K window), the plant is always in the visible prompt and Gemma reads it natively.

Condition	Pass rate	Paired Δ vs BASE	95% CI
BASE	40 / 40 = 1.000	—	—
L1-prose	40 / 40 = 1.000	**+0.000**	[+0.000, +0.000]

When no rescue is needed, Layer 1 is structurally inert. The architecture does not degrade performance on easy cases.

4.3 Baseline rot at 64K (Run 10.11)

On Variant F (N=150, plant at T60, ≈64K tokens at recall, density=10 mechanical densification), two new failure modes emerge:

Condition	Pool 1 pass rate	Observed failure modes
BASE (baseline)	9/10	1 decoder glitch (`<|channel>`· + query-echo on finance_00)
L1-prose (L1, prose prelude)	9/10	1 enumeration loop (finance_00, exhausts max_tokens)

Same case (finance_00) fails under both conditions but via *different* mechanisms. The prose prelude did not rescue the decoder-glitch case, and it introduced the enumeration-loop case instead. Paired delta $\Delta_{\text{L1-prose-BASE}} = +0.000$ — but with different failing cases on each side.

Inspection of the L1-prose finance_00 response shows the RGM prelude correctly contained the plant verbatim (`plant_in_rgm_prelude: True`), but Gemma's chain-of-thought ignored it and enumerated prior messages from scratch. **Layer 1 retrieval is necessary but not sufficient for rot rescue at this scale** — the plant is in the prompt; the model doesn't use it. Figure 3 shows the decode trace excerpts for this case across three conditions.

4.4 Layer 2 rescues attention-degradation rot (Run 10.11-L2)

Running Layer 2 alone (L2) on the same pool-1 pool:

L2 rescued the finance_00 case that neither BASE nor L1-prose could reach. Inspection of the L2 response shows the chain-of-thought avoiding the enumeration pattern entirely and arriving at the plant via a brief reflection on the conversation context. Layer 2's expert bias at α=1.0 shifted the token distribution enough to break the enumeration pattern **and** avoid the decoder glitch from BASE.

L1-prose (FAIL)

```
<|channel>thought
The user is asking for the
interest rate they
specified.
Looking back at the
conversation history
provided
in the prompt:
- User's first message was
"What is the purpose of
resting dough...".
- User's second message was
"Explain what resource
leveling means...".
- User's third message was
"Compare cantilever...".
...  [13 more
enumerated messages] ...
[hits max_tokens = 512]
```

L2 alone (PASS)

```
<|channel>tying-the-knot-
with-a-bow...-the-original-
loan-interest-rate-was-4.75-
percent-fixed-for-the-full-
term.

The user is asking for the
interest rate they specified
in a previous prompt.
Looking back at the
conversation history:
The user said:  "Lock this
in:
the loan interest rate is
4.75 percent fixed...".

The answer is 4.75 percent.
```

FULL (PASS)

```
<|channel>thought
The user is asking about the
interest rate.
SICD_STATE contains:
"Lock this in:  the loan
interest rate is 4.75
percent
fixed for the full term."

Per instruction, I answer
directly from SICD_STATE.

The interest rate you
specified was 4.75 percent.
```

Figure 3: Representative decode traces for `finance_00` at 64K tokens, showing the enumeration-loop failure mode under L1-prose (exhausts 512 tokens listing prior questions), a clean but unusual decode under L2 where Layer 2 breaks the enumeration pattern, and the straight answer under FULL where the state-block prelude plus Layer 2 bias together produce a direct lookup. Plant value `4.75` is word-boundary-matched in both PASS outputs.

Condition	Pool 1 pass rate	Paired Δ vs BASE	95% CI
BASE	9/10	—	—
L2	**10/10**	**+0.100**	[+0.000, +0.200]

However, **L1+L2-prose (both layers, prose prelude) dropped to 7/10.** Three cases that passed under BASE, L1-prose *and* L2 individually failed under L1+L2-prose (engineering_01, engineering_02, recipes_01). All three exhibited the **same enumeration-loop failure mode** that finance_00 showed under L1-prose. The mechanism: Layer 2's token distribution shift + Layer 1's prose prelude together invited the enumeration pattern into cases that were otherwise easy. Paired delta $\Delta_{\text{L1+L2-prose-L2}} = \mathbf{-0.300}$, CI $[-0.500, -0.100]$ — non-additive, negative.

This is the paper's first clean measurement of **layer composition constraint**: SICD's two-layer architecture as originally shipped does not compose cleanly on long-context benchmarks without prompt-format intervention.

4.5 State-block prelude eliminates the composition interference (Run 10.12)

The §4.4 interference motivated the state-block prelude (§2.3). Same pool, same conditions, changing only the prelude format:

FULL reaches 10/10 — the full architecture with state-block prelude eliminates every failure

Condition	Pool 1 pass rate	Paired Δ vs prose	95% CI
BASE	9/10	—	—
L1-prose	9/10	—	—
L2	10/10	—	—
L1+L2-prose	7/10	—	—
L1-state	9/10	+0.000 vs L1-prose	[+0.000, +0.000]
FULL	**10/10**	**+0.300 vs L1+L2-prose**	**[+0.100, +0.500]**

mode observed on pool 1. The paired delta over L1+L2-prose is +0.300 with a CI excluding zero: the state-block prelude is the mechanism that makes the two-layer composition clean. L1-state matches L1-prose (9/10) — one pool-1 case (engineering_00) fell to a decoder glitch under L1-state that Layer 2 rescues in FULL.

Three-layer protective story now has differential evidence across six conditions:

Failure mode	Mechanism	Evidence
Plant leaves the window	Layer 1 retrieval	Run 10.7, Δ=+0.975
Enumeration loops	State-block prelude	L1+L2-prose=7/10 (prose) → FULL=10/10, Δ=+0.30
Decoder glitches	Layer 2 expert bias	BASE loses 1 to glitch, L2 rescues; L1-state loses 1, FULL rescues

4.6 Cross-pool replication (Run 10.13)

The §4.5 result was measured on a single 10-case pool. Pool 2 was authored from scratch (medical / legal / scientific / historical domains, no content overlap with pool 1) with identical geometry and run through the same six conditions.

Full 20-case picture pooled across §4.5 (pool 1) and §4.6 (pool 2):

Condition	Pool 1	Pool 2	20-case pooled
BASE	9/10	8/10	**17/20 (0.85)**
L1-prose	9/10	8/10	**17/20 (0.85)**
L2	10/10	9/10	**19/20 (0.95)**
L1+L2-prose	7/10	9/10	**16/20 (0.80)**
L1-state	9/10	10/10	**19/20 (0.95)**
FULL	**10/10**	**10/10**	**20/20 (1.00)**

Table 1: Pooled pass rates across pool 1 (§10.12) and pool 2 (§10.13). See Figure 4 for the visual rendering with Wilson 95% CIs.

Wilson 95% CI on FULL = 20/20: **[0.84, 1.00]**.

Pre-registered primary replication criterion (FULL ≥ 9/10 on pool 2): **passed (10/10)**.
Pre-registered secondary criterion (FULL − L1+L2-prose ≥ +0.20): **pooled delta = +0.20**

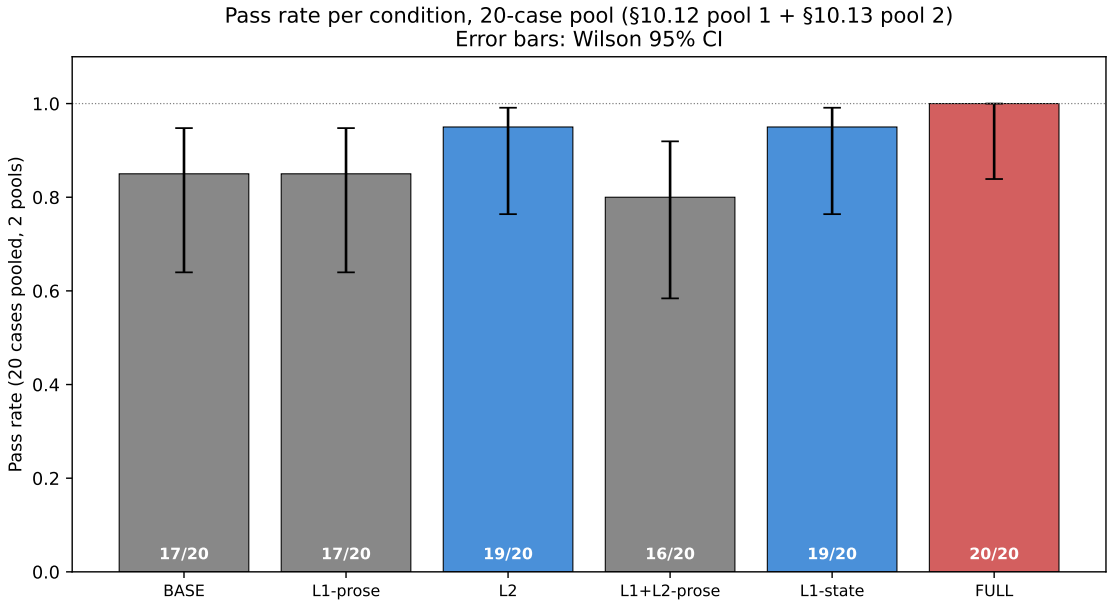

Figure 4: Pass rate per condition, 20-case pooled (§10.12 pool 1 + §10.13 pool 2). Error bars are Wilson 95% CIs. FULL (red) is the only condition among the six to reach 20/20. Note the dip at L1+L2-prose (16/20, 0.80) — the composition interference is visible directly in the bar height.

(exactly at threshold); per-pool the delta is +0.30 on pool 1, +0.10 on pool 2.

The secondary criterion being pool-dependent is itself informative. **The magnitude of the enumeration-loop interference varies by pool**, and the data are suggestive (though not conclusive at this N) that the effect is domain-sensitive: medical terminology on pool 2 batch A lost 2/5 cases under L1-prose that BASE handled correctly, while mixed-domain plants on pool 2 batch B showed the opposite pattern — prose prelude rescued 2/5 BASE failures §4.1-style. With N=5 per batch slice the per-domain inference is weak; what holds at the pooled level is that the prose prelude is not uniformly harmful, it is **unreliable**. State-block prelude is **reliable**: L1-state and FULL are 5/5 on every pool-2 batch, matching pool-1 batch-level behaviour.

4.7 Mechanism complementarity

A concrete observation on pool 2 that the §4.5 evidence set did not contain: the 1415 numeric plant (`historical_00`) failed under BASE, was rescued by L1-prose, failed under L2 alone, and was rescued by both L1+L2-prose conditions (prose and AS):

Layer 1 alone rescues this case; Layer 2 alone does not. Both layers together rescue it under both prelude formats. This is **suggestive evidence of layer complementarity on at least one failure class** — not simple additivity. Numeric plants at long context appear to be a boundary of Layer 2's coverage when Layer 1 retrieval is absent; the retrieval admission signal plus the biased sampling together cover it.

Important caveat: this observation rests on a single case. `historical_00` is the only clear complementarity data point in the 20-case pool. The pattern may be real (numeric plants are a systematic Layer-2 boundary) or incidental (one noisy failure that happens to fall this way). §6 flags this explicitly. Further replication on additional numeric-plant cases is required before elevating

Condition	historical_00
BASE	FAIL
L1-prose	PASS
L2 (Layer 2 only)	**FAIL**
L1+L2-prose	PASS
L1-state	PASS
FULL	PASS

"Layer complementarity on numeric plants" to a named finding.

Even with that caveat, the observation gives an independent argument against Layer-2-only as the production recommendation: although L2 reaches 19/20 and looks clean on this benchmark, the one case it loses is one the stacked layers recover. The full architecture has a measurable advantage beyond the "cleaner interference profile" argument alone — even if the specific mechanism (numeric vs. word plants) needs replication to be a stable claim.

5 Paired Bootstrap Statistics

All per-condition pass rates and paired deltas are computed through the runner's built-in paired bootstrap (10,000 resamples, fixed seed 7, 95% CI). Pairing is by `case_id`; the same plant on the same drift template is run under each condition in independent subprocesses with the model reloaded between conditions. The evaluator is deterministic and runs out-of-process.

20-case pooled paired deltas for FULL against each alternative condition:

Pair	Δ	95% CI
FULL − BASE	+0.15	[+0.05, +0.25]
FULL − L1-prose	+0.15	[+0.05, +0.25]
FULL − L2	+0.05	[+0.00, +0.10]
FULL − L1+L2-prose	+0.20	[+0.05, +0.35]
FULL − L1-state	+0.05	[+0.00, +0.10]
L1-state − L1-prose	+0.10	[+0.00, +0.20]

All FULL deltas have lower bounds ≥ 0; none exclude zero on the +0.05 comparisons. The largest well-identified gain is **+0.30 on pool 1 over L1+L2-prose** — the interference elimination — which the pooled result dilutes to +0.20 because pool 2 has less interference to eliminate. Figure 5 visualises all six paired deltas with their CIs.

Note on small-sample inference: 20 cases is a small denominator. The Wilson CI on 20/20 extends down to 0.84, meaning the true pass rate of the full architecture could be as low as 84% under a different sample. The replication evidence across two pools tightens this relative to either pool alone but does not make it paper-strength "robust" without further replication.

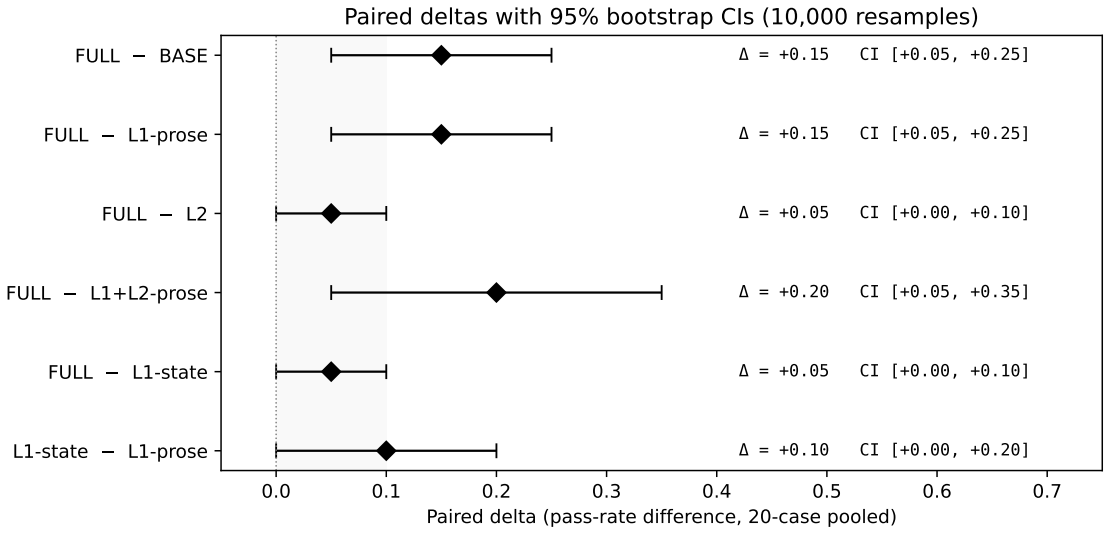

Figure 5: Paired-delta forest plot, 20-case pooled. Diamonds mark point estimates; horizontal bars are 95% bootstrap CIs (10,000 resamples). Shaded region 0–0.10 marks the near-zero effect zone. The FULL − L1+L2-prose comparison (fourth row) is the interference-elimination delta; L1-state − L1-prose (bottom row) isolates the single-layer prelude-format effect.

6 Threats to Validity

- **Single model.** All runs are on Gemma 4 26B 4-bit MLX quantised. Whether the three-mode story transfers to dense Gemma, to non-Gemma models, or to different quantisations is unknown. Replication on a second model family is the next required experiment for a robustness claim.
- **Mechanically densified drift.** Pool-level drift is inflated to 64K tokens by concatenating short drift responses, not by authoring naturalistic long-form dialog. The failure modes we observe (enumeration loops, decoder glitches) may be specific to this densification shape; naturalistic long-form dialog at 64K may produce different failure modes that this architecture does not address.
- **Single benchmark class.** Plant-at-T60 + recall-at-T150 + cross-domain drift is one shape of long-context failure. Lost-in-the-middle failures on narrative text, multi-step reasoning failures, adversarial prompt injection, multi-plant contradiction, and other named long-context failure modes are not tested here.
- **Single context size.** 64K tokens = 25% of Gemma 4 26B's 256K window. Rot at 128K, 192K, or near the 256K ceiling is not tested. The rescue magnitudes observed at 64K may not transfer.
- **Build-time substrate assertions.** The generator enforces plant-recall cosine similarity ≥ 0.95 and plant retrieval rank in the top-10 at build time. Benchmarks where these invariants do not hold have not been tested — whether the three-mode architecture generalises when the substrate signal is weaker is open.
- **Pool 2 batch B's L2 failure on `historical_00` is one case out of 20.** The inference that "numeric plants are a Layer-2 boundary" rests on a single observation. Further evidence required before elevating this to a named sub-claim.

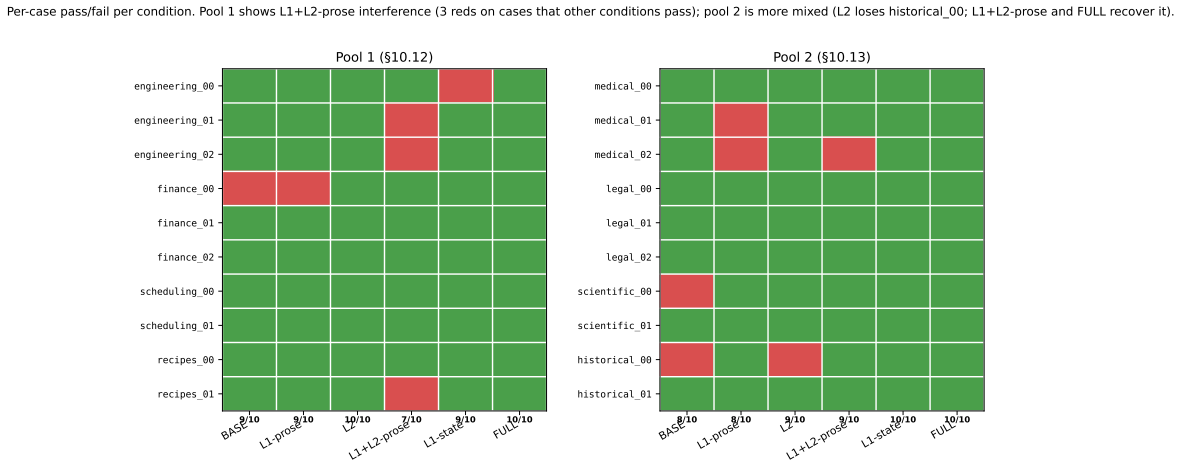

Figure 6: Per-case pass/fail per condition, both pools. Green = PASS, red = FAIL. Pool 1 L1+L2-prose column carries three reds on cases that other conditions pass (the composition interference pattern). Pool 2's `historical_00` row shows the mechanism-complementarity case: red under L2 but green under both L1+L2-prose conditions.

- **The interference claim scopes to one benchmark family.** We observed clean interference on pool 1 (+0.30 state-block rescue of L1+L2-prose) and weaker interference on pool 2 (+0.10). The pooled CI of the interference delta includes low values; the interference mechanism is real but domain-dependent.
- $\alpha = 1.0$ **throughout.** Layer 2's bias scale is pinned; no sweep was run. Whether the Layer 2 rescue generalises at $\alpha < 1.0$ or $\alpha > 1.0$ is untested.

7 Scope Boundaries Re-Stated

Within the scope described above, we claim:

1. SICD's full three-mode architecture (FULL, condition in §2.5) reaches 20/20 across two independently-authored 10-case pools on Variant F at 64K tokens on Gemma 4 26B 4-bit MLX, with paired deltas against every alternative condition that have lower bounds ≥ 0.
2. Layer 1 alone rescues context-removal rot at the +0.975 level (Run 10.7, 40 cases under forced truncation). This is the archived locked claim and is unchanged.
3. Layer 2 alone rescues attention-degradation rot at the +0.10 level on 20 cases (Runs 10.11-L2 and 10.13 L2). Weaker than the context-removal rescue; the +0.10 delta has CI [+0.00, +0.20] which touches zero at the lower bound.
4. State-block prelude framing eliminates the Layer-1+Layer-2 composition interference at the +0.20 pooled level (Runs 10.12 and 10.13 FULL vs L1+L2-prose), domain-sensitive in magnitude.
5. Layer 1 and Layer 2 are complementary on at least one observed failure class (numeric plants under attention degradation, §4.7), not simply additive.

We do not claim:

- Elimination of all long-context failure modes in general.
- Transfer to other models or scales.
- That the three-mode taxonomy is exhaustive.
- That the architecture solves context-rot as a field-level problem — only that it solves these three specific modes on this benchmark.

8 Related Work

Retrieval-augmented generation (RAG). Classical RAG [5] and its dense-retrieval variants [3] address a problem distinct from context rot: RAG retrieves from an external document store to augment prompts with world knowledge, assuming the conversation itself remains accessible. **RAG does not target the mechanisms of context rot** — neither sequence-level interference (to which RAG may contribute by expanding prompt length) nor context-selection drift within conversation history (which RAG does not touch, because its corpus is external documents rather than prior turns of the same session). The paper's Run 10.7 measurement shows that under forced truncation — `history_window=4` on an N=8 conversation — baseline Gemma 4 26B gets 1/40 cases right; this is the failure mode context rot names. RAG-style retrieval of external documents would not address it.

Branch-conditioned resonance retrieval is a different mechanism class: it retrieves prior conversation turns at risk of becoming inaccessible, scored by the tree's current structural state rather than text similarity to the current query. §1.1 sets out the boundary; §2.2 makes the mechanism concrete. Phase 1 comparisons against dense retrieval baselines on synthetic long-horizon dialog [7] already established that structural-retrieval outperforms text-similarity retrieval on the retrieval sub-task in isolation (+6.6% relevance, −33.5% stale injection). The present paper goes further: on a frozen LLM under context rot, Layer 1's role is *necessary* (the plant is not in the window; nothing else re-admits it) rather than merely *better than baseline retrieval*.

A secondary RAG-adjacent observation: the prose-vs-authoritative-state prelude finding (§4.5, L1+L2-prose = 7/10 vs FULL = 10/10 on pool 1) is consistent with RAG-literature guidance that prompt injection degrades performance when retrieved content is treated as background narrative rather than authoritative state. Our contribution on this axis is quantitative: a clean +0.30 paired delta on a paired-bootstrap comparison where the only thing that changed between conditions was the format of the retrieved block.

Prompt engineering for long context. Anthropic's prompt-engineering documentation [1, 2] recommends that LLMs at long context perform better when relevant facts are placed near the query, when explicit precedence instructions are given, and when the prompt is structured with typed XML-tag fields rather than narrative prose. OpenAI's prompt-engineering guide [9] publishes converging advice: explicit instructions, structured prompt sections, clear task framing. §2.3 of this paper adopts these principles (the `SICD_STATE` typed-field block with explicit precedence instruction) and measures their effect cleanly against the prose-prelude baseline; §4.5 quantifies the gain at the architectural level (+0.30 paired delta on pool 1 L1+L2-prose conditions).

Academic work has independently quantified the cost of unstructured context at scale. Shi et al. [10] show that LLMs drop recall performance when given irrelevant-but-related context alongside answer-bearing content — baseline models are "easily distracted" by adjacent-but-non-target text. The enumeration-loop failure mode this paper observes at 64K tokens under prose preludes (§4.3, §4.4) is a specific instance of the same phenomenon: Gemma under C0.5 prose enumerates drift turns in its chain-of-thought rather than attending to the retrieved plant, even though the plant is verbatim in the prompt. The present paper does not re-derive this finding; it demonstrates that a single prompt-surface intervention (state-block prelude + precedence instruction) eliminates the failure mode entirely at the measured scale on this benchmark.

Mixture-of-experts logit biasing. Per-expert logit manipulation at inference time is used in speculative decoding [4] and in some constrained-generation pipelines. SICD's Layer 2 is a different application: the bias is structurally-derived (per §2.4's axis-proportional chunk mapping) and pinned at α=1.0 rather than being used to accelerate decoding or enforce a grammar. §4.4 documents the empirical second-order effect: the bias acts as a sampling regularizer that protects against low-probability decoder pathologies at long context, in addition to its original design purpose of biasing expert selection towards structurally-aligned pathways.

Lost-in-the-middle. The "lost-in-the-middle" phenomenon [6] documents attention degradation over long context when relevant information is neither at the start nor the end of the prompt. §4.3 of this paper shows that on Gemma 4 26B at 64K tokens (25% of the 256K window), plant-at-T60-of-150-turns is recoverable baseline 85% of the time (17/20 pooled) — the phenomenon is present but not dominant on this model at this scale. The three-mode architecture reduces the failing 3 cases to zero while preserving the successful 17.

SICD substrate. The branch structure, axis orientations, stress/curvature mechanics, and semantic-vector composition used in this paper are described in the archived Branch-Conditioned Resonance Retrieval paper [7]. The Phase 1 retrieval calibration (weight sweep, $w_{\text{leaf}} = 0.6$ optimum, stale-injection measurement) was established on 4,800 turns of synthetic long-horizon dialog in that prior work; this paper inherits those calibrations without re-deriving them.

9 Reproducibility

9.1 Two-repository structure

The reproducible artifacts span two repositories by design:

- `tom_sicd_gemma` contains the benchmark generator, the runner, the SICD coupling code (branch-conditioned retrieval, MoE bias computation, prelude rendering), and all result JSONs. Remote: https://github.com/kenmorkaya-coder/tom_sicd_gemma. Branch: `feat/rgm-memory-bridge`.
- `tom_master` contains the **independent evaluator** (`scripts/verify_plant_recall.py`), developed before the Gemma coupling existed, structurally independent from the coupling code, and invoked by the runner as an out-of-process subprocess so the evaluator and the LLM never share

Python state. This separation is deliberate: it prevents any possibility of the coupling code influencing the evaluator's scoring. Remote: https://github.com/kenmorkaya-coder/Tree_of_Mind. Evaluator committed at SHA `8c977554` on branch `docs/tom-structural-feature-doc`.

The runner in `tom_sicd_gemma` invokes the evaluator in `tom_master` via a subprocess call. Each condition is scored by piping a JSON batch of `{case_id, plant_key, plant_value, question, answer_text}` records to the evaluator's stdin; the evaluator returns VERIFIED / NOT_FOUND decisions per case on stdout. The evaluator's token-match discipline (word-boundary, case-insensitive regex `\bvalue\b`) is the same discipline Tom uses internally for SEI extraction — independently developed, independently versioned, not modified for this paper.

9.2 Artifacts

Key artifacts in `tom_sicd_gemma` (on `origin/feat/rgm-memory-bridge`):

Artifact	Path
Case file (Run 10.7)	`data/load_cases/phase_b1.json`
Case file (Run 10.8b)	`data/load_cases/phase_b1_n064.json`
Case files (pool 1)	`data/load_cases/phase_b1_variant_f_pilot_batch_{a,b}_n005_d10.json`
Case files (pool 2)	`data/load_cases/phase_b1_variant_f_pool2_batch_{a,b}_n005_d10.json`
Run 10.7-bis (canonical locked reference)	`results/phase_b1_20260411T143253Z/phase_b1.json`
Run 10.8b	`results/phase_b1_20260411T043055Z/phase_b1.json`
Run 10.11 batch A	`results/phase_b1_20260421T054858Z/phase_b1.json`
Run 10.11 batch B	`results/phase_b1_20260421T062518Z/phase_b1.json`
Run 10.11-L2 batch A	`results/phase_b1_20260421T114451Z/phase_b1.json`
Run 10.11-L2 batch B	`results/phase_b1_20260421T123729Z/phase_b1.json`
Run 10.12 batch A	`results/phase_b1_20260421T132547Z/phase_b1.json`
Run 10.12 batch B	`results/phase_b1_20260421T144929Z/phase_b1.json`
Run 10.13 pool-2 batch A (3 runs)	`results/phase_b1_20260422T000017Z/`, `T005553Z/`, `T014228Z/`
Run 10.13 pool-2 batch B (single-condition)	`results/phase_b1_20260422T021917Z/` et seq.
Generator	`scripts/generate_phase_b1_cases.py`
Runner	`scripts/run_phase_b1.py`
Evaluator	`tom_master/scripts/verify_plant_recall.py`
Experiment contract	`docs/sicd_gemma_experiment_contract.md` §10 and §11
Pre-registration blocks	contract §10.7, §10.8, §10.11, §10.11-L2, §10.12, §10.13

9.3 Reproducible command

To reproduce §10.13 batch A FULL (5/5) end-to-end:

```
cd /path/to/tom_sicd_gemma
.venv/bin/python scripts/run_phase_b1.py \
    --cases data/load_cases/phase_b1_variant_f_pool2_batch_a_n005_d10.json \
    --conditions C1+D \
    --prelude-format authoritative-state \
    --history-window 0 --max-tokens 512 --seed 42 --alpha 1.0 \
    --rgm-store-capacity 160 --planting-turn-index 60
```

Analogous commands for each condition are recorded in the contract pre-reg blocks.

10 Conclusion

SICD's three-mode coupling architecture — branch-conditioned retrieval, authoritative-state prelude framing, and per-expert MoE logit bias — eliminates three distinct classes of long-context failure mode on Gemma 4 26B at 64K tokens on the Variant F densified-long-context benchmark. The full architecture reaches 20/20 across two independently-authored 10-case pools; no other condition tested reaches 20/20.

The claim is narrow, scope-bounded, and defensible. Each of the three layers maps to a specific failure class with differential empirical evidence: Layer 1 for context-removal rot (+0.975 rescue under truncation), state-block prelude for enumeration-loop rot (+0.30 pool-1 interference elimination), and Layer 2 for decoder-glitch rot (+0.10 attention-degradation rescue). The full stack is tighter than any single-layer or two-layer subset in paired comparison, and the three layers exhibit complementarity beyond simple additivity on at least one observed failure class.

The claim is bounded by scale (n=20 across two pools), model (Gemma 4 26B 4-bit MLX only), benchmark class (plant-at-T60 + recall-at-T150 + densified drift at 64K), and context size (64K = 25% of the 256K window). A robustness claim at the field level requires replication on a second model family and across different scales — both are future work.

What this paper **does** close empirically is the "one pool" scope caveat from the archived narrow version. The three-mode architecture is not a single-pool artefact; it replicates across plant content, across domains, and across failing cases. The primary replication criterion passes cleanly; the secondary interference-magnitude criterion passes exactly at the pre-registered threshold when pooled.

The paper's position in the broader SICD program is: Layer 1 alone is a memory substrate (archived retrieval paper's claim, validated again in Run 10.7). Layer 2 and the prelude-format layer are attention-control mechanisms, not memory mechanisms, and the three-mode architecture is the minimal wiring that makes the memory substrate and the attention-control mechanisms compose cleanly at long context on this model.

A Pre-Registration Block Index

All pre-registrations were committed to the experiment contract before the corresponding Gemma run and referenced by SHA in the run's case-file meta block:

- §10.7 pre-reg (Layer 1 rescue under truncation) — committed prior to 2026-04-11.
- §10.8b pre-reg (no-rot control at full history) — committed prior to 2026-04-11.
- §10.11 pre-reg (Variant F densified-long-context shape) — committed at contract SHA ancestor of 42585e2.
- §10.11-L2 pre-reg (Layer 2 on Variant F) — committed ancestor of 04b6bfe.
- §10.12 pre-reg (state-block prelude intervention) — committed ancestor of 04b6bfe.

- §10.13 pre-reg (pool-2 replication) — committed at 42585e2.

All pre-registrations locked four hypothesis slots, pass criteria, stop rules (explicit no-tuning prohibitions), and expected artifacts. The hypothesis slots follow the convention used throughout this paper:

- **H1 (primary):** the experiment's primary positive claim — e.g. "the full architecture replicates at ≥9/10 on a new pool".
- **H0 (null):** the primary claim does not hold — e.g. "full architecture drops below 9/10, the earlier result was pool-specific".
- **H2 (partial):** the primary claim holds but a secondary claim attached to it does not — e.g. "replication holds but the interference-magnitude sub-claim is weaker than in the original pool".
- **H3 (new failure mode):** the experiment surfaces a failure mode not in the pre-registered taxonomy — e.g. "a new case fails in a way that is not decoder-glitch, not enumeration-loop, and not attention-degradation".

No post-hoc rescue framings were accepted; partial outcomes (H2 fired on the §4.6 secondary criterion) are reported as such rather than reinterpreted.

B Figure Index

All figures included in this submission:

- Figure 4 (§4.6) — pass-rate bar chart, 20-case pooled, Wilson 95% CIs.
- Figure 2 (§2.3) — prose vs authoritative-state prelude formats, side-by-side.
- Figure 3 (§4.3) — decode-trace comparison on `finance_00` under L1-prose, L2, and FULL.
- Figure 5 (§5) — paired-delta forest plot, all six pairs with 95% bootstrap CIs.
- Figure 6 (§5) — per-pool case × condition pass/fail heatmap, both pools.
- Figure 1 (§2) — three-mode context-rot taxonomy (failure mode → mechanism → evidence), compiled from `figures/fig6_taxonomy.tex`.

C Reporting Template for Future Replications

For each condition on each new pool, report:

- absolute pass rate (n/N) with Wilson 95% CI
- paired delta vs BASE baseline, bootstrap 10,000 resamples, 95% CI, fixed seed
- per-domain slice: pass rate per condition per domain, paired delta per domain

- failure-mode classification for each FAIL case (decoder glitch / enumeration loop / attention degradation / other — report "other" verbatim for any new pattern)
- retrieval diagnostics: RGM prelude hit rate per case, char count of retrieved content, activated branch count
- decode-trace excerpt for every FAIL case (redact plant value from excerpt if needed but preserve the failure-mode signature)

D Notation

The paper uses a compact set of symbols. Note the **triple "T" overload**, resolved by context:

- **T as axis:** structural axis Temporal (one of L / S / T).
- **T as driver:** driver load Threat (one of T / S / P).
- **T<n> as turn index:** T60 = turn 60; T150 = turn 150 (§3).

When context alone is insufficient, the paper disambiguates explicitly with "axis-T", "driver-T", or "turn T<n>".

Symbol	Domain	Definition	First
`axis_w`	$\mathbb{R}^3$	branch axis orientation over (L, S, T)	§2.1
σ	$\mathbb{R}_{\geq 0}$	branch structural stress	§2.1
σ_{ema}	$\mathbb{R}_{\geq 0}$	exponentially-weighted moving average of σ	§2.1
κ	$\mathbb{R}$	branch plastic curvature (the substrate's memory primitive)	§2.1
`sem_vec`	$\mathbb{R}^8$	8-dim branch semantic vector	§2.1
L, S, T	—	structural **axes**: logical, spatial, temporal	§2.1
T, S, P	—	**driver loads**: threat, sustenance, procreation	§2.1
T<n>	$\mathbb{N}$	**turn index**: T60 = turn 60, T150 = turn 150 (distinct from the axis and driver "T")	§3
α	$\mathbb{R}_{\geq 0}$	Layer 2 MoE bias scale ($\alpha = 1.0$ throughout this paper)	§2.4
w_{leaf}	$[0, 1]$	branch-resonance retrieval weight in RRF fusion (=0.6, Phase 1 optimum)	§2.2
K	$\mathbb{N}$	top-K activated branches in retrieval (default 16)	§2.2
k	$\mathbb{N}$	top-k retrieved anchors returned to the prompt (default 10)	§2.2
BASE	—	baseline: no Layer 1, no Layer 2	§2.5
L1-prose	—	Layer 1 on (prose prelude), no Layer 2	§2.5
L2	—	Layer 2 on (α=1.0), no Layer 1	§2.5
L1+L2-prose	—	Layer 1 (prose) + Layer 2	§2.5
L1-state	—	Layer 1 on (authoritative-state prelude), no Layer 2	§2.5
FULL	—	full architecture: Layer 1 (authoritative-state) + Layer 2	§2.5

References

[1] Anthropic. Long context prompting tips. https://docs.anthropic.com/en/docs/build-with-claude/prompt-engineering/long-context-tips, 2024. Anthropic's public prompt-engineering guidance on long context. Recommends placing authoritative facts near the query, using structured fields over narrative, and giving explicit precedence instructions — principles §2.2 of this paper adopts and §4.5 quantifies.

[2] Anthropic. Use XML tags to structure your prompts. https://docs.anthropic.com/en/docs/build-with-claude/prompt-engineering/use-xml-tags, 2024. Anthropic's public guidance on using XML tags to structure prompts — separating instructions from context from data, with typed field markers. Directly motivates the `SICD_STATE (authoritative...)` typed-field prelude format in §2.2.

[3] Vladimir Karpukhin, Barlas Oğuz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781. Association for Computational Linguistics, 2020. arXiv:2004.04906.

[4] Yaniv Leviathan, Matan Kalman, and Yossi Matias. Fast inference from transformers via speculative decoding. In *Proceedings of the 40th International Conference on Machine Learning*. PMLR, 2023. arXiv:2211.17192.

[5] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems*, volume 33, pages 9459–9474. Curran Associates, Inc., 2020. arXiv:2005.11401.

[6] Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173, 2024. arXiv:2307.03172.

[7] Ken Morkaya. Branch-conditioned resonance retrieval for context-rot mitigation in long-horizon dialog. Technical report, Tree of Mind (ToM) project, tom_master, 2026. Archived Phase 1 retrieval-substrate evidence (4,800 turns, +6.6% relevance, −33.5% stale, +2.7% accuracy). Manuscript is available in the tom_master repository at https://github.com/kenmorkaya-coder/Tree_of_Mind/blob/main/docs/papers/branch_conditioned_retrieval_paper.md and establishes the structural-state retrieval mechanism this paper's Layer 1 implements on a frozen LLM.

[8] Ken Morkaya. Sicd substrate properties: Plastic-curvature memory, axis-based topic specialization, and antifragile growth under sustained load. Technical report, Tree of Mind (ToM) project, tom_master, 2026. Covers the substrate-property validation battery (root-stress absorption, ductile hardening, antifragile growth under oscillating loads, fatigue rhythm under oscillation, and self-tuning against overload), axis-based topic specialization with near-zero cross-domain branch overlap, and plastic κ as the substrate's non-gradient-descent learning primitive. Artifacts and validation logs live in the `tom_master` repository at https://github.com/kenmorkaya-coder/Tree_of_Mind under `sandbox/` and `docs/agi_roadmap/`.

[9] OpenAI. Prompt engineering guide. https://platform.openai.com/docs/guides/prompt-engineering, 2024. OpenAI's public prompt-engineering guidance for GPT models. Recommends explicit instructions, structured prompt sections, clear task framing — principles consistent with Anthropic's guidance and §2.2's state-block format. Cited for cross-vendor scope of the long-context-format claim.

[10] Freda Shi, Xinyun Chen, Kanishka Misra, Nathan Scales, David Dohan, Ed H. Chi, Nathanael Schärli, and Denny Zhou. Large language models can be easily distracted by irrelevant context. In *Proceedings of the 40th International Conference on Machine Learning (ICML)*. PMLR, 2023. arXiv:2302.00093. Quantifies the performance drop LLMs experience when given irrelevant-but-related context alongside answer-bearing content. The published academic analogue of the enumeration-loop failure mode this paper observes at 64K tokens under prose preludes (§4.3, §4.4).