Partitioned versus Undivided Compaction Carriers

A capacity-matched study of verbatim fact retention in a twenty-configuration panel

Aleksandr Tikhonov

Independent researcher

July 28, 2026

Abstract

Long-running language agents often replace earlier dialogue with a recursively updated compact carrier. We test whether a compound partition treatment changes verbatim retention of planted, non-derivable identifiers: one schema field with a 9,600-character allowance versus five named fields with 1,920 characters each. System prompt, input, fold protocol, temperature and output-token budget were held constant. Twenty-three served endpoints were swept; the primary panel contains the twenty whose recorded model-publication dates met the declared six-month rule. The equal-configuration mean within-repeat effect was +0.135 exact recall, the median was +0.027, and configuration effects ranged from -0.411 to +0.929. Excluding three configurations with fewer than four complete pairs reduced the mean to +0.070; planned-repeat missingness bounds were [+0.048,+0.139]. In a separate anchor experiment outside those partition-panel runs, two sweeps gave a +0.700 mean difference. These finite-panel results show substantial heterogeneity, not a universal remedy or a unique mechanism. Fill and rendered-heading analyses are exploratory. We release sanitized evidence, analysis code, a public reconstruction of the load-bearing protocol and a separate public-safe v2 candidate fixture for follow-up experiments. The historical v1 fixture, raw outputs and unrecovered harness patch remain controlled, so the published experiment cannot be rerun exactly.

1 Introduction

An agent that outlives its context window may compact. The design studied here folds the conversation forward: each fold receives the previous running summary plus a new chunk of history, and returns a replacement summary. In this delta-only implementation, nothing re-reads the original transcript. Once an identifier is absent from the only persisted carrier, later turns cannot retrieve it from that transcript. A model may still guess or reconstruct it from other cues, but exact recovery is no longer supported by the stored dialogue state.

That the operation loses facts is established. Zahn and Chana (2026) report summarization destroying 60% of stored facts and cascading compaction eroding 54% of project constraints, and locate the same behaviour across several frontier models; the pre-LLM ancestry runs back to iterative human retelling (Horta Ribeiro et al. 2019). That the shape of a model's requested output changes its behaviour is also established, for single-turn tasks (Tam et al. 2024) and, more precisely, as a capacity phenomenon: Fan (2026) shows that models with headroom pay no penalty for JSON while models near their limit pay a large one, and attributes roughly 87% of one model's penalty to token exhaustion.

What is not established is *which* property of a structured request is responsible. Fan (2026) is explicit about this, and names the gap as future work: "Our gradient varies prompt length, field count, and nesting depth simultaneously; future work should isolate each factor." A gradient that moves field count and nesting and prompt length together cannot say whether a schema hurts because it is a schema, because it is long, or because of how it divides the model's output.

This paper measures a capacity-matched **compound partition treatment** inside an iterative memory fold. The comparison holds declared total character allowance constant: one field permitted 9,600 characters against five semantically named fields permitted 1,920 characters each. The system prompt is byte-identical between the arms (same recorded digest), as are the input, fold protocol, sampling temperature and API output-token budget. Field assignment, repeated local ceilings and the rendered

carrier structure change together; the design estimates their combined effect and does not identify one component as the mechanism.

The treatment also changes later prompts. Although the five-field schema request is larger, its rendered carrier is shorter, so subsequent fold prompts contain less carried text. The partitioned arm sends **1,534 fewer** estimated prompt tokens than the undivided arm pooled over all sweeps, and 2,319 and 3,089 fewer in the two anchor sweeps. This treatment-induced difference is a possible pathway in the total policy effect, not a separately randomized explanation.

Our contributions:

1. A **capacity-matched compound partition comparison** across a fixed panel of twenty served configurations. Matching the declared total character allowance separates the treatment from that quantity, while semantic labels, per-field constraints, rendering and downstream prompt length remain part of the assigned policy.
2. **A heterogeneous fixed-panel effect.** The equal-configuration paired mean is +0.135, the median is +0.027, and effects range from -0.411 to +0.929. Sparse-cell, missingness and model-family sensitivities make the dependence on the measured panel explicit.
3. **A structural manipulation with observational mechanism evidence.** All 134 observed valid five-field rows carried five rendered headings; within model, arm and execution group, rows that happen to carry more headings also tend to recall more. The latter association is not a causal mediation estimate.
4. **An exploratory moderator hypothesis.** Fill fraction separates outcomes under the original post-hoc binary rescue rule, but not robustly under continuous effect thresholds. It is reported as a target for prospective validation rather than as a predictive classification.
5. **Secondary manipulations of declared limits**, including a reduced declared ceiling and a reduced recursive carry cap, with conclusions restricted to the tested settings.
6. **Evidence from the carrier itself.** We record the carried summary, not only its length, which lets us separate facts lost during the fold from facts retained in the carrier but missed by the answer turn, including schema-coerced fabrication (Usman 2026).

We do not claim that structured output is harmful, that partitioning is a general remedy, or that the observed fill and heading patterns identify a mechanism. The contribution is a controlled finite-panel comparison showing that carrier schema assignment can materially change retention, with benefits and costs that vary by served configuration.

2 Related work

Compaction destroys facts. Zahn and Chana (2026) is the closest prior result on the phenomenon and reports it at scale, proposing an external store as the remedy. An (2026) reaches a compatible conclusion from the retrieval side, reporting that verbatim chunks beat lossy artifact extraction by 15.9 points on one long-conversation benchmark (43.9% against 28.0%) and 22.0 points on another (67.4% against 45.4%), and attributing the loss to lossy distillation rather than to structure as such. Our remedy class is different from both: we stay inside the summary carrier and compare two assigned carrier policies. Our result is consistent with An's diagnosis but narrower: carrier partition is one intervention that changes retention under this protocol, not a generally validated fix for lossy distillation.

Capacity. Fan (2026) is the work we position against most carefully. It establishes that the penalty for structured output scales with schema complexity, is not explained by prompt length alone, and largely disappears for models with headroom; it also shows a budget sweep in which truncation falls from 49% to 2% as accuracy rises from 52.5% to 84.0%. Two points of contact matter. First, its complexity axis runs *more fields and deeper nesting is worse*, while our partition axis produces a positive panel mean with effects in both directions. These are not in conflict — Fan measures reasoning accuracy under a structured answer, whereas we measure verbatim survival through an iterated carrier — but the divergence shows that the effect of schema shape is task-dependent. Second, Fan considered and rejected a verbosity account, observing that API JSON mode drops accuracy slightly while emitting *fewer* tokens than chain-of-thought. Any claim of the form "structure induces verbosity" must therefore be argued against a published negative result; we do not make that claim, and our capacity-matched design does not rest on it.

Serialisation cost. A line of work measures the token overhead of notation itself. Kutschka and Geiger (2026) benchmarks token-optimised alternatives to JSON across four agent benchmarks and five models,

reporting reductions up to 27% and 18% against accuracy tradeoffs. That quantity is serialisation overhead — the cost of the syntax. Ours is realized content volume — how much material the model elects to write into the slot it is given. The quantities are distinct; this study does not establish their statistical or causal independence.

Structured representations inside an iterative update. Hwang et al. (2024) is the closest prior work on our mechanism, and it reaches a conclusion that must be engaged rather than filed away. It updates a structured JSON knowledge representation in place as new sources arrive — rather than rebuilding it each time, which is our fold — and reports that doing so beats an unstructured-memory baseline by 40% and 14% across two datasets, with a further 7% and 4% from dynamically augmenting the representation. Comparing that to our own arms is harder than it looks: the contrast that would speak to it directly pairs our unheaded plain arm against our decomposed JSON arm, and those two differ in system prompt as well as format (Section 4.7), so we decline to read an ordering off it.

We do not think these conflict, and we do not claim to overturn it. Their comparison is structured JSON against an undifferentiated memory, evaluated with LLM-assisted precision, recall and macro-F1 over attribute-value summaries on SUMIE and an LLM-based error metric on the book task. Ours compares assigned recursive carrier policies using label-bound exact identifier matching. The anchor policy contrast raises recall by 0.700 at matched declared total allowance, but it jointly changes semantic fields, local ceilings and rendering. The studies therefore differ in intervention, task and metric; neither supports "JSON is better" or "plain text is better" as a general rule.

Related systems broaden the structured-memory design space. PRISM uses typed hierarchical schemas and programmatic revisions for token-efficient long-range reasoning (Jayalath et al. 2025). Structured Distillation compresses exchanges into a four-field retrieval object while retaining a verbatim source for drill-down (Lewis 2026). StructMem combines event bindings, temporal anchors and periodic consolidation for long-horizon conversational memory (Xu et al. 2026). These works motivate structured or decomposed memory; the present study contributes a narrower capacity-matched comparison of two recursively replaced carriers with exact-identifier grading.

The mechanism itself has standard references, and we use it rather than propose it. Wang et al. (2025) is the canonical statement of recursive summarisation for dialogue memory — previous memory plus new context yields new memory — which is the fold this paper measures; Packer et al. (2023) is the standard systems treatment of tiered virtual context around it. We contribute nothing to the mechanism and study only what its summariser is asked to emit.

Why a bespoke fixture rather than a public benchmark. LongMemEval (Wu et al. 2024) and LoCoMo (Maharana et al. 2024) are the obvious substrates, and we do not use them. The reason is the dependent variable: this study needs facts that are non-derivable and graded by exact string, planted at known positions so the fold that lost one can be identified, whereas those benchmarks include judge- or overlap-based outcomes that permit paraphrase. This limits direct comparability.

Length regulation under iteration. Adams et al. (2023) establishes that summary length is controllable independently of information content, by explicit instruction. Chang et al. (2024) characterises running-summary folding directly, and is the canonical prior on the incremental-update regime we operate in. Under the tested carrier policies, partitioning is associated with shorter realized output even without an additional length instruction. Because realized length is post-treatment, this does not identify a general length-regulation mechanism.

Structure and fabrication. Usman (2026) demonstrates across thirteen models that required schema fields coerce fabrication, driving it to 100% in ten of them, and that an explicit "insufficient evidence" escape is largely ineffective for open-weight models. Our recall turn asks the model to list values, which is itself a form demanding an answer, so the fabrication we observe is best read as this phenomenon occurring *downstream of compaction loss* rather than as a new failure mode.

Kwon (2026) makes the closely related point for memory specifically, and states the danger more sharply than we would have: a memory that keeps a wrong conclusion but drops the work behind it leads a model to re-emit the stale value confidently, "where an empty memory leads it to abstain" — studied under a fixed token budget, as ours is. We take that as established and do not re-claim it. What our setup adds is that we record the carrier as well as the answer, so we can show the fact was absent from the carrier rather than inferring it, and can attribute the loss to the fold rather than to the answer turn.

One candidate mechanism for *which* values get invented comes from Parikh (2026), which finds that

requesting JSON collapses answer diversity toward the modal response across 44 models — the modal answer's share rises from 41% to 64%, distinct answers fall from 52 to 36. That paper concerns valid answer spaces rather than missing information, so the connection is a conjecture rather than a result. But it would explain the specific character of what we observe: the substituted values are not arbitrary, they are the highest-frequency instance of their type — port 8080, `example.com`, the RFC 4122 example UUID.

Causal caution. Yuan et al. (2025) reports that apparent format effects largely vanish under causal analysis, finding no causal impact in 43 of 48 scenarios, and enumerates the collider structures that produce spurious format effects. Conditioning on a post-treatment variable — such as whether a summary hit its cap — is exactly the error it warns about. Our primary result is the assigned policy contrast; we report realized fill and headings separately as observational, post-treatment analyses.

Structured memory. Contrary to a simple "plain text wins" reading, Sharma et al. (2026) finds JSON extraction reaching 0.96 feasibility against 0.48 for narrative hand-off, *despite* narrative producing the smallest payload — which by itself refutes "shorter is better". Petrov et al. (2026) reports strong results for schema-grounded memory. Our data are compatible with this direction while also showing losses under partitioning for some served configurations.

3 Method

3.1 The fold

A synthetic session of 150,000 tokens is divided into chunks and folded run by run. Each fold receives the previous running summary and one chunk, and returns a replacement. Nothing re-reads the transcript. After the final fold, a recall turn asks for fourteen planted facts by label, and the answer is graded against the recorded values.

Fifty facts are planted; **fourteen are graded**, and the two sets are not the same. The graded values are two ports, a filesystem path, a shard count, a UUID, three UTC timestamps, a cache origin URL, a DNS zone, a signing-key id, an on-call handle, a release tag, and one Cyrillic department name. Cluster ids, tenant codes, licence keys and image digests are planted but never asked.

For row (i), exact recall is (Y_i=c_i/14), where (c_i) is the number of labelled checks passed. Each check requires the expected value as a whole token on the answer line identifying that label; a line containing another graded value also fails. Text comparisons preserve case unless the check explicitly declares otherwise. Repeated appearances do not add credit: each of the fourteen checks is binary. None of the expected values can be reconstructed from another, a range or a pattern. Four include a unit word (`19 shards`, and three timestamps), so a response retaining only the number fails. This deliberate hard-case outcome scopes the claim to verbatim retention; see Section 6.

3.2 Arms

Arms vary the policy under which the summariser writes and the runtime carries its replacement memory. Within the central comparison, the system prompt, input, fold protocol, temperature and API output-token budget are held fixed.

Arm	Representation	Declared capacity
`f`	Deployed shape: shared schema, one `summary` string	1,024 chars
`n`	Same schema, ceiling lifted	9,600 chars
`j`	Dedicated schema, `summary` field only	9,600 chars, 1 field
`p`	As `j`, with an expanded field description	9,600 chars, 1 field
`q`	Decomposed schema, five named string fields	1,920 chars x 5 = 9,600
`k`	Semantic schema, several named fields	—

Arm	Representation	Declared capacity
`o`	Plain text, no headings	—
`m`	Plain text under five fixed headings	—
`t`	As `j`, declared ceiling halved	4,800 chars, 1 field
`u`	As `m`, carry cap reduced 2,400 -> 1,400 tokens	—
`l`	Plain text under the same five headings, at the deployed 900-token budget	—
`g`	The deployed shared schema, parsed rather than carried raw	1,024 chars

The `j` to `q` contrast is the paper's centre. `q` declares five semantic fields whose local ceilings sum to the 9,600-character total declared for `j`. It then deterministically renders those fields under five headings into one carrier. The treatment therefore combines semantic assignment, five non-transferable local ceilings and rendered structure; it does not isolate any component.

We distinguish five quantities that can otherwise be conflated as "capacity": **declared total character allowance** (9,600 in both central arms); **declared per-field allowance** (9,600 once versus 1,920 five times); **API output-token budget** for each fold response; **recursive carry cap**, the runtime token allowance applied before the next fold; and **realized carrier characters**, whose ratio to declared total allowance is the observed fill fraction. Estimated prompt-token usage is recorded separately and includes the treatment-induced carrier length. Matching the first quantity does not match or randomize the others.

3.3 Execution and validity

Arms are interleaved within each sweep on a rotating Latin square, so every arm leads equally often and execution position is balanced against endpoint drift. This balances position but **not** first-order carryover — each arm has the same single predecessor in most repeats — and a Williams design (Williams 1949) would be required for that. One sweep's rotation does not close, and is disclosed where used.

A single validity rule decides whether a row may enter any aggregate; every reported statistic reads that one rule. Rows invalidated by infrastructure faults may be retried; rows invalidated by an arm's own behaviour may not, since retrying those would select for the arm succeeding.

For the capacity-matched `j`/`q` study, one experimental unit is a complete arm-specific fold-and-recall trajectory on S5, identified by served configuration, execution group and planned repeat. The paired block is the same triple without the arm; the fourteen graded facts are measurement components inside that trajectory, not experimental units. Cross-configuration summaries give each served model/provider/quantization configuration equal weight regardless of valid repeat count. These twenty configurations are the finite target panel and the generalization units; they are not a random sample from a population of recent models.

The **primary** outcome is the continuous within-repeat difference in exact recall, `q-j`, summarized first within served configuration and then with equal weight across configurations. The primary complete-pair estimate uses only blocks with valid outcomes in both arms; planned-repeat bounds allow each missing outcome to range over [0,1], including blocks where neither arm is valid. Full recall and recall at or above 0.5 are **secondary operational thresholds**. The `q`-versus-`m` plain-heading control and `j`-versus-`t` reduced-ceiling comparison are secondary. Threshold-defined rescue groups, fill, rendered headings, fabrication, carrier inspection and cross-scenario comparisons are **exploratory**. The 0.5 and 1.0 thresholds are descriptive operating points, not optimized confirmatory endpoints.

3.4 Models and provider pinning

Nine configurations carry the broad supporting-arm sweep, selected for open weights where available, recency, small-to-medium size and family diversity. Hosted models are reached through a gateway that routes across multiple endpoints per model. Left unpinned this is a **validity** defect and not merely a cost one: endpoints for one model differed in quantization (bf16, fp8, fp4) and in whether they honour

strict schema requests at all, so an unpinned sweep can silently vary the treatment. Every sweep pins one endpoint with fallbacks disabled, and records the pin, its quantization, its context length and its price.

For the capacity-matched partition study, 23 served endpoints were swept. The primary table applies the recorded publication-date rule—publication on or after 28 January 2026, six months before the study date—and contains 20 configurations. Three measured configurations fall outside the window and remain in the evidence package but not the primary panel. The cutoff is a declared analysis rule, but the available records do not establish that it was timestamped before outcome inspection. Publication dates and aliases come from the preserved provider-catalog snapshot, so the panel is a dated convenience sample rather than a probability sample of model families or checkpoints.

4 Results

> **Note**
>
> Recall is effectively binary at the row level: rows land at or near 0 or 1. Over all 620 valid rows of this scenario, a one-way random-effects decomposition of the fourteen binary checks gives **ICC = 0.750** and a design effect of **10.745**, so a fourteen-check row is worth about **1.30** independent observations rather than fourteen. Cells are therefore reported with their denominators wherever a table has room, and where a table gives means without them — the partition table, the iteration table and the failure-mode table — the per-cell row counts are in the released package rather than the page; the fourteen checks within a row are not independent trials (Miller 2024). The figures above are recomputed by `generate.py` from the released row- and check-level artifacts.

4.1 The deployed shape fails on every measured supporting configuration

The starting point was a deployed configuration: a shared schema carrying one `summary` string with a 1,024-character ceiling. It fails on all eight models on which it was measured, while plain text under fixed headings succeeds on all nine — but at 2,400 fold tokens against `f`'s 900, so that pair is not budget-matched. The arm that *is* matched to `f` at 900 tokens is `l`, which also succeeds on all nine (0.694-0.929); that is the comparison to read.

Table 1: Verbatim recall on the distributed-facts scenario, by model and arm. Cells are means over three to seven valid rows, with per-cell denominators in `s5-recall-by-model-arm.csv` of the released package; rows are effectively binary, so a mean here stands for a small count. The `qwen3.6-flash` row comes from the one sweep whose rotation does not close (2 arms over 7 repeats, so `l` leads four times and `m` three); every other sweep closes. The rotation is also a pure cycle, so first-order carryover is confounded with arm throughout.

Model	`f` deployed	`n` shared schema, ceiling lifted	`k` named-field JSON	`l` plain, 900 tok	`m` plain, 2,400 tok
Qwen3.6 (local reference)	0.051	0.061	0.969	0.725	0.969
qwen3.6-35b-a3b	0.000	0.051	0.878	0.735	0.990
qwen3.6-flash	—	—	—	0.786	0.990
tencent/hy3	0.000	0.857	0.816	0.694	0.786
z-ai/glm-5.2	0.086	0.000	0.800	0.929	1.000
minimax/minimax-m3	0.000	0.750	1.000	0.871	1.000
mistral-small-2603	0.257	0.786	1.000	0.829	0.986

Model	`f` deployed	`n` shared schema, ceiling lifted	`k` named-field JSON	`l` plain, 900 tok	`m` plain, 2,400 tok
deepseek-v4-flash	0.071	0.171	0.929	0.929	1.000
xiaomi/mimo-v2.5	0.000	0.000	0.857	0.857	1.000

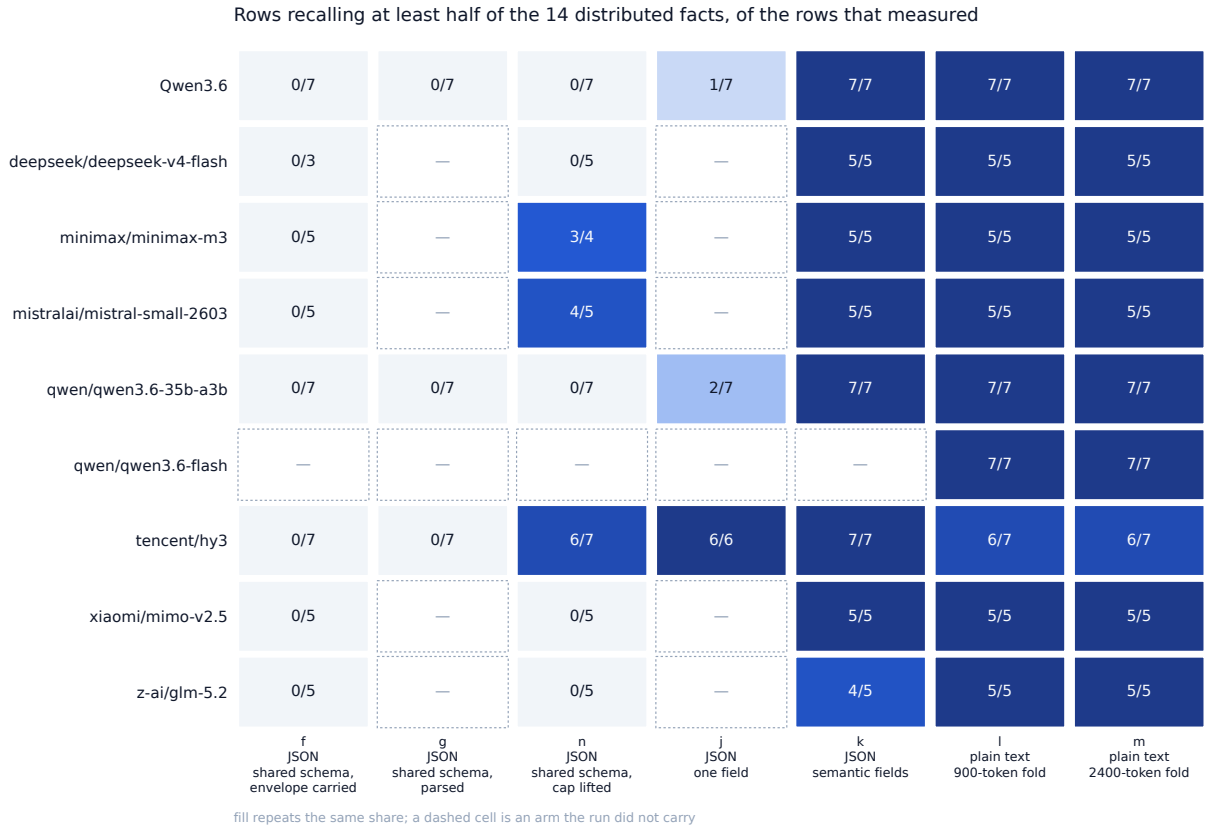

Figure 1: Rows recalling at least half of the fourteen distributed facts, by arm and model. A dashed cell is an arm the run did not carry.

Two cautions on reading Table 1. First, `f` differs from `m` in more than one respect — schema, ceiling and budget all change — so it does not isolate anything; its 1,024-character ceiling cannot hold the material regardless of shape, and it is reported as a floor result rather than as an ablation. Second, `n` is the arm that shows how model-dependent the schema penalty is, ranging from 0.000 to 0.857 across models. That variance is the reason the isolating comparisons below hold the model fixed.

Named-field JSON (`k`) scores 0.80 to 1.00 across this supporting set. Its descriptive mean trails `m` by 0.066; the public package does not reproduce a formal equivalence analysis, so no equivalence claim is made.

4.2 Capability probes measure an endpoint and request, not a model alone

Three configurations — `granite-4.1-8b`, `ling-2.6-flash` and `trinity-large-thinking` — produced no measurable rows in an initial capability probe.

The recorded verdicts are `probe_rejected_no_endpoint_for_parameters` for the first two and `probe_rejected_reasoning_not_disableable` for the third — routing and parameter-support failures, not schema-compliance failures. Neither of the first two was ever sent a schema: only the two plain-text arms were attempted on them, and those arms put no `response_format` on the request at all.

The observed schema non-conformance in this study is `qwen/qwen3.6-flash`, recorded as `schema_advertised_reply_nonconforming`: it advertises structured outputs, returned JSON, and the JSON omitted a required property. Its sweep still ran and still measured, and it is included in Table 1 on the plain-text arms, which send no schema.

The lesson is about instruments, not models. A capability probe routed through an aggregator measures the conjunction of the model, the endpoint, and the exact parameter set the harness happens to send. Such a verdict must not be reported as a property of the model alone.

4.3 Anchor endpoint: a large partition effect and a plain-heading control

The four arms `o`, `m`, `j` and `q` do not form a clean format-by-partition 2x2. `o` is the only arm that carries a **different system prompt** (`unstructured_prose`, digest `5ebf263a...`) from the other three (`fixed_headings`, digest `62a20557...`). Any contrast involving `o` therefore moves the instruction as well as the representation, and the two cannot be separated in that comparison.

Three arms do share one system prompt exactly and form an assigned-policy ladder (Table 2):

Table 2: Three arms that share one system prompt. Each rung changes an assigned carrier policy: `j` to `q` is the compound partition treatment, while `q` to `m` also changes schema enforcement and rendering.

rung	change	S5 recall	effect
`j`	one opaque field, ceiling 9,600	0.071	—
`q`	the same 9,600 split into five named fields	0.771	**+0.700**
`m`	the same five sections, schema removed	0.986	**+0.214**

The second rung deserves the same standard we apply elsewhere. `q` is a large improvement on `j` and it is **not** a return to `m`: it reaches full recall on **1 of 10** rows against `m`'s **8 of 10** (seven pairs are full only under `m`, one under both and two under neither; two-sided exact paired McNemar/binomial **p = 0.0156**), a gap of 0.214. That is more than three times the `k-m` difference this paper declines to call equivalence, so we do not call `q` a recovery either. On this endpoint, the partition treatment accounts for most of the observed ladder difference, while the plain-heading arm performs better still. Because `q` versus `m` also changes schema enforcement and rendering, this is a policy contrast rather than a decomposition of a unique mechanism.

On this anchor endpoint, the capacity-matched partition treatment increases mean exact recall from 0.071 to 0.771 across two sweeps (`n = 10` paired rows), a difference of **+0.700**. The subsequent plain-heading control reaches 0.986. These are endpoint-specific contrasts under this protocol; the cross-configuration effects below show that neither direction nor magnitude generalizes uniformly.

The `o` arm, for completeness and with its confound restated, scores 0.921 on the same sweeps. Read against `j` it would suggest that removing the schema is worth more than partitioning; read against `m` it would suggest partitioning in plain text is worth almost nothing. Neither reading is available, because `o` also changes the system prompt. We report the number and draw nothing from it.

A third arm adds a 1,835-character natural-language description to the single field and scores 0.016. This does not isolate comprehension. If the description reached the model on all five folds, the arm's prompt should exceed the undivided arm's by roughly **2,294 tokens**, and the observed difference is **+298** — most of which is accounted for by this arm carrying a longer summary anyway. The arm therefore bounds nothing about comprehension, because we cannot show that the full description arrived.

On which measurement of `j` the ladder uses. The 0.071 above is the undivided arm as measured in the two decomposition sweeps (n = 10), which is where `q` was also measured; a contrast has to come from one design. The same arm on the same model over the larger anchor corpus scores **0.214** (n = 28), and over every model and sweep in this study its mean is higher still. The ladder's +0.700 is therefore a

within-design effect, not a claim that 0.071 is the arm's characteristic value. Section Section 4.6 gives the range across models, which is wide and is the point.

4.4 Exploratory scenario comparison is consistent with amplification under repeated folding

The same ablation run on a scenario whose facts all arrive in a single chunk, rather than distributed across folds, separates a mild effect from a severe one:

Arm	Facts in one chunk	Facts distributed across folds
`j` undivided JSON	0.300	**0.071**
`p` undivided JSON, described	0.300	**0.016**
`q` decomposed JSON	1.000	0.771
`o` plain, no headings	1.000	0.921
`m` plain, headings	1.000	0.986

This comparison moves more than iteration, and the difference is not small. The single-chunk scenario grades **7** of its 28 planted facts; the distributed one grades **14** of 50, and uses a different grader and a different set of planted values. So the contrast above is between two scenarios, not between two levels of one factor, and the "fourteen graded facts" stated throughout this paper describes the distributed scenario only. The grader change at least is empirically inert here: the label-bound and presence counts agree on **619 of 620** valid distributed rows.

Recall is lower in the deeper-fold scenario for every displayed arm. Because the scenarios also differ in injected facts, grading composition and prompt content, this comparison is consistent with, but does not isolate, amplification by iteration. A factorial study holding the fixture and grader fixed while varying fold depth would be needed for a causal iteration estimate.

4.5 Reducing the tested declared ceiling did not rescue the undivided carrier

This secondary comparison changes the declared per-field character allowance while retaining the undivided schema.

Arm	Change	n	Rows recalled	Mean	Carried chars	Folds clipped at cap
`j`	ceiling 9,600	11	3	0.325	7,818	17
`t`	ceiling **4,800**	12	1	0.131	4,697	0
`m`	plain	4	4	1.000	6,269	0
`u`	carry cap **1,400**	4	4	1.000	5,505	13

Halving the declared ceiling halves the carried text and does not recover recall; the point estimate moves in the *opposite* direction to the capacity prediction. Restricting the calculation to the three ceiling runs gives **7/15** valid `j` rows against **3/16** valid `t` rows at or above half recall. Fifteen repeats have both outcomes: five cross the threshold only under `j`, one only under `t`, two under both and seven under neither (two-sided exact paired McNemar/binomial $\mathbf{p = 0.2188}$). One further `t` row, itself below the threshold, has no valid `j` partner. Treating that missing `j` outcome as a failure leaves $p = 0.2188$; treating it as a success gives $p = 0.125$. The failing arm carries less text than either high-recall arm. Assigning a reduced carry cap to the plain-heading representation — which clips it on 13 of 20 folds, the most clipping of any arm — leaves recall at 1.000 in four rows, while `j`, clipped on 17 of about 53 folds, recalls 0.325 in eleven rows. The carry-cap comparison is a small descriptive robustness check, not evidence of capacity independence.

The declared ceiling is not inert, however. In these runs it is associated with whether the model returns schema-valid output: `j` fails to produce a conforming fold on 4 of 60 folds (6.7%), while `t` fails on none.

Halving the ceiling improved well-formedness without improving fact retention, which rules out a simple monotonic account in which the larger declared character ceiling alone caused the anchor failure. It does not exclude pathways through realized output length, prompt length, the API token budget or recursive carry cap.

4.6 The reduced-ceiling direction appears on two endpoints

Repeating the ceiling manipulation on a second model (`z-ai/glm-5.2`, four repeats per arm) gives a result worth reporting in full because half of it does not transfer.

Model	`j` ceiling 9,600	`t` ceiling 4,800	`j` fill of its ceiling
qwen3.6-35b-a3b	6/28 rows	1/12 rows	81%
z-ai/glm-5.2	**4/4 rows**	2/4 rows	**56%**

On both tested endpoints, the lower-ceiling arm is at best equal to the full-ceiling arm. The same observed direction on two endpoints weakens a simple monotonic larger-ceiling explanation, but the paired contrast is imprecise (two-sided exact p = 0.2188) and does not establish a general capacity result.

What does not: the collapse of the undivided arm is not universal. On `z-ai/glm-5.2` the undivided schema arm recalls perfectly.

In a broader exploratory dataset of **twenty-five measured configurations**, undivided-arm fill and recall have Pearson r = -0.225 and Spearman rho = -0.426. This population differs from the twenty-configuration primary panel and does not support fill as a general one-dimensional explanation.

The examples span both ends of the fill scale:

model	rows	carried chars	of a 9,600 ceiling	recall
`p`, described field	9	9,044	94%	0.016
qwen3.6-35b-a3b	28	8,501	89%	0.214
deepseek-v4-pro	4	4,740	49%	**1.000**
granite-4.1-8b	16	3,238	34%	0.250
ling-2.6-flash	16	621	**6%**	0.000

Some rows fill the slot with prose describing the session rather than preserving identifiers; others write a short gist — `ling-2.6-flash` carries 621 characters on average summarising which maintenance steps passed, and not one identifier. Healthy carriers also occur between these examples. This makes fill an exploratory marker rather than a standalone explanation.

4.7 Partitioning across twenty configurations: a heterogeneous panel effect

The ladder above is one endpoint. The primary capacity-matched panel (Table 3) gives a more heterogeneous picture. The primary panel is formed only from a row in `partition-by-model.csv` marked inside the declared recency window, a configured partition run containing both `j` and `q`, and at least one complete within-repeat pair. This admits exactly the twenty configurations in Table 3. It excludes the three swept out-of-window configurations `meta-llama/llama-3.2-3b-instruct`, `mistralai/ministral-14b-2512` and `qwen/qwen3-235b-a22b-2507`; the local reference and the `qwen/qwen3.6-35b-a3b` decomposition anchor are not partition-panel runs.

Table 3: The capacity-matched partition contrast in the fixed 20-configuration panel, whose configurations meet the recorded 6-month rule as of 2026-07-28. Effects pair valid rows by execution group and repeat; the displayed arm means use all valid rows and therefore need not equal the paired contrast when a partner is missing. `j` fill is the share of the declared 9,600-character ceiling used.

Model	j mean	q mean	Complete pairs	Paired q-j	j rows	q rows	j fill
deepseek/deepseek-v4-flash	0.161	0.947	4	+0.786	4	4	81%
deepseek/deepseek-v4-pro	1.000	1.000	4	+0.000	4	4	49%
google/gemma-4-26b-a4b-it	0.947	0.857	4	-0.089	4	4	69%
google/gemma-4-31b-it	1.000	1.000	4	+0.000	4	4	63%
ibm-granite/granite-4.1-8b	0.304	0.250	12	-0.054	12	12	36%
inclusionai/ling-2.6-1t	0.929	0.929	4	+0.000	4	4	61%
inclusionai/ling-2.6-flash	0.000	0.161	12	+0.161	12	12	6%
minimax/minimax-m3	0.643	0.875	4	+0.232	4	4	93%
mistralai/mistral-small-2603	0.982	0.929	4	-0.054	4	4	71%
nvidia/nemotron-3-super-120b-a12b	0.982	0.589	4	-0.393	4	4	69%
nvidia/nemotron-3-ultra-550b-a55b	0.875	0.929	4	+0.054	4	4	60%
qwen/qwen3.5-122b-a10b	0.000	0.714	4	+0.714	4	4	100%
qwen/qwen3.5-35b-a3b	0.000	0.625	3	+0.619	3	4	100%
qwen/qwen3.5-397b-a17b	0.625	0.821	4	+0.196	4	4	79%
qwen/qwen3.5-9b	0.482	0.661	4	+0.179	4	4	81%
qwen/qwen3.6-27b	0.750	0.536	4	-0.214	4	4	68%
tencent/hy3	0.881	0.661	3	-0.024	3	4	62%
xiaomi/mimo-v2.5	0.000	0.929	1	+0.929	2	1	99%
xiaomi/mimo-v2.5-pro	1.000	0.589	4	-0.411	4	4	58%
z-ai/glm-5.2	0.487	0.518	11	+0.078	11	12	59%

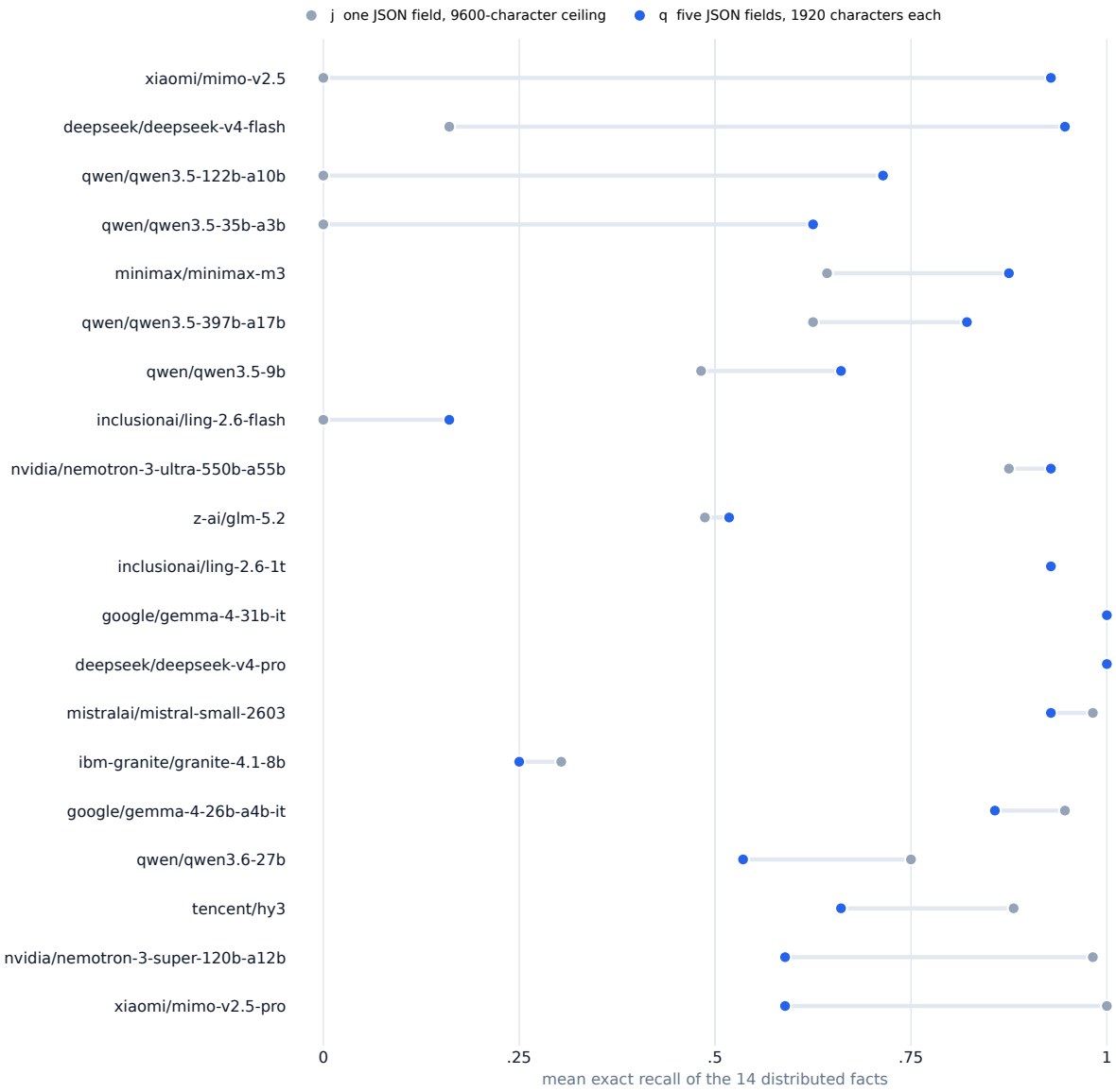

Figure 2: The same contrast drawn: each model's undivided arm against its partitioned one, at identical declared capacity.

The primary estimand is continuous: for each configuration we average `q-j` over complete within-execution-group, within-repeat pairs, then weight the twenty configurations equally. The panel mean is **+0.135**, but the median is only **+0.027**, the IQR is **[-0.054,+0.205]**, and the range is **[-0.411,+0.929]** (10 positive, 7 negative and 3 zero). These are summaries of this fixed convenience panel, not estimates for a superpopulation of recent models. Leaving out one configuration at a time moves the mean between +0.094 and +0.164. A t-style model-dispersion interval conditional on this panel is [-0.036,+0.307]; it is not a confidence interval for recent models.

Sparse cells matter. Three configurations have fewer than four complete pairs; excluding them nearly halves the equal-configuration mean:

population	configurations	equal-configuration paired mean	median
at least one complete pair	20	+0.135	+0.027
at least four complete pairs	17	+0.070	0.000

population	configurations	equal-configuration paired mean	median
equal valid-arm counts	16	+0.069	0.000
exclude denominator-one `mimo-v2.5` only	19	+0.094	0.000

The planned-repeat sensitivity includes invalid outcomes and repeats where neither arm is valid. Letting each absent outcome range over [0,1] identifies the equal-configuration panel effect only to **[+0.048,+0.139]**. Equal weighting across eleven declared model families gives +0.091, and leave-one-family-out means range from +0.060 to +0.117; the corresponding family-dispersion interval conditional on these eleven families is [-0.030,+0.211]. Thus the panel-average direction is stable under these deterministic checks, while its magnitude depends strongly on sparse configurations.

4.7.1 Exploratory threshold and fill analysis

The original binary summary calls `j < 0.5` a failure and `q >= 0.5` a rescue. Under that post-hoc rule, `j` fails on 8 of 20 configurations and `q` crosses the threshold on 6 of those 8. Ordering these eight by fill fraction separates the threshold outcomes:

original threshold class	fill of the 9,600-char ceiling	configurations	crossing 0.5 under `q`
higher-fill `j` rows	59%, 81%, 81%, 99%, 100%, 100%	6	**6 of 6**
lower-fill `j` rows	6%, 36%	2	**0 of 2**

This separation is descriptive and threshold-dependent. Among the eight `j < 0.5` configurations, fill versus paired effect gives Pearson r = +0.747 and Spearman rho = +0.714; excluding the two configurations within 0.02 of the failure cutoff gives +0.855 and +0.429. Across all twenty configurations the corresponding associations are +0.542 and +0.589.

Varying the `j` failure cutoff from 0.40 through 0.60 preserves the original threshold separation. Changing the outcome from "`q` crosses 0.5" to the continuous paired effect does not: 7 of 8 have a non-negative effect, 6 of 8 gain at least 0.10, and 4 of 8 gain at least 0.20; fill does not cleanly separate the first two definitions and has a margin of only 0.002 for the third. The binary pattern is therefore a post-hoc hypothesis for prospective validation, not a predictive rule or evidence of zero effect in low-fill configurations.

Each effect is paired within its declared execution group. A model ID can appear in unrelated sweeps, so the public configuration binds every partition run to one execution group and the generator checks its valid counts and means against the aggregate table before pairing by repeat.

The borderline cases, and the separation with and without them. Two models sit within 0.02 of the threshold on the undivided arm and cannot be confidently classed either way. `glm-5.2`, re-run at twelve rows per arm, gives `j` 0.487 and `q` 0.518 — a gap of +0.031, with 5 of 11 and 6 of 12 rows above half. `qwen3.5-9b` gives `j` 0.482. Calling either a failure or a success is an artifact of where the line is drawn.

Under the original binary rule, the separation does not depend on that call:

classification	rescued fills	non-rescued fills	margin
all models	59, 81, 81, 99, 100, 100	6, 36	23 pts
excluding both borderline models	81, 99, 100, 100	6, 36	**45 pts**

Dropping the ambiguous models widens the margin under that rule, but it does not validate the selected covariate, direction, cut point or outcome definition. We therefore do not attach a p-value to this post-hoc separation.

Deeper repeats for the two original non-crossing configurations. Both were repeated at twelve rows per arm rather than four:

model	arm	n	mean	rows >= 0.5	fill	headings carried
granite-4.1-8b	`j`	12	0.304	6	36%	0 on 10 rows, 5 on 2
granite-4.1-8b	`q`	12	**0.250**	**0**	**20%**	5 on all 12
ling-2.6-flash	`j`	12	0.000	0	6%	0 on 11 rows, 1 on 1
ling-2.6-flash	`q`	12	0.161	2	13%	5 on all 12

Neither crosses the original rescue threshold at depth. The paired continuous effects are +0.161 for `ling-2.6-flash` and -0.054 for `granite-4.1-8b`; the latter also writes less under five enforced fields. These two configurations illustrate heterogeneity but do not establish distinct model classes.

Its undivided arm is also the clearest within-model case for Section 4.8: the 14 rows that carried no headings average 0.194, the 2 that carried five average 0.643. The model does better when it happens to partition, and worse when the partition is imposed — which is a real tension and we do not resolve it.

The depth runs reduce repeat-level noise for these two configurations. They do not turn the exploratory fill split into a general classification; the positive continuous effect on `ling-2.6-flash` also contradicts a uniform no-benefit interpretation for low-fill carriers.

4.8 Exploratory association between rendered headings and recall

The arms analyzed here receive the **same** system prompt asking for the same five named sections — the recorded digest is identical across `j`, `q`, `m`, `p`, `t` and `u`. What differs is whether the carrier that comes back is actually divided. The harness records this per row as `carried_fixed_headings`, precisely so that "the prompt asked for five headings" is never mistaken for "five headings arrived".

The raw single-field-schema breakdown is:

headings that arrived	rows	mean recall	rows >= 0.5
0	38	0.086	5
1	58	0.041	2
2	5	0.457	2
3	5	0.529	3
4	11	0.610	7
5	69	**0.900**	64

Over all 570 valid rows where the count was recorded, the pooled correlation is $\mathbf{r = +0.606}$. This pools arm, model and sweep differences and is only a raw descriptive statistic; even the table is not monotone at the bottom, falling from 0.086 at zero headings to 0.041 at one.

What was observed under the renderer. When a conforming five-field response is parsed, the renderer deterministically emits five headings; the published fail-safe may instead carry raw text after a parse failure. All **134 of 134 observed valid `q` rows** carried five sections, as did 39 of 39 `k` rows. A single field leaves division to the model, and configurations differ sharply in whether it appears: of `j`'s 161 rows, 37 carry no section, 40 carry one, and 69 carry all five.

The model-specific patterns are consistent with headings as one marker of carrier behaviour. For example, `deepseek-v4-pro` and `gemma-4-31b-it` both carry five sections on 4 of 4 rows and recall 1.000. The models that partitioning helps most under the original binary rule include configurations that do not partition spontaneously: `deepseek-v4-flash` carries one section per row and recalls 0.161, while the same configuration given five enforced fields carries five and recalls 0.947.

The marker is not sufficient: `ling-2.6-flash` and `granite-4.1-8b` receive all five headings under `q` — on every row — and still recall 0.161 and 0.250. Enforcing the partition changes the carrier as assigned, but the observed recall response remains heterogeneous.

Stated as an association, deliberately. Headings carried is a post-treatment variable, and conditioning on it causally would be exactly the error Yuan et al. (2025) warns of. After removing a separate mean for every model, arm and execution-group cell, 31 informative cells (178 rows) give a recall slope of +0.120 per additional heading and residual r = +0.557. Restricting to single-field schema arms gives 17 cells, 100 rows, slope +0.134 and r = +0.597; `j` alone gives 14 cells, 87 rows, slope +0.124 and r = +0.576. Within these cells, rows that happened to carry more requested headings also tended to have higher recall. This is descriptive mechanism-consistent evidence, not a causal mediation estimate; the five-field arm has no within-arm heading variation with which to estimate mediation.

4.9 What the carrier actually contains

The released scan-selected excerpts illustrate two carrier styles, but not within one endpoint configuration. A `qwen/qwen3.6-35b-a3b` undivided excerpt writes *about* the session—stating that "the assistant has recorded 50 configuration decisions so far"—without carrying the listed values. A `deepseek/deepseek-v4-flash` five-field excerpt enumerates literal identifiers under headings. These qualitative examples motivate inspection; they do not identify the treatment mechanism.

In the reduced-ceiling arm, realized output length is consistent with sensitivity to the declared limit rather than mechanical truncation.

It is not a guillotine. Across every fold in the halved-ceiling arm, **80 of 80 calls finished with `stop`**: none hit the output bound and none was truncated, while the carried text landed within 0.5% of the declared ceiling on 14 of 16 rows. The model wrote to the offered ceiling and stopped there of its own accord. The undivided arm at the full ceiling behaves the same way on the large majority of its calls — 694 `stop` against 20 that hit the length bound, the latter being the schema-invalid folds reported above.

These observations show that the tested lower ceiling changed realized output without API truncation. They do not establish that models generally treat declared ceilings as writing targets.

4.10 Audited carriers show additional loss at the answer turn

Because the carried summary is recorded, not merely measured, we can ask of every row how many graded values are present in the carrier and how many the answer then gets right. Over **215 valid rows** the two agree on 171.

In all **44 rows with unequal carrier-presence and answer-recall counts**, the carrier count was higher. No audited row contained a graded identifier answered correctly while absent from the retained carrier.

Table 4: Row-level carrier/answer discrepancies in the 215-row audit; the third column is a count of rows with one or more discrepancies, not a count or fraction of facts.

arm	audited rows	rows with at least one carrier-present, answer-missed discrepancy
`q` five named fields	88	**31 (35.2%)**
`j` one opaque field	103	13 (12.6%)
`m` plain, five headings	4	0
`t` halved ceiling	16	0
`u` reduced carry cap	4	0

This is an empirical one-way discrepancy in the 215 audited rows, not a logical bound: guessing or reconstruction from other cues remains possible. Downstream reading nevertheless adds another measured failure surface. At least one carrier-present identifier was missed in **31 of 88 `q` rows** and **13 of 103 `j` rows**. Those are row frequencies, not the fraction of carrier-present facts lost. The measured partition effect is therefore a net result across fold-stage retention and answer-stage reading under this protocol.

4.11 Exploratory answer-prefix fabrication and abstention

Classifying every failing row (recall < 0.5) by whether its answer contains any abstention marker at all:

arm	fabricates with no abstention	abstains somewhere
`j` one opaque field	23	64
`q` five named fields	4	48
`n` shared schema	12	20
`t` halved ceiling	8	5
`f` the deployed shape	**1**	**44**

So silent fabrication is real but it is a minority behaviour overall, and it is concentrated in the undivided-schema arms. The deployed configuration this study began with does the opposite: in 44 of its 45 failing rows the model says it does not know. That is a loud failure, and a system that logs abstention would have caught it.

Where fabrication does occur it has the character Usman (2026) describes, arising here downstream of compaction loss rather than from absent input. The substituted values are drawn from the high-frequency instance of their type: in one failing row, port `8443` for `62114`, `/var/lib/migration/state.json` for `/opt/kestrel/state/install.lock`, `02:00 UTC` for `01:26 UTC`, `16 shards` for `19 shards`, a canonical example UUID for the planted one, and an `example.com` origin for an internal host. Round numbers, stock ports and documentation placeholders form a greppable signature, which is the practical value of the observation.

We cannot say that row fabricated on *all fourteen* items: only the first 400 characters of each answer were retained, which reaches item nine. The counts above are likewise computed on that prefix, so they classify what the answer began by doing, not necessarily what it did throughout.

5 Discussion

The primary result concerns an assigned **carrier policy**, not JSON in general. Both central arms ask for structured output; the partitioned fields are rendered into headed text before the next fold. The study therefore does not compare raw JSON carriers with prose carriers, and it does not isolate semantic labels from field boundaries, local ceilings or rendering.

The positive panel mean makes partitioned carriers worth testing, but the median near zero, seven negative configuration effects and strong sparse-cell sensitivity argue against a universal design rule. For deployments resembling this protocol, the practical implication is to validate carrier policies on the intended model endpoint, serving stack, workload and recursion depth, using continuous retention outcomes and preserving failed trajectories in the analysis.

The exploratory evidence motivates—not resolves—mechanism questions. Realized fill and rendered headings are jointly produced with carrier content and are downstream of treatment. Later prompt length also changes because the previous carrier changes. A future factorial design should manipulate semantic labels, field boundaries, per-field ceilings, rendering and recursion depth separately. The carrier inspection further shows two measured stages of loss: identifiers can disappear during the fold, and identifiers that survive the fold can still be missed by the answer turn.

6 Limitations

Finite convenience panel. Twenty-three served endpoints were swept and twenty met the recorded six-month publication-date rule. The primary panel was selected for recency, endpoint availability,

context length and advertised schema support; it is not a probability sample. Its generalization unit is a served model/provider/quantization configuration, not an abstract model or an individual row. Related configurations share families and potentially serving infrastructure. Equal-family and leave-one-family-out results are sensitivities within this panel, not population inference. Supporting arms (`f`, `n`, `k`, `l`, `m`) ran on a smaller and partly older set.

Compound treatment. Matching the declared total character allowance does not isolate a unique partition mechanism. Semantic labels, local non-transferable field ceilings, rendered structure, realized carrier length and later prompt length change together. Results apply to the total assigned policy under the tested serving conditions.

The fill split is exploratory. The covariate, direction, cut point, failure subset and binary rescue definition were selected after seeing the same twenty configurations. Only two configurations form the low-fill side under the original rule, and one of them has a positive continuous paired effect. The split is a prospective hypothesis, not a validated model taxonomy or prediction rule.

Selection. Configurations were required to have a serving endpoint advertising structured outputs at 131,072 tokens or more, which the fold prompt needs. That requirement excludes most small models on the aggregator used. Publication dates and served aliases come from a dated provider snapshot; aliases are not immutable checkpoints, and the records do not prove that the recency cutoff preceded outcome inspection.

Sparse cells and missingness. Three configurations have fewer than four complete pairs, including one with only a single complete pair. Excluding those three reduces the equal-configuration mean from +0.135 to +0.070. Invalidity is arm-correlated, so complete-pair analysis alone may select outcomes. The planned-repeat bounds include one-arm-missing and both-arms-missing blocks and restrict the panel effect to [+0.048,+0.139] under outcomes bounded in [0,1].

Fact class. The planted facts are high-entropy, non-derivable identifiers, chosen as the hardest case. We do not establish that the effect holds for facts a model could paraphrase or reconstruct, and the claim should be read as scoped to verbatim retention of non-derivable tokens.

Scenario weight. One scenario carries the decisive comparisons. Its five planted segments land in three folds rather than five, so it exercises accumulation less than its design intends.

Validity and arm-correlated exclusion. Parse failures **stay in the means by design**: a schema the model cannot satisfy at its budget is arm behaviour rather than an infrastructure fault. Of the 23 single-field-schema rows carrying a parse failure, **18 are valid** and enter the aggregates, and they average **0.560** recall against **0.399** for the 175 clean rows of the same arms — so they are not dragging that mean down either. Over all arms the counts are 63 rows carrying a parse failure, 57 valid, averaging 0.476 against 0.553 for the 563 clean rows; the two populations differ because the arms do, which is why the schema-arm figures are the ones the claim needs.

A distributed-facts row is invalidated when the fold's catch-up branch fires, because its chunks were then folded together and it answers a different question. That happened on **6 rows, every one a schema arm** (`j` four times, `p` once, `q` once) and never on a plain arm. Their recalls were 1.000, 1.000, 0.000, 0.000, 0.000 and 0.000 — two removed from an arm that was working and four from arms that were failing — so the direction is not systematically flattering, but the correlation with arm is real and a reader should have it. Parse-failure rate is reported beside recall regardless.

Sampling and dispersion. Sampling is stochastic and no seed was sent, so runs are not bit-reproducible; this is a reproducibility limitation we disclose rather than one we corrected retroactively (Biderman et al. 2024).

Order control. The rotating square balances execution position but not first-order carryover, and one sweep's rotation does not close.

Token accounting. Prompt sizing used a character-ratio estimator for gateway sweeps and an exact tokenizer for the local reference, so absolute token figures are not comparable across those two. The estimator is also content-dependent in both directions: measured against a reference tokenizer it *under*counts identifier-dense text and *over*counts prose. We report the measured per-class ratios rather than a single multiplier, and note that this asymmetry is itself why a character-denominated ceiling is easier to fill with prose than with identifiers.

Grading. The scenario's fourteen graded checks bind a value to its label and require it to appear as a whole token. A weaker substring grader is used for the presence diagnostic on other scenarios; it was given the same boundary rule for the release, but figures collected before that change were not re-graded, because full answer text was not retained. No graded value is a substring of another, which bounds the effect.

Exploratory analyses. The fill covariate, direction, cut point, failure subset and binary rescue definition were selected after observing the same panel. Headings and content are co-produced, and the five-field policy structurally fixes heading count. Fill and heading associations are post-treatment, clustered descriptions rather than predictive validation or causal mediation (Yuan et al. 2025).

Reproducibility boundary. The public v2 fixture preserves the historical scenario's topology and prose but substitutes public-safe values and uses a frozen offline size heuristic where v1 queried a live tokenizer. Provider and tokenizer calibration and any v2 result replication remain future work. The historical v1 fixture, raw outputs and unrecovered dirty harness patch remain controlled, so exact v1 rerun is unsupported.

7 Data availability

The released package supports arithmetic reproduction of the published v1 results from sanitized aggregate, row-level and check-level evidence. It includes the evidence manifest, study configuration, analysis generator, generated tables and figures, and a provenance lock that pins controlled inputs by SHA-256 for authorized verification. The generator recomputes the paired continuous panel effect, sparse-cell and planned-repeat missingness sensitivities, family-weighted summaries, exploratory fill and heading analyses, within-row dependence, and the paired secondary contrasts.

The public `protocol/` subtree is a public-safe reconstruction and extraction of the load-bearing prompts, schemas, `j/q` parse-and-carry fail-safe, fold/chunk/catch-up/seed/artifact-ledger logic, graders and validity rule from named current commits. It also provides a distinct public-safe v2 candidate fixture. Fidelity to the unrecovered dirty historical harness cannot be proven. The published measurements used the private-shaped historical v1 fixture and controlled raw outputs, and served aliases are not immutable checkpoints. The package therefore supports audit and follow-up experiments, not an exact v1 rerun. Account state, credentials, private endpoints and local paths are excluded.

8 Version note

This revision corrects paired analyses for the plain-heading and reduced-ceiling comparisons, replaces threshold-led mechanism claims with a continuous finite-panel analysis and deterministic sensitivities, recomputes within-row dependence from the full released row set rather than earlier partial sets, and distinguishes arithmetic reproducibility of v1 from the public v2 follow-up protocol.

9 AI-assistance disclosure

Generative AI tools materially assisted literature discovery, analysis and test-code development, and manuscript editing. The author reviewed and verified the resulting citations, code, statistical outputs and prose and remains responsible for the study and its conclusions.

10 Conclusion

In this synthetic iterative-compaction protocol, assigning one undivided schema field or five named fields materially changed exact identifier retention. The anchor effect was large, but effects across the twenty-configuration primary panel were heterogeneous: the equal-configuration complete-pair mean was +0.135, the median was +0.027, and seven configuration effects were negative. The panel mean remained positive under sparse-cell, family-weighting and worst-case planned-repeat sensitivity analyses, while its magnitude depended strongly on sparse configurations.

Reducing the tested declared ceiling did not rescue the undivided anchor condition. On the anchor endpoint, the plain-heading arm outperformed the partitioned-schema arm (mean 0.986 versus 0.771;

8/10 versus 1/10 full-recall rows), so the `j-to-q` contrast cannot be attributed to visible headings alone. Exploratory fill, rendered-heading and carrier-inspection analyses suggest several pathways but do not identify a unique mechanism. The practical implication is to treat compaction schema as an empirical design choice and validate it on the intended model endpoint, serving stack, workload and recursion depth.

Adams, Griffin, Alexander Fabbri, Faisal Ladhak, Eric Lehman, and Noémie Elhadad. 2023. "From Sparse to Dense: GPT-4 Summarization with Chain of Density Prompting." *Proceedings of the 4th New Frontiers in Summarization Workshop*, 68–74. https://doi.org/10.18653/v1/2023.newsum-1.7.

An, Tao. 2026. *Fidelity Before Structure: Verbatim Chunks Beat Lossy Artifact Extraction in Long-Conversation LLM Memory*. https://arxiv.org/abs/2601.00821.

Biderman, Stella, Hailey Schoelkopf, Lintang Sutawika, et al. 2024. *Lessons from the Trenches on Reproducible Evaluation of Language Models*. https://arxiv.org/abs/2405.14782.

Chang, Yapei, Kyle Lo, Tanya Goyal, and Mohit Iyyer. 2024. "BooookScore: A Systematic Exploration of Book-Length Summarization in the Era of LLMs." *The Twelfth International Conference on Learning Representations*. https://arxiv.org/abs/2310.00785.

Fan, Hengxin. 2026. *Capacity, Not Format: Rethinking Structured Reasoning Failures*. https://arxiv.org/abs/2606.09410.

Horta Ribeiro, Manoel, Kristina Gligorić, and Robert West. 2019. "Message Distortion in Information Cascades." *Proceedings of the 2019 World Wide Web Conference*, 681–92. https://doi.org/10.1145/3308558.3313531.

Hwang, EunJeong, Yichao Zhou, James Bradley Wendt, et al. 2024. "Enhancing Incremental Summarization with Structured Representations." *Findings of the Association for Computational Linguistics: EMNLP 2024*, 3830–42. https://doi.org/10.18653/v1/2024.findings-emnlp.220.

Jayalath, Dulhan, James Bradley Wendt, Nicholas Monath, Sandeep Tata, and Beliz Gunel. 2025. "PRISM: Efficient Long-Range Reasoning with Short-Context LLMs." *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, 10196–218. https://doi.org/10.18653/v1/2025.emnlp-main.517.

Kutschka, Lorenz, and Bernhard Geiger. 2026. *Notation Matters: A Benchmark Study of Token-Optimized Formats in Agentic AI Systems*. https://arxiv.org/abs/2605.29676.

Kwon, Alex. 2026. *Reclaim Evaluation: A Lossy Memory Is Worse Than an Empty One*. https://arxiv.org/abs/2606.25449.

Lewis, Sydney. 2026. *Structured Distillation for Personalized Agent Memory: 11x Token Reduction with Retrieval Preservation*. https://arxiv.org/abs/2603.13017.

Maharana, Adyasha, Dong-Ho Lee, Sergey Tulyakov, Mohit Bansal, Francesco Barbieri, and Yuwei Fang. 2024. "Evaluating Very Long-Term Conversational Memory of LLM Agents." *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 13851–70. https://doi.org/10.18653/v1/2024.acl-long.747.

Miller, Evan. 2024. *Adding Error Bars to Evals: A Statistical Approach to Language Model Evaluations*. https://arxiv.org/abs/2411.00640.

Packer, Charles, Sarah Wooders, Kevin Lin, et al. 2023. *MemGPT: Towards LLMs as Operating Systems*. https://arxiv.org/abs/2310.08560.

Parikh, Tapan. 2026. *Structured Output Collapses Answer Diversity Across 44 Language Models*. https://arxiv.org/abs/2607.18476.

Petrov, Alex, Alexander Gusak, Denis Mukha, and Dima Korolev. 2026. *From Unstructured Recall*

to Schema-Grounded Memory: Reliable AI Memory via Iterative, Schema-Aware Extraction. https://arxiv.org/abs/2604.27906.

Sharma, Anantha, Sheeba Elizabeth John, Kaarthik Senthil Kumar, and Saratsuhas Vijayababu. 2026. *State Compression in Two-Agent LLM Relays: A Closed-World Study of Constraint Preservation*. https://arxiv.org/abs/2607.18265.

Tam, Zhi Rui, Cheng-Kuang Wu, Yi-Lin Tsai, Chieh-Yen Lin, Hung-yi Lee, and Yun-Nung Chen. 2024. "Let Me Speak Freely? A Study on the Impact of Format Restrictions on Performance of Large Language Models." *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Industry Track*, 1218–36. https://doi.org/10.18653/v1/2024.emnlp-industry.91.

Usman, Rana Muhammad. 2026. *PhantomFill: When the Form Demands an Answer, Language Models Invent One*. https://arxiv.org/abs/2607.20492.

Wang, Qingyue, Yanhe Fu, Yanan Cao, Shuai Wang, Zhiliang Tian, and Liang Ding. 2025. "Recursively Summarizing Enables Long-Term Dialogue Memory in Large Language Models." *Neurocomputing* 639: 130193. https://doi.org/10.1016/j.neucom.2025.130193.

Williams, E. J. 1949. "Experimental Designs Balanced for the Estimation of Residual Effects of Treatments." *Australian Journal of Scientific Research A* 2 (2): 149–68. https://doi.org/10.1071/CH9490149.

Wu, Di, Hongwei Wang, Wenhao Yu, Yuwei Zhang, Kai-Wei Chang, and Dong Yu. 2024. *LongMemEval: Benchmarking Chat Assistants on Long-Term Interactive Memory*. https://arxiv.org/abs/2410.10813.

Xu, Buqiang, Yijun Chen, Jizhan Fang, et al. 2026. "StructMem: Structured Memory for Long-Horizon Behavior in LLMs." *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, 122–46. https://doi.org/10.18653/v1/2026.acl-short.12.

Yuan, Han, Yue Zhao, Li Zhang, Wuqiong Luo, and Zheng Ma. 2025. *Quantifying the Impact of Structured Output Format on Large Language Models Through Causal Inference*. https://arxiv.org/abs/2509.21791.

Zahn, Oliver, and Simran Chana. 2026. *Facts as First Class Objects: Knowledge Objects for Persistent LLM Memory*. https://arxiv.org/abs/2603.17781.