ONEBASEKV: LOW-BIT CANONICAL KV SHARING WITH PER-REQUEST REPLACEMENT OVERLAYS FOR HIGH-FANOUT RAG

Anonymous authors
Paper under double-blind review

ABSTRACT

Long-context services increasingly process recurring content across many requests. High-fanout retrieval-augmented generation (RAG) is a representative workload in which the same document chunks are repeatedly combined with different queries, orders, and retrieval results. Autoregressive inference retains key–value (KV) states for these long inputs, so per-request cache ownership replicates document state and rapidly consumes memory even after low-bit quantization. Exact prefix sharing removes duplication only for byte-identical prefixes, while cache-fusion systems do not encode shared document state and query-conditioned repairs as jointly executable low-bit objects. We propose OneBaseKV, which stores each reusable chunk once as an immutable INT4 canonical base and represents query-sensitive differences with private INT4 replacement overlays. An architecture-normalized selector identifies repair positions, and materialization-equivalent composite attention merges the base, overlay, and dynamic KV through one global online softmax without a persistent dense request cache. We prove exact equivalence to standard attention over the corresponding materialized dequantized KV and derive a mixed-partition perturbation bound for heterogeneous precisions. Experiments show that OneBaseKV achieves the best quality among the compared efficient methods under fixed persistent-KV budgets. At a $0.52\times$ BF16 copy-on-write (BF16-COW) persistent-KV cap, it exceeds an oracle choosing the best feasible external method for each combination of model and task by 5.48%. At a 12K-token context and fanout 16, OneBaseKV uses only 37.7% of BF16-COW persistent KV. In two fixed-capacity serving traces, it achieves $6.06\times$–$8.57\times$ the service-level objective (SLO) goodput of Fusion-style. On non-retrieval RULER at 32K, it exceeds every efficient external method on three of four models.

1 INTRODUCTION

Long-context services support document analysis, code understanding, and retrieval-augmented generation (RAG), which connects large language models (LLMs) to external knowledge (Lewis et al., 2020; Bai et al., 2024). To avoid recomputing the history at every decoding step, autoregressive serving retains the key–value (KV) states of preceding tokens. As context length and concurrent batch size grow, these caches can consume memory comparable to model weights and substantially reduce batching headroom (Kwon et al., 2023; Liu et al., 2024; 2026b). The pressure becomes more acute when many requests reuse stable chunks but differ in their queries, document order, or composition, as commonly occurs in high-fanout RAG. Serving capacity therefore depends on both the bytes used to encode each KV state and the number of physical copies retained for recurring context.

Existing studies address individual aspects of this cost. KIVI, ChanMix, and PM-KVQ reduce per-request precision (Liu et al., 2024; Liao & Wen, 2026; Liu et al., 2026b). PyramidKV and LouisKV reduce within-request residency (Cai et al., 2025; Wu et al., 2026). CacheBlend and Cache-Craft reuse document KV and selectively recompute cross-chunk interactions, while FusionRAG separates shared chunk pages from query-exclusive repair pages and attends to them directly, but keeps both resident at full precision without low-bit composite execution (Yao et al., 2025; Agarwal et al., 2025; Wang et al., 2026a). ProphetKV, InfoFlow KV, and QCFuse improve repair selection with

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

query evidence (Wang et al., 2026b; Teng et al., 2026; Yan et al., 2026). These advances reduce copies, prefill work, or repair cost. However, they leave a systems gap: low-bit encoding, cross-request ownership, query-conditioned repair, and direct execution are not integrated into one cache representation.

A unified cache representation must reconcile shared ownership with query-specific repair and direct execution. Recurring document states should have a single request-independent physical owner, while corrected states remain request-private. Both classes must be compressed without rebuilding a complete low-bit cache for every request. During execution, repaired positions must replace their canonical counterparts, and the canonical base, private repairs, and generated KV must participate in one globally normalized attention distribution. Dense request caches would forfeit the sharing benefit, while independent stream normalization would change the resulting attention weights.

OneBaseKV defines a retriever-agnostic cache contract for high-fanout RAG and repeated or partitioned long-context inputs. Each reusable chunk has one immutable INT4 canonical base. An architecture-normalized selector identifies query-sensitive positions whose recomputed KV form a private INT4 replacement overlay. The overlay masks corresponding base entries while preserving logical coordinates. Composite attention combines the remaining base, overlay, and private dynamic KV through one global online softmax without materializing a persistent dense request cache. Public storage scales with unique chunks, while sessions own only private state. We prove equivalence to materialized execution over identical dequantized KV and derive a partition-aware quantization bound for mixed base and overlay precision.

To summarize, our contributions are as follows:

- We propose a cache representation that stores each reusable chunk once as an INT4 canonical base and encodes query-conditioned differences in private INT4 replacement overlays, making public memory scale with unique chunks rather than cumulative references.
- We combine architecture-normalized selection, disjoint replacement support, and one globally normalized online-softmax pass over the base, overlay, and dynamic KV. This design avoids a persistent dense request cache while remaining exactly equivalent to materialized attention over dequantized states, and it supports a mixed-partition perturbation bound.
- Experiments show that OneBaseKV retains 99.10% of Full BF16 quality. Under a $0.52\times$ BF16-COW persistent-KV cap, its mean quality exceeds an oracle selecting the best feasible external method per model–task cell by 5.48%. At 12K and fanout 16, it uses 37.7% of BF16-COW persistent KV. Two fixed-capacity traces yield $6.06\times$–$8.57\times$ the Fusion-style SLO goodput. On non-retrieval RULER at 32K, it outperforms all compared efficient external methods on three of four models.

2 Related Work

2.1 Per-Request KV Compression

Low-bit and low-rank methods reduce the representation owned by one request. KIVI applies asymmetric 2-bit key/value quantization (Liu et al., 2024). KVQuant combines pre-RoPE keys, non-uniform codes, and sparse outliers (Hooper et al., 2024). ChanMix and PM-KVQ allocate mixed precision by sensitivity or decoding stage (Liao & Wen, 2026; Liu et al., 2026b), while KVTC adds decorrelation and entropy coding (Staniszewski & Łańcucki, 2026). Palu and MatryoshkaKV compress feature dimensions with low-rank projections (Chang et al., 2025; Lin et al., 2025). These methods improve coding efficiency but retain per-request physical ownership.

2.2 Within-Request Residency

Eviction, merging, and tiering bound the cache retained by one session. H_2O preserves heavy hitters (Zhang et al., 2023). StreamingLLM preserves attention sinks and a recent window, whereas FastGen selects policies per attention module (Xiao et al., 2024; Ge et al., 2024). PyramidKV, RazorAttention, and HeadKV allocate capacity across layers or heads (Cai et al., 2025; Tang et al., 2025; Fu et al., 2025). KeepKV merges states losslessly (Tian et al., 2026). ThinKV and TRIM-KV combine importance with progressive eviction (Ramachandran et al., 2026; Bui et al., 2026). LouisKV and FreeKV use hierarchical retrieval and CPU–GPU placement (Wu et al., 2026; Liu et al., 2026a).

These techniques determine which states one request retains; they do not share recurring document states across requests.

2.3 Cross-Request Reuse and Repair

Reuse methods expand cache hits or restore interactions missing from independently encoded context. Prompt Cache reuses schema-defined modules (Gim et al., 2024). Block-Attention and APE relax sequential encoding through re-encoding or representation alignment (Ma et al., 2025; Yang et al., 2025b). RAGCache manages knowledge KV hierarchically, and SCBench evaluates shared-context systems (Jin et al., 2026; Li et al., 2025). ICaRus and Cartridges build reusable context representations (Woo et al., 2026; Eyuboglu et al., 2026). CacheBlend, Cache-Craft, and Ada-Cache reuse document states with selective recomputation or adaptive caching (Yao et al., 2025; Agarwal et al., 2025; Zeng et al., 2026). ProphetKV, InfoFlow KV, and QCFuse make repair query-aware (Wang et al., 2026b; Teng et al., 2026; Yan et al., 2026), while FusionRAG directly attends to shared and exclusive pages at native precision (Wang et al., 2026a).

These methods broaden reuse coverage and improve query-conditioned repair, but they do not jointly define the low-bit representation, physical ownership, and direct execution of shared and request-private states.

3 Method

3.1 Ownership-Separated KV Cache Representation

Let $D = (d_1, \ldots, d_{n_c})$ be an ordered list of reusable chunks, q a query, and $\mathsf{Seq}(D, q)$ the causal RAG prompt whose probe-query tokens follow all repairable document tokens. The number m of requests sharing this working set is the *fanout*. Let D_{tok} denote document positions in the prompt. OneBaseKV stores one immutable low-bit object B_u per chunk d_u and binds it to request-specific logical positions, producing a read-only ordered view $\overline{B}_q$.

A query probe selects a repair support $R_q \subseteq D_{\mathrm{tok}}$. Partial prefilling recomputes these positions and packs them at their original coordinates in a request-private replacement overlay P_q; bitmap I_q records the support. The effective base is $F_q = \overline{B}_q|_{D_{\mathrm{tok}} \setminus R_q}$, so base and overlay positions are disjoint. All valid non-document prompt states and generated history form a private BF16 dynamic stream G_t at decoding step t. OneBase means one base per chunk, not a corpus-wide object.

Figure 1 summarizes the workflow clockwise from (a) to (d). In (a), a trusted builder prefills each chunk once and registers an immutable INT4 base. Read-only views bind shared objects to request-specific positions, accommodating different chunk orders and subsets. In (b), an architecture-normalized selector uses relative-depth probes and KV-head aggregation to identify repair support R_q. In (c), causal partial prefill recomputes selected KV and packs it into the private INT4 replacement overlay P_q. Bitmap I_q masks the corresponding base entries without modifying shared storage. In (d), the effective base F_q, overlay P_q, and BF16 dynamic stream G_t contribute local online-softmax states that merge under one global normalization. Only G_t grows during decoding, with no persistent dense request-cache materialization. Algorithm A1 in Appendix A.4 details registration, initialization, decoding, and secure registry key construction.

3.2 Canonical Low-Bit Base and Query-Private Repair Overlays

Each nonempty chunk is prefilled once in a fixed within-chunk order, producing canonical states $\{K^{(0)}_{u,\lambda}, V^{(0)}_{u,\lambda}\}_{\lambda=1}^{N_{\mathrm{layer}}}$, where (0) denotes query-independent construction. Keys and values use group-wise affine quantization, following KIVI (Liu et al., 2024): keys are grouped along sequence and values along head dimension, with FP16 metadata. The default OneBaseKV configuration stores both the canonical base and replacement overlay in INT4; INT8 overlays are used only in precision ablations. Appendix C.1 gives the exact encoder and reconstruction error.

For rotary position embedding (RoPE), let the canonical key at local position π be $k^{\mathrm{rope}}_{\pi} = \Omega(\pi)\bar{k}$. Rebinding it to absolute position ϱ gives $k^{\mathrm{rope}}_{\varrho} = \Omega(\varrho - \pi)k^{\mathrm{rope}}_{\pi}$ under the composition law $\Omega(a)\Omega(b) = \Omega(a + b)$ (Su et al., 2024). This rotation is applied once during canonical construction

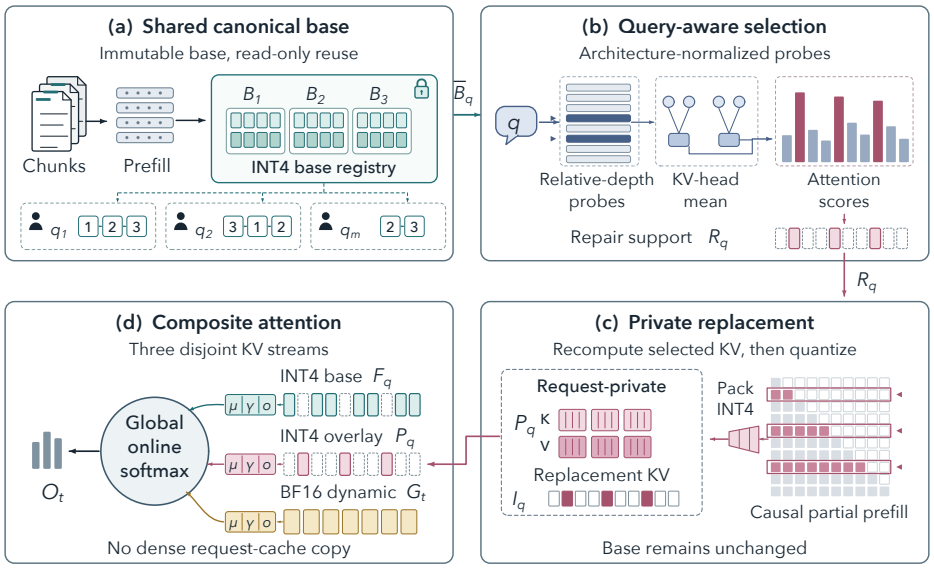


Figure 1: OneBaseKV's clockwise workflow with shared INT4 bases, private replacement overlays, and globally normalized attention. Numbered boxes identify shared-base references. Triplets (μ, γ, o) denote the local states in Equation 2, with stream subscripts omitted.

or chunk placement at cost $O(N_{\text{chunk}} \cdot d)$ per rebound chunk, not during per-step attention. Position rebinding therefore aligns canonical keys with their request-specific logical coordinates, while query-dependent and cross-chunk interactions are represented by the replacement overlay P_q.

To transfer selection across model depths and grouped-query-attention (GQA) layouts (Ainslie et al., 2023), OneBaseKV probes relative layers and aggregates query heads by their owning KV head. Let $\mathcal{T}_q$ be the nonempty probe-query token set and $\Xi \subset (0, 1]$ a fixed set of relative depths. Numbering Transformer blocks from zero, the implementation resolves $\ell(\xi) = \min\{N_{\text{layer}} - 1, \lfloor \xi N_{\text{layer}} \rfloor\}$ and sets $\mathcal{L}_{\text{probe}} = \{\ell(\xi) : \xi \in \Xi\}$. Let H_{KV} be the number of KV heads and $\{\mathcal{H}_h\}_{h=1}^{H_{\text{KV}}}$ the nonempty disjoint partition of all query heads induced by KV-head ownership. If $A_{\nu i}^{(\ell,a)}$ is attention from probe token ν to document position i at block index ℓ and query head a, the selector score is:

$$\omega_i = \frac{1}{|\mathcal{L}_{\text{probe}}| H_{\text{KV}} |\mathcal{T}_q|} \sum_{\ell,h,\nu} \frac{1}{|\mathcal{H}_h|} \sum_{a \in \mathcal{H}_h} A_{\nu i}^{(\ell,a)}, \tag{1}$$

where the outer sums cover $\ell \in \mathcal{L}_{\text{probe}}$, $h \in \{1, \ldots, H_{\text{KV}}\}$, and $\nu \in \mathcal{T}_q$. All evaluated models use uniform GQA with equally sized query-head groups, so Equation 1 reduces to an equal-weight query-head mean. Grouping serves only cross-architecture normalization, and both aggregation orders select bit-identical positions (Appendix B.1). For repair ratio $r \in (0, 1)$, R_q contains the $\text{round}(r|D_{\text{tok}}|)$ highest-scoring document positions, with ties broken deterministically by position. System, separator, query, and padding positions are excluded. Causal partial prefill (Yao et al., 2025; Wang et al., 2026b) recomputes selected KV layer by layer and writes only to P_q.

3.3 Materialization-Equivalent Composite Attention

Fix a request, step, layer, and attention head. The valid entries of $\mathcal{S}_{q,t} = \{F_q, P_q, G_t\}$ have pairwise-disjoint supports whose union is $\mathcal{J}_{q,t}$. For logit z_i and value v_i at position i, every nonempty stream c forms the following local online-softmax state:

$$\mu_c = \max_{i \in c} z_i, \qquad \gamma_c = \sum_{i \in c} e^{z_i - \mu_c}, \qquad o_c = \sum_{i \in c} e^{z_i - \mu_c} v_i. \tag{2}$$

The quantity γ_c is the local normalizer accumulator, and o_c the unnormalized value accumulator. Writing $\mathcal{S}_{q,t}^{+}$ for nonempty streams, blockwise online softmax (Milakov & Gimelshein, 2018; Dao

et al., 2022) merges them as follows:

$$\mu_\star = \max_{c \in \mathcal{S}^+_{q,t}} \mu_c, \quad O_t = \frac{\sum_{c \in \mathcal{S}^+_{q,t}} e^{\mu_c - \mu_\star} o_c}{\sum_{c \in \mathcal{S}^+_{q,t}} e^{\mu_c - \mu_\star} \gamma_c}. \tag{3}$$

Proposition 1 (Stream equivalence)**.** *If F_q, P_q, and G_t partition all valid positions and use the same dequantized KV, logical positions, positional encoding, head mapping, and causal mask as a materialized sequence, then Equation 3 is identical to standard softmax attention on that sequence under real arithmetic.*

Indeed, $e^{z_i - \mu_c} e^{\mu_c - \mu_\star} = e^{z_i - \mu_\star}$ reduces the composite numerator and denominator to the corresponding global sums. Appendix A.1 proves the claim and states the finite-precision boundary.

3.4 Persistent-Memory Scaling and Mixed-Precision Error Bound

For requests $q_1, \ldots, q_m$, let $\mathcal{U}_j$ be the referenced chunk set and $\mathcal{U} = \bigcup_j \mathcal{U}_j$ its unique union. Let $C_B(u)$ be bytes owned by base B_u, and let $C_P(j)$, $C_I(j)$, $C_H(j)$, and $C_G(j,t)$ be request j's bytes for its overlay, bitmap, metadata, and dynamic stream. Payload, scales, offsets, padding, and indices are charged to their physical owner exactly once. Persistent and resident KV are therefore:

$$\begin{aligned} M_{\text{persist}}(m) &= \sum_{u \in \mathcal{U}} C_B(u) + \sum_{j=1}^{m} \big(C_P(j) + C_I(j) + C_H(j)\big), \\ M_{\text{resident}}(m,t) &= M_{\text{persist}}(m) + \sum_{j=1}^{m} C_G(j,t). \end{aligned} \tag{4}$$

Thus, the public term follows unique chunks and is independent of cumulative references. Appendix C.2 specifies lifetime boundaries and sharing conditions.

Proposition 2 (Mixed-partition perturbation)**.** *Fix one attention call on common nonempty support $\mathcal{J}_{q,t}$. Let reference logits, values, probabilities, and output be z_i, v_i, p_i, and O. Let $\widetilde{z}_i = z_i + \delta_i$, $\widetilde{v}_i = v_i + \mathbf{e}_i$, and $\widetilde{p}, \widetilde{O}$ be their quantized counterparts. For each $c \in \mathcal{S}_{q,t}$, suppose $|\delta_i| \le \varepsilon_c$ and $\|\mathbf{e}_i\| \le \eta_c$ for $i \in c$, and define:*

$$\alpha_c = \sum_{i \in c} p_i, \quad Z^-_{\text{mix}} = \sum_c \alpha_c e^{-\varepsilon_c}, \quad \Psi_{\text{mix}} = \sum_c \alpha_c (e^{\varepsilon_c} - 1). \tag{5}$$

Let $\mathcal{S}^+_{q,t} = \{c \in \mathcal{S}_{q,t} : c \neq \emptyset\}$ and $\eta_{\max} = \max_{c \in \mathcal{S}^+_{q,t}} \eta_c$. Then, for any vector norm and $V_{\max} = \max_i \|v_i\|$, the following bounds hold:

$$\begin{aligned} \|\widetilde{p} - p\|_1 &\le \min\left\{2, \frac{2\Psi_{\text{mix}}}{Z^-_{\text{mix}}}\right\}, \\ \|\widetilde{O} - O\| &\le V_{\max} \min\left\{2, \frac{2\Psi_{\text{mix}}}{Z^-_{\text{mix}}}\right\} + \min\left\{\eta_{\max}, \frac{\sum_c \eta_c \alpha_c e^{\varepsilon_c}}{Z^-_{\text{mix}}}\right\}. \end{aligned} \tag{6}$$

The first cap on the value term follows because $\widetilde{p}$ remains a probability distribution, so the value perturbation is a convex combination of per-position error vectors and is therefore bounded by $\eta_{\max}$. The dynamic BF16 stream has $\varepsilon_{G_t} = \eta_{G_t} = 0$. Moving fixed positive reference mass from a partition with larger (ε, η) to one with both envelopes strictly smaller cannot increase either bound and strictly decreases the probability-mass-weighted terms. The proof and exact counterfactual conditions appear in Appendix A.2. Proposition 2 therefore characterizes fixed-support quantization for a single attention call under these conditions.

4 Experiments

4.1 Experimental Setup

Models and tasks. We evaluate Qwen3-8B, Qwen3-14B, Llama-3.1-8B-Instruct, and Mistral-7B-Instruct-v0.3 (Yang et al., 2025a; Grattafiori et al., 2024; Jiang et al., 2023), abbreviated as Q3-8B,

Q3-14B, L3.1-8B, and Mis-7B. Table 1 covers 13 LongBench tasks spanning single- and multi-document QA, few-shot learning, synthetic retrieval, and code completion (Bai et al., 2024). Core-RAG comprises 2WikiMQA, HotpotQA, MuSiQue, and Qasper. All-13 is the equally weighted task macro-average.

Baselines. We select representative baselines from the three directions reviewed in related work, including per-request compression, within-request residency, and cross-request reuse and repair, with both established methods and recent advances. Full BF16 is the uncompressed quality reference. BF16-COW applies OneBaseKV's selector and repair support to a BF16 copy-on-write base and overlay to isolate low-bit representation. KIVI represents per-request asymmetric quantization, while ChanMix represents channel-aware mixed-precision compression (Liu et al., 2024; Liao & Wen, 2026). PyramidKV evaluates retention-based capacity reduction without cross-request sharing (Cai et al., 2025). CacheBlend and Fusion-style represent RAG cache reuse and repair. CacheBlend selectively recomputes reused document KV, while Fusion-style follows the published native-precision shared and exclusive page design (Yao et al., 2025; Wang et al., 2026a). These baselines and OneBaseKV form the 416-cell main matrix. Per-request INT4, matched low-bit reuse controls, B4P8, and other precision pairs are evaluated only in ablations. Provenance and adapter boundaries are given in Appendix C.

Protocols and metrics. LongBench uses standard scorers scaled by 100. Relative probe depths are calibrated on disjoint model-specific development sets and frozen. Confirmatory runs use repair ratio $r = 0.30$ and group size 16 unless ablated. Probing uses eager attention and generation a common scaled dot-product attention backend. Physical KV measurements use 12K HotpotQA and Qasper contexts with three seeds, covering all models at fanout 16 and selected controls at fanout 8. Fanout 16 denotes 16 referencing sessions per document. Fixed-capacity traces run Qwen3-8B on one RTX 5090 (32 GB) GPU per run with a 6,144-MiB persistent-KV pool. Their working sets contain 16 documents/48 sessions or 32 documents/64 sessions, distinct from per-document fanout. Metrics include completion rate, SLO goodput in requests/s for time-to-first-token (TTFT) ≤ 600 ms, evictions, and full rebuilds. TTFT includes online probing and patch construction, while one-time base construction is accounted for separately. RULER uses the same cache path from 4K to 32K at fanout one. Appendices B and C provide ablation and lifecycle details.

Controlled isolation protocol. Controlled studies freeze all non-target variables: the representation study crosses $\mathrm{base}, \mathrm{patch} \in \{\mathrm{BF16}, \mathrm{INT8}, \mathrm{INT4}\}$ under identical masks on Q3-8B and L3.1-8B over 2WikiMQA, HotpotQA, and Qasper with 30 examples per cell; executable counterfactuals isolate selection, repair, low-bit storage, one-copy ownership, and direct composition; numerical checks compare disjoint global-softmax composition with matched materialization; matched low-bit controls compare selectors under a common store and executor; selector-factor tests fix repair budget and persistent bytes; same-document pairs fix the base while varying the query and support.

4.2 End-Task Quality Across Models and Tasks

Table 1 reports the complete four-model, 13-task comparison. OneBaseKV exceeds the strongest reproduced external baseline on All-13 for every model. Its four-model macro-average is 43.47, versus 42.69 when the best external result is selected per model. Its single-document, multi-document, and few-shot scores are 39.97, 46.06, and 34.86, compared with 39.41, 43.22, and 34.52 for the strongest single external method. The largest margin occurs on multi-document QA, where evidence is dispersed across documents.

Full BF16 is unconstrained by precision, repair budget, or shared ownership. OneBaseKV retains 99.10% of its macro-average quality and differs from BF16-COW by 0.13 points under the same selector and repair plan; with Proposition 1, this shows that low-bit conversion changes quality marginally while direct composition preserves materialized-attention semantics.

Appendix B.3 evaluates the same cache path on six RULER tasks from 4K to 32K. At 32K, OneBaseKV exceeds every efficient external method on three of four models while using 48.7% of BF16-COW single-request persistent KV.

Table 1: End-task quality on four models and 13 LongBench tasks. Full BF16 and BF16-COW are references; bold values mark the best point estimate among the remaining efficient methods, including ties. NrtvQA, MF-en, PCount, PRe, and LCC denote NarrativeQA, MultiFieldQA-en, PassageCount, PassageRetrieval-en, and Long Code Completion.

Model	Method	Single-doc. QA			Multi-doc. QA			Few-shot			Synthetic		Code		All-13
		NrtvQA	Qasper	MF-en	HotpotQA	2WikiMQA	MuSiQue	TREC	TriviaQA	SAMSum	PCount	PRe	LCC	RepoBench-P	
Qwen3-8B	Full BF16	27.10	45.81	51.95	63.03	42.82	34.71	38.33	20.57	26.36	0.00	95.00	60.31	51.41	42.88
	BF16-COW	28.77	40.85	52.53	64.13	43.06	34.69	36.67	19.97	28.03	1.00	95.00	56.26	52.36	42.56
	KIVI	**28.66**	36.24	47.75	53.19	34.74	28.95	**38.00**	19.42	26.71	1.00	93.00	59.76	50.96	39.88
	ChanMix	28.07	37.75	48.68	53.15	34.82	29.42	37.83	19.69	26.68	2.00	94.00	55.95	51.83	39.99
	PyramidKV	17.51	36.09	35.27	43.21	27.87	15.25	31.17	19.78	**29.97**	3.00	30.00	**60.96**	**56.83**	31.30
	CacheBlend	26.50	39.99	50.32	59.33	37.77	29.88	35.83	**20.63**	26.99	**5.00**	91.00	59.04	53.11	41.18
	Fusion-style (FP16)	26.96	37.51	51.07	56.11	41.81	31.19	37.33	20.02	27.24	4.00	91.00	55.70	50.44	40.80
	OneBaseKV	27.23	**40.82**	**51.70**	**63.33**	**43.71**	**33.11**	37.50	19.94	27.26	3.00	**95.00**	56.03	51.94	**42.35**
Qwen3-14B	Full BF16	27.66	43.35	51.13	61.55	59.35	36.61	42.17	18.90	25.68	7.00	97.00	58.34	51.29	44.62
	BF16-COW	28.46	38.87	52.00	62.09	56.15	39.22	41.50	18.51	25.06	4.00	97.00	55.04	52.00	43.84
	KIVI	**30.66**	39.31	52.26	56.39	51.68	35.90	**43.67**	**19.46**	**26.47**	4.45	94.00	55.84	50.96	43.16
	ChanMix	29.70	39.42	52.38	57.73	49.91	35.40	42.33	19.40	25.84	4.40	94.00	55.80	52.30	42.97
	PyramidKV	20.38	36.16	38.61	50.22	40.69	19.39	35.83	18.22	26.22	**12.00**	44.00	56.89	50.46	34.54
	CacheBlend	26.90	**40.05**	51.82	60.28	46.37	30.63	41.50	19.27	26.14	4.00	96.00	**57.21**	**52.97**	42.55
	Fusion-style (FP16)	26.70	37.99	52.92	55.39	43.13	**38.15**	40.50	18.59	25.88	4.00	94.00	56.24	50.43	41.84
	OneBaseKV	29.58	39.50	**53.33**	**62.93**	**54.89**	37.85	41.50	18.32	24.54	4.00	**97.00**	56.27	52.39	**44.01**
Llama-3.1-8B	Full BF16	28.25	44.78	57.09	58.36	47.69	33.12	35.00	18.10	23.92	7.92	97.00	55.19	50.64	42.85
	BF16-COW	28.80	43.89	54.14	54.17	47.38	32.88	37.83	19.12	26.66	6.00	94.00	55.76	47.68	42.18
	KIVI	26.39	43.11	**55.30**	55.02	41.82	24.22	33.50	18.39	23.93	**9.06**	97.00	48.28	47.54	40.27
	ChanMix	**28.63**	44.46	53.94	53.01	45.61	30.34	33.00	18.54	23.96	6.25	**98.00**	50.27	45.55	40.89
	PyramidKV	19.33	32.09	29.87	34.74	36.32	12.96	30.67	17.00	23.32	5.50	15.50	**57.49**	**52.33**	28.24
	CacheBlend	26.05	44.50	51.57	55.89	43.27	28.74	35.50	18.73	23.83	6.00	81.00	50.70	46.49	39.40
	Fusion-style (FP16)	26.67	**45.72**	52.54	53.49	44.70	**34.97**	38.00	**18.80**	25.34	5.00	86.00	56.63	45.82	41.05
	OneBaseKV	28.21	42.90	54.57	**56.60**	**49.33**	31.06	**38.33**	18.67	**26.07**	3.00	92.00	53.68	50.75	**41.94**
Mistral-7B-v0.3	Full BF16	28.90	32.98	48.56	51.93	35.70	26.37	41.33	74.78	47.84	1.11	85.00	55.86	56.07	45.11
	BF16-COW	28.81	35.74	47.17	51.28	39.78	30.19	41.50	75.20	48.46	2.00	87.00	54.90	53.50	45.81
	KIVI	**28.87**	33.36	**49.62**	50.77	40.57	25.52	40.25	68.28	48.58	3.82	82.00	56.51	56.24	44.95
	ChanMix	28.65	32.16	49.04	50.64	39.93	24.66	39.83	67.21	48.36	3.92	82.00	55.58	56.19	44.47
	PyramidKV	13.85	26.53	36.13	40.76	35.71	12.74	34.83	56.86	46.30	**4.09**	30.00	**60.30**	**56.64**	34.98
	CacheBlend	27.41	36.14	45.32	48.86	**40.65**	26.67	**42.00**	37.26	48.71	1.00	76.00	55.78	54.21	41.54
	Fusion-style (FP16)	25.98	34.80	47.73	**52.66**	38.09	28.91	**42.00**	71.15	**49.41**	3.00	85.00	56.23	54.73	45.36
	OneBaseKV	27.58	**36.67**	47.56	49.31	39.25	**31.31**	**42.00**	**75.14**	49.00	2.00	**86.00**	53.09	53.73	**45.59**

Table 2: Per-model quality and measured persistent KV at fanout 16. All-13 quality covers all 13 tasks; 12K KV is averaged over the physical HotpotQA and Qasper records. Lower KV (GiB) and KV/COW are better. Aggregates use unrounded records. Full BF16 is the unconstrained quality reference and is not part of the shared-layout physical measurement, so its KV cells are marked with a dash.

Method	Q3-8B		Q3-14B		L3.1-8B		Mis-7B		Overall		
	All-13	12K KV	All-13	12K KV	All-13	12K KV	All-13	12K KV	Quality	Rel. Full	KV/COW
Full BF16	42.88	–	44.62	–	42.85	–	45.11	–	43.86	100.00%	–
BF16-COW	42.56	15.742	43.84	17.491	42.18	13.993	45.81	13.993	43.60	99.39%	1.000
KIVI	39.88	6.662	43.16	7.402	40.27	5.879	44.95	5.922	42.06	95.90%	0.422
ChanMix	39.99	6.820	42.97	7.578	40.89	6.929	44.47	6.062	42.08	95.93%	0.449
PyramidKV	31.30	**4.741**	34.54	**5.274**	28.24	**4.239**	34.98	**4.235**	32.27	73.56%	**0.302**
CacheBlend	41.18	15.764	42.55	17.516	39.40	14.013	41.54	14.013	41.17	93.86%	1.001
Fusion-style (FP16)	40.80	15.742	41.84	17.491	41.05	13.993	45.36	13.993	42.26	96.35%	1.000
OneBaseKV	**42.35**	5.939	**44.01**	6.598	**41.94**	5.279	**45.59**	5.279	**43.47**	**99.10%**	0.377

4.3 Quality–Memory Efficiency

Table 2 compares All-13 quality with measured persistent KV. At 12K and fanout 16, OneBaseKV averages 5.774 GiB, 62.3% below BF16-COW, while retaining 99.71% of its quality. It exceeds KIVI and ChanMix by 1.41 and 1.39 points while using 10.7% and 16.0% less KV. PyramidKV uses less memory but scores 11.20 points lower.

OneBaseKV therefore advances the high-quality Pareto frontier. Figure 2(a) shows that it uses $0.377\times$ BF16-COW persistent KV at fanout 16, versus $0.637\times$ for per-request INT4. Figure 2(b) and Table 3(a) compare fixed-budget quality. Under the $0.40\times$ cap, OneBaseKV reaches 49.01 versus 37.47 for the feasible external method; at $0.52\times$, it is 2.55 points (5.48%) above the external per-cell oracle. The caps require $2.5\times$ and $1.92\times$ persistent-KV compression.

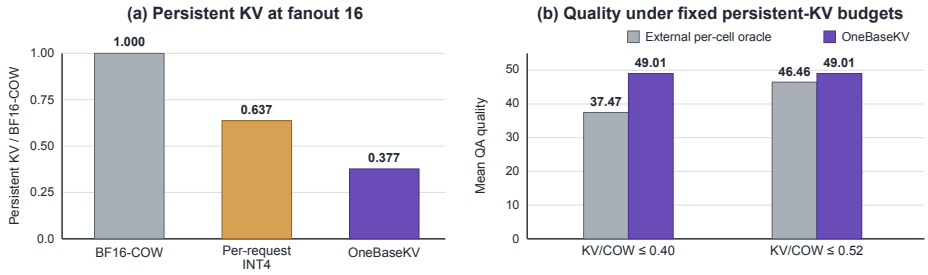


Figure 2: Persistent storage and fixed-budget quality. **(a)** Persistent KV normalized to BF16-COW at fanout 16. OneBaseKV uses 0.377 of the reference storage, while per-request INT4 is an ownership control. **(b)** Mean QA quality under two persistent-KV caps; the external oracle picks the best feasible external method per model–task cell. OneBaseKV retains the highest mean quality across the eight cells.

Table 3: **(a)** Quality under fixed persistent-KV budgets. Candidates must satisfy the same KV/COW cap on every model. Higher quality and lower KV/COW are better. Means use unrounded cell records.

Persistent-KV cap	Method	KV/COW	Q3-8B		Q3-14B		L3.1-8B		Mis-7B		Mean
			HotpotQA	Qasper	HotpotQA	Qasper	HotpotQA	Qasper	HotpotQA	Qasper	
Capacity ($\leq$0.40×)	PyramidKV	0.302	43.21	36.09	50.22	36.16	34.74	32.09	40.76	26.53	37.47
	OneBaseKV	0.377	**63.33**	**40.82**	**62.93**	**39.50**	**56.60**	**42.90**	**49.31**	**36.67**	**49.01**
High-quality ($\leq$0.52×)	KIVI	0.422	53.19	36.24	56.39	39.31	55.02	43.11	**50.77**	33.36	45.92
	ChanMix	0.449	53.15	37.75	57.73	39.42	53.01	**44.46**	50.64	32.16	46.04
	PyramidKV	0.302	43.21	36.09	50.22	36.16	34.74	32.09	40.76	26.53	37.47
	OneBaseKV	0.377	**63.33**	40.82	**62.93**	39.50	**56.60**	42.90	49.31	**36.67**	**49.01**

Table 3(b) reports Q/KV, the cell quality divided by persistent KV normalized to BF16-COW. OneBaseKV reaches 129.91 overall, 4.69% above the next-best 124.09, while retaining the highest raw mean of 49.01. It leads on three of four model aggregates. This result complements the fixed-budget comparison in Table 3(a) and the quality–memory tradeoff in Table 2.

Table 3: **(b)** Quality per unit of persistent KV over the same eight HotpotQA/Qasper cells as panel (a). Higher Q/KV, quality, and compression are better; lower persistent GiB is better.

Method	Q3-8B / Q/KV	L3.1-8B / Q/KV	Mis-7B / Q/KV	Q3-14B / Q/KV	Mean quality	Persistent KV (GiB)	Compression vs. COW	Overall Q/KV
BF16-COW	52.49	49.03	43.51	50.48	48.88	15.305	1.00	48.88
KIVI	105.66	116.78	99.40	113.07	45.92	6.466	2.37	108.73
ChanMix	104.91	98.41	95.55	112.11	46.04	6.848	2.23	102.75
PyramidKV	131.66	110.30	111.16	**143.24**	37.47	**4.622**	**3.31**	124.09
CacheBlend	49.59	50.12	42.44	50.09	48.13	15.326	1.00	48.06
Fusion-style (FP16)	46.81	49.60	43.73	46.69	46.71	15.305	1.00	46.71
OneBaseKV	**138.04**	**131.86**	**113.97**	135.76	**49.01**	5.774	2.65	**129.91**

With a fixed or sublinear unique-document working set, reuse rises with fanout: OneBaseKV saves 62.15%–62.29% persistent KV relative to BF16-COW from fanout 1 to 64. Fanouts 8 and 16 are measured directly, and the other points expand the same measured fixed and per-request terms (Appendix B).

Table 4 reports matched 12K serving traces under a 6,144-MiB persistent-KV pool. OneBaseKV achieves 6.06×–8.57× the SLO goodput of Fusion-style and reduces full rebuilds by 39.0%–57.3%. Against matched INT4 controls, it improves SLO goodput by 1.5×–2.0× over Fusion-style and 1.9×–2.4× over CacheBlend, confirming gains beyond quantization alone. Appendix C reports detailed runtime results.

Table 4: 12K traces in a 6,144-MiB persistent-KV pool. Entries with $\pm$ give three-seed means and standard deviations. SLO goodput counts completions/s with TTFT ≤ 600 ms. Paired goodput gives the mean [min, max] of within-seed ratios to Fusion-style. +INT4 uses per-request INT4 in the pool.

Method	Completion rate	SLO goodput	Paired goodput	Evictions	Full rebuilds
16 documents / 48 sessions					
Fusion-style	0.222 ± 0.012	0.0062 ± 0.0025	$1.00\times$	137.0 ± 2.9	72.7 ± 6.9
Fusion-style + INT4	0.206	0.0259	$4.47 \times [3.33, 5.22]$	117.0	78.0
CacheBlend-style + INT4	0.170	0.0214	$3.70 \times [2.76, 4.29]$	117.0	78.0
OneBaseKV	0.362 ± 0.029	0.0514 ± 0.0196	$8.57 \times [6.41, 10.80]$	114.3 ± 7.1	31.0 ± 10.2
32 documents / 64 sessions					
Fusion-style	0.212 ± 0.008	0.0055 ± 0.0002	$1.00\times$	184.7 ± 0.5	104.3 ± 6.6
Fusion-style + INT4	0.204	0.0220	$3.98 \times [3.36, 5.11]$	162.3	107.3
CacheBlend-style + INT4	0.169	0.0182	$3.29 \times [2.80, 4.21]$	162.3	107.3
OneBaseKV	0.320 ± 0.015	0.0337 ± 0.0102	$6.06 \times [4.51, 8.29]$	164.0 ± 5.4	63.7 ± 7.8

4.4 Ablation and Mechanism Analysis

Table 5 isolates five executable mechanisms. Removing query-aware selection or private repair lowers macro quality by 15.39 points (95% CI: [9.42, 21.48]) and 21.50 points (95% CI: [15.32, 27.75]), respectively. Replacing B16P16 with OneBaseKV changes macro quality by only 1.25 points; the paired interval for OneBaseKV minus B16P16 includes zero (95% CI: $[-3.05, 0.29]$), while persistent KV falls from 14.87 to 5.61 GiB. Replicating the base preserves quality but uses 12.28 GiB. Materialization preserves predictions but adds a 1.56-GiB temporary dense KV copy.

Table 5: Key executable-component ablations. Quality averages two models, three tasks, and 30 examples per cell; memory is measured at 12K and fanout 16. Higher quality and lower memory are better. The without-query-aware-selection row is prefix-first, retaining the leading $|R_q|$ document positions per sample.

Configuration	Macro quality	Persistent KV (GiB)	Temporary dense KV copy (GiB)
OneBaseKV	52.46	5.61	0.00
Without query-aware selection	37.07	5.61	0.00
Without private repair	30.96	2.81	0.00
Without low-bit representation (B16P16)	53.71	14.87	0.00
Without one-copy ownership	52.46	12.28	0.00
Without direct composite execution	52.46	5.61	1.56

Under frozen masks, OneBaseKV saves 62.27% persistent KV relative to B16P16; a matched low-bit CacheBlend selector scores 0.95 points lower at identical bytes, and CacheBlend-style + per-request INT4 uses 69.0% more persistent KV while scoring 0.66 points lower. All 30 natural Qasper query pairs produce distinct support hashes within each pair, with a mean repair-set Jaccard of 0.840 and mean evidence enrichment of 1.250. OneBaseKV exceeds equal-budget prefix-first selection by 10.15 points. Direct and materialized execution agree at all 180 generation endpoints. The maximum absolute attention-output difference across 540 layer records is 4.05×10^{-6}. Appendix Sections B.1 and B.2 provide the full controls and mechanism measurements, respectively.

5 Conclusion

In this paper, we introduce OneBaseKV, an ownership-separated low-bit KV cache for high-fanout RAG and recurring long-context workloads. Each reusable chunk is stored once as a canonical INT4 base, while request-specific differences are represented by private INT4 replacement overlays selected by an architecture-normalized selector. Direct composite attention combines the base, overlay, and dynamic KV with one global online softmax without materializing a dense request cache. OneBaseKV retains 99.10% of Full BF16 quality. At 12K and fanout 16, it uses 37.7% of BF16-COW persistent KV. At a $0.52\times$ BF16-COW persistent-KV cap, it exceeds an oracle choosing the best feasible external method for each combination of model and task by 5.48%. In two fixed-capacity serving traces, it delivers $6.06\times$–$8.57\times$ the Fusion-style SLO goodput. On non-retrieval RULER at 32K, it exceeds every efficient external method on three of four models.

AI USE STATEMENT

We use generative-AI tools only for language polishing and literature search. The authors independently verify the resulting text and cited sources and take full responsibility for the final content of the paper.

ETHICS STATEMENT

Cross-request KV reuse can create cache-poisoning or cross-tenant disclosure risks if request-derived state is admitted into a public cache. OneBaseKV therefore treats canonical bases as immutable objects created by a trusted offline process, binds cache identity to the model and serving configuration as well as tenant and knowledge-base authorization state, and keeps every replacement overlay request-private. Appendix E states the threat boundary and reports implementation-level invariant tests. These controls reduce risks introduced by shared state, but they do not replace application-level access control, provenance checks, or security review of the retrieval corpus.

REPRODUCIBILITY STATEMENT

The main paper defines the representation, selector, composite-attention operator, and physical-memory metric. Appendix A provides complete proofs and the executable cache path. Appendices B and C provide the reported matrices, ablations, timing protocol, lifecycle ledger, and validity criteria. The anonymized supplementary package contains code only: the implementation, frozen configurations, evaluation and aggregation programs, deployment-contract tests, and a README that maps each experiment to its reproduction command.

REFERENCES

Shubham Agarwal, Sai Sundaresan, Subrata Mitra, Debabrata Mahapatra, Archit Gupta, Rounak Sharma, Nirmal Joshua Kapu, Tong Yu, and Shiv Kumar Saini. Cache-craft: Managing chunk-caches for efficient retrieval-augmented generation. *Proceedings of the ACM on Management of Data*, 3(3):136:1–136:28, 2025. doi: 10.1145/3725273. URL https://doi.org/10.1145/3725273.

Joshua Ainslie, James Lee-Thorp, Michiel de Jong, Yury Zemlyanskiy, Federico Lebron, and Sumit Sanghai. GQA: Training generalized multi-query transformer models from multi-head checkpoints. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 4895–4901, 2023. doi: 10.18653/v1/2023.emnlp-main.298. URL https://aclanthology.org/2023.emnlp-main.298/.

Yushi Bai, Xin Lv, Jiajie Zhang, Hongchang Lyu, Jiankai Tang, Zhidian Huang, Zhengxiao Du, Xiao Liu, Aohan Zeng, Lei Hou, Yuxiao Dong, Jie Tang, and Juanzi Li. LongBench: A bilingual, multitask benchmark for long context understanding. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3119–3137, 2024. doi: 10.18653/v1/2024.acl-long.172. URL https://aclanthology.org/2024.acl-long.172/.

Ngoc Bui, Shubham Sharma, Simran Lamba, Saumitra Mishra, and Rex Ying. Cache what lasts: Token retention for memory-bounded KV cache in LLMs. In *International Conference on Learning Representations*, pp. 120316–120342, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/c3887741ee4e655494770be95303383f-Paper-Conference.pdf.

Zefan Cai, Yichi Zhang, Bofei Gao, Yuliang Liu, Yucheng Li, Tianyu Liu, Keming Lu, Wayne Xiong, Yue Dong, Junjie Hu, and Wen Xiao. PyramidKV: Dynamic KV cache compression based on pyramidal information funneling. In *Second Conference on Language Modeling*, 2025. URL https://openreview.net/forum?id=ayi7qezU87.

Chi-Chih Chang, Wei-Cheng Lin, Chien-Yu Lin, Chong-Yan Chen, Yu-Fang Hu, Pei-Shuo Wang, Ning-Chi Huang, Luis Ceze, Mohamed S. Abdelfattah, and Kai-Chiang Wu. Palu: KV-cache compression with low-rank projection. In *International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=LWMS4pk2vK.

Tri Dao, Daniel Y. Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. FlashAttention: Fast and memory-efficient exact attention with IO-awareness. In *Advances in Neural Information Processing Systems*, volume 35, pp. 16344–16359, 2022. doi: 10.52202/068431-1189. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/67d57c32e20fd0a7a302cb81d36e40d5-Abstract-Conference.html.

Sabri Eyuboglu, Ryan Ehrlich, Simran Arora, Neel Guha, Dylan Zinsley, Emily Liu, Atri Rudra, James Y. Zou, Azalia Mirhoseini, and Christopher Ré. Cartridges: Lightweight and general-purpose long context representations via self-study. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/4681359a7b1e94571598ad1adda35e6e-Paper-Conference.pdf.

Yu Fu, Zefan Cai, Abedelkadir Asi, Wayne Xiong, Yue Dong, and Wen Xiao. Not all heads matter: A head-level KV cache compression method with integrated retrieval and reasoning. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/f649556471416b35e60ae0de7c1e3619-Paper-Conference.pdf.

Suyu Ge, Yunan Zhang, Liyuan Liu, Minjia Zhang, Jiawei Han, and Jianfeng Gao. Model tells you what to discard: Adaptive KV cache compression for LLMs. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/639a9a172c044fbb64175b5fad42e9a5-Paper-Conference.pdf.

In Gim, Guojun Chen, Seung-seob Lee, Nikhil Sarda, Anurag Khandelwal, and Lin Zhong. Prompt cache: Modular attention reuse for low-latency inference. In *Proceedings of Machine Learning and Systems*, volume 6, 2024. URL https://proceedings.mlsys.org/paper_files/paper/2024/hash/a66caa1703fe34705a4368c3014c1966-Abstract-Conference.html.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, et al. The Llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024. URL https://arxiv.org/abs/2407.21783.

Coleman Hooper, Sehoon Kim, Hiva Mohammadzadeh, Michael W. Mahoney, Yakun Sophia Shao, Kurt Keutzer, and Amir Gholami. KVQuant: Towards 10 million context length LLM inference with KV cache quantization. In *Advances in Neural Information Processing Systems*, volume 37, 2024. doi: 10.52202/079017-0040. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/028fcbcf85435d39a40c4d61b42c99a4-Abstract-Conference.html.

Cheng-Ping Hsieh, Simeng Sun, Samuel Kriman, Shantanu Acharya, Dima Rekesh, Fei Jia, Yang Zhang, and Boris Ginsburg. RULER: What's the real context size of your long-context language models? In *First Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=kIoBbc76Sy.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mistral 7b. *arXiv preprint arXiv:2310.06825*, 2023. URL https://arxiv.org/abs/2310.06825.

Chao Jin, Zili Zhang, Xuanlin Jiang, Fangyue Liu, Shufan Liu, Xuanzhe Liu, and Xin Jin. RAGCache: Efficient knowledge caching for retrieval-augmented generation. *ACM Transactions on Computer Systems*, 44(1):2:1–2:27, 2026. doi: 10.1145/3768628. URL https://doi.org/10.1145/3768628.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with PagedAttention. In *Proceedings of the 29th ACM Symposium on Operating Systems Principles*, pp. 611–626, 2023. doi: 10.1145/3600006.3613165. URL https://doi.org/10.1145/3600006.3613165.

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, pp. 9459–9474, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/6b493230205f780e1bc26945df7481e5-Abstract.html.

Yucheng Li, Huiqiang Jiang, Qianhui Wu, Xufang Luo, Surin Ahn, Chengruidong Zhang, Amir Abdi, Dongsheng Li, Jianfeng Gao, Yuqing Yang, and Lili Qiu. SCBench: A KV cache-centric analysis of long-context methods. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/a540b17fb2295c736d5afd6c507acf66-Paper-Conference.pdf.

Chengxi Liao and Zeyi Wen. Channel-aware mixed-precision quantization for efficient long-context inference. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/55f5aa8d997a8071d0651e0009ec49a2-Paper-Conference.pdf.

Bokai Lin, Zihao Zeng, Zipeng Xiao, Siqi Kou, Tianqi Hou, Xiaofeng Gao, Hao Zhang, and Zhijie Deng. MatryoshkaKV: Adaptive KV compression via trainable orthogonal projection. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/d7b351608d824a4680344a02b180a947-Paper-Conference.pdf.

Guangda Liu, Chengwei Li, Zhenyu Ning, Jing Lin, Yiwu Yao, Danning Ke, Minyi Guo, and Jieru Zhao. FreeKV: Boosting KV cache retrieval for efficient LLM inference. In *International Conference on Learning Representations*, pp. 49150–49173, 2026a. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/5137e40ee586c4f1fc3615cf8cee4f87-Paper-Conference.pdf.

Tengxuan Liu, Shiyao Li, Jiayi Yang, Tianchen Zhao, Feng Zhou, Xiaohui Song, Guohao Dai, Shengen Yan, Huazhong Yang, and Yu Wang. PM-KVQ: Progressive mixed-precision KV cache quantization for long-CoT LLMs. In *International Conference on Learning Representations*, pp. 107881–107900, 2026b. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/afb8caec018d3c8f6ef8b81fa52386fe-Paper-Conference.pdf.

Zirui Liu, Jiayi Yuan, Hongye Jin, Shaochen Zhong, Zhaozhuo Xu, Vladimir Braverman, Beidi Chen, and Xia Hu. KIVI: A tuning-free asymmetric 2bit quantization for KV cache. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 32332–32344, 2024. URL https://proceedings.mlr.press/v235/liu24bz.html.

Dongyang Ma, Yan Wang, and Tian Lan. Block-attention for efficient prefilling. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/a03037317560b8c5f2fb4b6466d4c439-Paper-Conference.pdf.

Maxim Milakov and Natalia Gimelshein. Online normalizer calculation for softmax. *arXiv preprint arXiv:1805.02867*, 2018. URL https://arxiv.org/abs/1805.02867.

Akshat Ramachandran, Marina Neseem, Charbel Sakr, Rangharajan Venkatesan, Brucek Khailany, and Tushar Krishna. ThinKV: Thought-adaptive KV cache compression for efficient reasoning models. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/b2b502c3629beadda06311386d2c6f73-Paper-Conference.pdf.

Konrad Staniszewski and Adrian Łańcucki. KV cache transform coding for compact storage in LLM inference. In *International Conference on Learning Representations*, pp. 37768–37815, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/3fb6f10bd2784f6cfb6a6ed6280df40c-Paper-Conference.pdf.

Jianlin Su, Murtadha H. M. Ahmed, Yu Lu, Shengfeng Pan, Bo Wen, and Yunfeng Liu. RoFormer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024. doi: 10.1016/j.neucom.2023.127063. URL https://doi.org/10.1016/j.neucom.2023.127063.

Hanlin Tang, Yang Lin, Jing Lin, Qingsen Han, Danning Ke, Shikuan Hong, Yiwu Yao, and Gongyi Wang. RazorAttention: Efficient KV cache compression through retrieval heads. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/2a98af4fea6a24b73af7b588ca95f755-Paper-Conference.pdf.

Xin Teng, Canyu Zhang, Shaoyi Zheng, Danyang Zhuo, Tianyi Zhou, and Shenji Wan. InfoFlow KV: Information-flow-aware KV recomputation for long context. In *Proceedings of the 43rd International Conference on Machine Learning*, volume 306 of *Proceedings of Machine Learning Research*, 2026. URL https://icml.cc/virtual/2026/poster/61648.

Yuxuan Tian, Zihan Wang, Yebo Peng, Aomufei Yuan, Zhiming Wang, Bairen Yi, Xin Liu, Yong Cui, and Tong Yang. KeepKV: Achieving periodic lossless KV cache compression for efficient LLM inference. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 33259–33267, 2026. doi: 10.1609/aaai.v40i39.40611. URL https://ojs.aaai.org/index.php/AAAI/article/view/40611. Issue 39.

Jiahao Wang, Weiyu Xie, Mingxing Zhang, Boxing Zhang, Jianwei Dong, Yuening Zhu, Chen Lin, Jinqi Tang, Yaochen Han, Zhiyuan Ai, Xianglin Chen, Yongwei Wu, and Congfeng Jiang. From prefix cache to fusion RAG cache: Accelerating LLM inference in retrieval-augmented generation. *Proceedings of the ACM on Management of Data*, 4(1):41:1–41:28, 2026a. doi: 10.1145/3786655. URL https://doi.org/10.1145/3786655.

Shihao Wang, Jiahao Chen, Yanqi Pan, Hao Huang, Yichen Hao, Xiangyu Zou, Wen Xia, Wentao Zhang, Haitao Wang, Junhong Li, Chongyang Qiu, and Pengfei Wang. ProphetKV: User-query-driven selective recomputation for efficient KV cache reuse in retrieval-augmented generation. *arXiv preprint arXiv:2602.02579*, 2026b. URL https://arxiv.org/abs/2602.02579.

Sunghyeon Woo, Jaeeun Kil, Hoseung Kim, Minsub Kim, Joonghoon Kim, Ahreum Seo, Sungjae Lee, Minjung Jo, Jiwon Ryu, Baeseong Park, Se Jung Kwon, and Dongsoo Lee. ICaRus: Identical cache reuse for efficient multi-model inference. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/37f6be32f832caf0f7980469fb06165b-Paper-Conference.pdf.

Wenbo Wu, Qingyi Si, Xiurui Pan, Ye Wang, and Jie Zhang. LouisKV: Efficient KV cache retrieval for long input-output sequences. In *International Conference on Learning Representations*, pp. 65401–65421, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/6b241c515433caae3051266668d808b7-Paper-Conference.pdf.

Guangxuan Xiao, Yuandong Tian, Beidi Chen, Song Han, and Mike Lewis. Efficient streaming language models with attention sinks. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/5e5fd18f863cbe6d8ae392a93fd271c9-Paper-Conference.pdf.

Jianxin Yan, Wangze Ni, Zhenxin Li, Jiabao Jin, Zhitao Shen, Haoyang Li, Jia Zhu, Peng Cheng, Xuemin Lin, Lei Chen, and Kui Ren. QCFuse: Query-aware cache fusion via compressed view for efficient RAG serving. *arXiv preprint arXiv:2606.05875*, 2026. URL https://arxiv.org/abs/2606.05875.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025a. URL https://arxiv.org/abs/2505.09388.

Xinyu Yang, Tianqi Chen, and Beidi Chen. APE: Faster and longer context-augmented generation via adaptive parallel encoding. In *International Conference on Learning Representations*, 2025b. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/ce186a37e63b37638ecd06dee6b9a355-Paper-Conference.pdf.

Jiayi Yao, Hanchen Li, Yuhan Liu, Siddhant Ray, Yihua Cheng, Qizheng Zhang, Kuntai Du, Shan Lu, and Junchen Jiang. CacheBlend: Fast large language model serving for RAG with cached knowledge fusion. In *Proceedings of the Twentieth European Conference on Computer Systems*, pp. 94–109, 2025. doi: 10.1145/3689031.3696098. URL https://doi.org/10.1145/3689031.3696098.

Zihao Zeng, Siyi Li, Xinyu Yan, Lei Xiao, and Wei Yang Bryan Lim. AdaCache: Adaptive caching and context augmentation for efficient LLM serving. In *International Conference on Learning Representations*, pp. 7118–7130, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/0c0d84e06a860e20ccd2907bff2b6339-Paper-Conference.pdf.

Yichi Zhang, Zhiqi Wang, Huan Zhang, and Yuchen Yang. HijackKV: New threat in position-independent KV cache reuse. In *35th USENIX Security Symposium (USENIX Security 26)*, pp. 4701–4720. USENIX Association, 2026. URL https://www.usenix.org/conference/usenixsecurity26/presentation/zhang-yichi.

Zhenyu Zhang, Ying Sheng, Tianyi Zhou, Tianlong Chen, Lianmin Zheng, Ruisi Cai, Zhao Song, Yuandong Tian, Christopher Ré, Clark Barrett, Zhangyang Wang, and Beidi Chen. H2O: Heavy-hitter oracle for efficient generative inference of large language models. In *Advances in Neural Information Processing Systems*, volume 36, 2023. doi: 10.52202/075280-1506. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/6ceefa7b15572587b78ecfcebb2827f8-Abstract-Conference.html.

A Theoretical Proof and Algorithmic Implementation

A.1 Stream Equivalence of Composite Attention

Proof of Proposition 1. Fix a request q, decoding step t, network layer, and attention head. Identify each stream with its set of valid KV entries and write $\mathcal{S}_{q,t} = \{F_q, P_q, G_t\}$. Each stream may be empty; jointly, the streams are pairwise disjoint and have the nonempty union $\mathcal{J}_{q,t} = F_q \dot\cup P_q \dot\cup G_t$. The masked base stream F_q excludes R_q, the private overlay P_q carries replacement KV states on R_q, and G_t contains the request-private non-document prompt KV states visible to the current attention call together with the generated history and current input token. Let z_i be the finite logit at position i after positional encoding, scaling, and causal masking, and let v_i be its corresponding value. The materialized reference places exactly the same dequantized KV states used by the composite path at their logical positions. Its attention output is:

$$O_{\mathrm{mat}} = \frac{\sum_{i \in \mathcal{J}_{q,t}} e^{z_i} v_i}{\sum_{i \in \mathcal{J}_{q,t}} e^{z_i}}. \tag{A.1}$$

Define the set of nonempty streams as $\mathcal{S}^+_{q,t} = \{c \in \mathcal{S}_{q,t} : c \neq \varnothing\}$. For each $c \in \mathcal{S}^+_{q,t}$, the local online-softmax state is:

$$\mu_c = \max_{i \in c} z_i, \qquad \gamma_c = \sum_{i \in c} e^{z_i - \mu_c}, \qquad o_c = \sum_{i \in c} e^{z_i - \mu_c} v_i. \tag{A.2}$$

For an empty stream, γ_c and o_c may be defined as zero, but that stream is excluded from the maximum. Since $\mathcal{J}_{q,t}$ is nonempty, $\mathcal{S}^+_{q,t}$ is nonempty. The quantity γ_c is the local normalizer accumulator. Let $\mu_\star = \max_{c \in \mathcal{S}^+_{q,t}} \mu_c$. The result of the global merge is:

$$O_{\mathrm{cmp}} = \frac{\sum_{c \in \mathcal{S}^+_{q,t}} e^{\mu_c - \mu_\star} o_c}{\sum_{c \in \mathcal{S}^+_{q,t}} e^{\mu_c - \mu_\star} \gamma_c}. \tag{A.3}$$

The identity $e^{\mu_c-\mu_\star}e^{z_i-\mu_c} = e^{z_i-\mu_\star}$ reduces the numerator and denominator of equation A.3, respectively, to:

$$\sum_{i\in\mathcal{J}_{q,t}} e^{z_i-\mu_\star}v_i, \qquad \sum_{i\in\mathcal{J}_{q,t}} e^{z_i-\mu_\star}. \tag{A.4}$$

Multiplying both expressions in equation A.4 by $e^{\mu_\star}$ recovers equation A.1; hence $O_{\mathrm{cmp}} = O_{\mathrm{mat}}$ under real arithmetic.

This equivalence requires both paths to use identical dequantized KV states, logical positions, rotary position embedding (RoPE) phases, grouped-query attention (GQA) head mappings, and causal masks. In particular, R_q must be removed from the valid base positions. Otherwise, F_q and P_q are not disjoint, and duplicated tokens change both the global numerator and denominator. Finite-precision implementations may still differ because of reduction order, exponential approximation, rounding, and kernel fusion. Proposition 1 therefore isolates only streamwise execution from materialized execution; the fixed-support quantization perturbation is controlled in Appendix A.2. □

A.2 Mixed-Partition Quantization Perturbation Bound

Proof of Proposition 2. Retain the request, step, network layer, attention head, and query vector fixed in Appendix A.1, and suppose that the reference and quantized computations share the same nonempty finite support $\mathcal{J}_{q,t}$. All logits z_i are assumed to be finite real scalars, and all values v_i finite real vectors. Define the reference probabilities and output as $p_i = e^{z_i}/\sum_{i'\in\mathcal{J}_{q,t}} e^{z_{i'}}$ and $O = \sum_{i\in\mathcal{J}_{q,t}} p_i v_i$. After quantization, the logits and values become:

$$\widetilde{z}_i = z_i + \delta_i, \qquad \widetilde{v}_i = v_i + \mathbf{e}_i, \tag{A.5}$$

The corresponding probabilities and output are:

$$\widetilde{p}_i = \frac{e^{\widetilde{z}_i}}{\sum_{i'\in\mathcal{J}_{q,t}} e^{\widetilde{z}_{i'}}}, \qquad \widetilde{O} = \sum_{i\in\mathcal{J}_{q,t}} \widetilde{p}_i\widetilde{v}_i. \tag{A.6}$$

Let $\mathcal{S}_{q,t} = \{F_q, P_q, G_t\}$ again denote the three streams, each of which may be empty, with pairwise-disjoint supports whose union is $\mathcal{J}_{q,t}$. For every $i \in c$, assume finite nonnegative bounds ε_c and η_c such that:

$$|\delta_i| \le \varepsilon_c, \qquad \|\mathbf{e}_i\| \le \eta_c, \qquad c \in \mathcal{S}_{q,t}. \tag{A.7}$$

The dynamic stream is not requantized in this comparison, so $\varepsilon_{G_t} = \eta_{G_t} = 0$. Let $\alpha_c = \sum_{i\in c} p_i$. Empty streams have $\alpha_c = 0$, while $\alpha_c \ge 0$ and $\sum_{c\in\mathcal{S}_{q,t}} \alpha_c = 1$. The derivation below holds for any vector norm.

Set $w_i = e^{\delta_i}$ and $Z_\delta = \sum_{i\in\mathcal{J}_{q,t}} p_i w_i$. The perturbed probabilities can be written as:

$$\widetilde{p}_i = \frac{p_i w_i}{Z_\delta}. \tag{A.8}$$

Equation A.7 gives the positive lower bound:

$$Z_\delta \ge Z^-_{\mathrm{mix}} = \sum_{c\in\mathcal{S}_{q,t}} \alpha_c e^{-\varepsilon_c} > 0. \tag{A.9}$$

Define the mixed logit coefficient:

$$\Psi_{\mathrm{mix}} = \sum_{c\in\mathcal{S}_{q,t}} \alpha_c\big(e^{\varepsilon_c} - 1\big). \tag{A.10}$$

Because $|\delta_i| \le \varepsilon_c$ implies $|e^{\delta_i} - 1| \le e^{\varepsilon_c} - 1$, and because $\sum_{i\in\mathcal{J}_{q,t}} p_i = 1$, we obtain:

$$\sum_{i\in\mathcal{J}_{q,t}} p_i|w_i - 1| \le \Psi_{\mathrm{mix}}, \qquad |Z_\delta - 1| \le \Psi_{\mathrm{mix}}. \tag{A.11}$$

Consequently:

$$\begin{aligned}\|\widetilde{p} - p\|_1 &= \frac{1}{Z_\delta} \sum_{i \in \mathcal{J}_{q,t}} p_i |w_i - Z_\delta| \\ &\leq \frac{1}{Z_\delta} \left(\sum_{i \in \mathcal{J}_{q,t}} p_i |w_i - 1| + |Z_\delta - 1| \right) \\ &\leq \frac{2\Psi_{\text{mix}}}{Z^-_{\text{mix}}}. \end{aligned} \tag{A.12}$$

The L_1 distance between any two probability distributions is at most two, so:

$$\|\widetilde{p} - p\|_1 \leq \min\left\{2, \frac{2\Psi_{\text{mix}}}{Z^-_{\text{mix}}}\right\}. \tag{A.13}$$

Let $V_{\max} = \max_{i \in \mathcal{J}_{q,t}} \|v_i\|$. Because $\sum_i (\widetilde{p}_i - p_i) = 0$, the probability-driven term is invariant to shifting all values by a common vector, so $V_{\max}$ can be tightened to $R_v = \min_{\mathbf{c}} \max_i \|v_i - \mathbf{c}\|$; we keep $V_{\max}$ for simplicity. Decompose the output error into probability and value perturbations:

$$\widetilde{O} - O = \sum_{i \in \mathcal{J}_{q,t}} (\widetilde{p}_i - p_i) v_i + \sum_{i \in \mathcal{J}_{q,t}} \widetilde{p}_i \mathbf{e}_i. \tag{A.14}$$

By the triangle inequality and absolute homogeneity of the vector norm, the first term is at most $V_{\max} \|\widetilde{p} - p\|_1$. For the second term, equations A.7–A.9 imply:

$$\begin{aligned}\left\| \sum_{i \in \mathcal{J}_{q,t}} \widetilde{p}_i \mathbf{e}_i \right\| &\leq \sum_{i \in \mathcal{J}_{q,t}} \widetilde{p}_i \|\mathbf{e}_i\| \\ &= \frac{1}{Z_\delta} \sum_{c \in \mathcal{S}_{q,t}} \sum_{i \in c} p_i e^{\delta_i} \|\mathbf{e}_i\| \\ &\leq \frac{\sum_{c \in \mathcal{S}_{q,t}} \eta_c \alpha_c e^{\varepsilon_c}}{Z^-_{\text{mix}}}. \end{aligned} \tag{A.15}$$

Because $\widetilde{p}$ remains a probability distribution, the same term is a convex combination of the per-position error vectors, giving the cap:

$$\left\| \sum_{i \in \mathcal{J}_{q,t}} \widetilde{p}_i \mathbf{e}_i \right\| \leq \sum_{i \in \mathcal{J}_{q,t}} \widetilde{p}_i \|\mathbf{e}_i\| \leq \max_{i \in \mathcal{J}_{q,t}} \|\mathbf{e}_i\| \leq \max_{c \in \mathcal{S}^+_{q,t}} \eta_c = \eta_{\max}, \tag{A.16}$$

where $\mathcal{S}^+_{q,t} = \{c \in \mathcal{S}_{q,t} : c \neq \emptyset\}$ and $\eta_{\max} = \max_{c \in \mathcal{S}^+_{q,t}} \eta_c$.

Combining equations A.13–A.16 yields:

$$\|\widetilde{O} - O\| \leq V_{\max} \min\left\{2, \frac{2\Psi_{\text{mix}}}{Z^-_{\text{mix}}}\right\} + \min\left\{\eta_{\max}, \frac{\sum_{c \in \mathcal{S}_{q,t}} \eta_c \alpha_c e^{\varepsilon_c}}{Z^-_{\text{mix}}}\right\}. \tag{A.17}$$

Equation A.17 is a data-dependent sufficient bound. Since $\varepsilon_{G_t} = \eta_{G_t} = 0$, the dynamic stream contributes neither to Ψ_{mix} nor to the numerator of the value-perturbation term, while contributing α_{G_t} to the lower bound in equation A.9.

It remains to establish the counterfactual comparison of the bound coefficients. Hold p_i and the error envelopes of every stream fixed. Let a nonempty subset $F_{q,P} \subseteq F_q$ have reference probability mass $\beta = \sum_{i \in F_{q,P}} p_i > 0$, and reassign this subset to P_q. Then $\alpha'_{F_q} = \alpha_{F_q} - \beta$, $\alpha'_{P_q} = \alpha_{P_q} + \beta$, and all other α_c remain unchanged. If $\varepsilon_{P_q} < \varepsilon_{F_q}$ and $\eta_{P_q} < \eta_{F_q}$, the change in equation A.10

is $\beta(e^{\varepsilon_{P_q}} - e^{\varepsilon_{F_q}}) < 0$; the change in equation A.9 is $\beta(e^{-\varepsilon_{P_q}} - e^{-\varepsilon_{F_q}}) > 0$; and the change in the numerator of equation A.15 is $\beta(\eta_{P_q} e^{\varepsilon_{P_q}} - \eta_{F_q} e^{\varepsilon_{F_q}}) < 0$. Therefore, the truncated probability bound in equation A.13 does not increase; the max-based cap in equation A.16 does not increase; and the probability-mass-weighted value term decreases strictly. Consequently, the minimum in equation A.17 does not increase and decreases strictly whenever the probability-mass-weighted term is the active branch of the minimum.

This algebraic comparison keeps the reference probability at every position fixed and characterizes how the sufficient bound changes when the same probability mass is associated with a partition having tighter error envelopes. Proposition 2 fixes the query vector, dynamic KV states, logical support, positional encoding, head mapping, and mask. The condition $\varepsilon_{G_t} = \eta_{G_t} = 0$ indicates that the dynamic stream is not requantized in this comparison. The resulting bound therefore characterizes fixed-call quantization error under these conditions. Extending the analysis across layers or autoregressive steps would require additional assumptions on the stability of later queries, dynamic KV states, and repair choices. □

A.3 Error-Source Decomposition for a Fixed Attention Call

Fix the same request, decoding step, network layer, attention head, query vector, repair support R_q, logical support, positional encoding, head mapping, causal mask, dynamic KV states, and all KV values not under comparison. Every output below is produced by an ideal real-arithmetic attention operator in the same normed vector space. Let O_{full} denote the output of query-conditioned BF16 full prefilling. Let $O^{\text{mat}}_{16,16}$ denote the output after materializing a BF16 canonical base and BF16 overlay, and let $O^{\text{mat}}_{4,16}$ denote the corresponding output with only the base quantized to INT4. For $b_P \in \{4, 8\}$, let O^{mat}_{4,b_P} additionally quantize the overlay to b_P bits. Finally, let O^{cmp}_{4,b_P} execute streamwise composite attention on the same dequantized KV states as O^{mat}_{4,b_P}. Proposition 1 and the triangle inequality give:

$$\begin{aligned}\|O^{\text{cmp}}_{4,b_P} - O_{\text{full}}\| &= \|O^{\text{mat}}_{4,b_P} - O_{\text{full}}\| \\ &\le \|O^{\text{mat}}_{4,b_P} - O^{\text{mat}}_{4,16}\| \\ &\quad + \|O^{\text{mat}}_{4,16} - O^{\text{mat}}_{16,16}\| \\ &\quad + \|O^{\text{mat}}_{16,16} - O_{\text{full}}\|. \end{aligned} \tag{A.18}$$

The three terms on the right-hand side of equation A.18 correspond, in order, to overlay quantization, base quantization, and canonical-context approximation together with selector omissions. When Proposition 2 is applied to the first term, the error envelopes on F_q and G_t are set to zero. When it is applied to the second term, the corresponding envelopes on P_q and G_t are set to zero. The third term changes the KV representation itself and therefore does not follow from the fixed-support quantization bound; it must be examined separately through mechanism analysis and end-task experiments.

Define the finite-precision composite–materialized discrepancy as:

$$E_{\text{fp}} = \|\text{fl}_{\text{cmp}}(\mathcal{X}_{4,b_P}) - \text{fl}_{\text{mat}}(\mathcal{X}_{4,b_P})\|\,, \tag{A.19}$$

where $\mathcal{X}_{4,b_P}$ is the input tuple formed by the common query vector, dequantized KV states, logical positions, positional encoding, head mapping, and mask used by the two kernels, and fl_{cmp} and fl_{mat} denote their finite-precision implementations. Proposition 1 shows only that the corresponding ideal real-arithmetic discrepancy is zero; it does not provide an analytic upper bound on E_{fp}. Equation A.18 separates operator-level error sources for one fixed attention call. It neither assumes that the three terms are independent nor claims that they accumulate linearly across network layers.

A.4 Algorithm Execution Procedure and Secure Cache Key

In Algorithm A1, Hash is a collision-resistant digest function, and θ is the execution-configuration fingerprint. The identity digest of chunk d_u under its tenant and content version is ι_u, while κ_u is the registry key. The access control list (ACL) version is denoted by $\mathsf{ACL\ epoch}$. The request-private session state is $\mathcal{C}_q$, and τ_t is the input token at step t. The state Y_t is the request-private history KV persisted before that step, while G_t is the visible dynamic stream after adding the KV states of τ_t.

Algorithm A1 OneBaseKV document registration and direct composite decoding

Require: Ordered admitted chunks $D = (d_1, \ldots, d_{n_c})$; query q; repair budget r; relative probe depths Ξ; GQA groups $\{\mathcal{H}_h\}_{h=1}^{H_{\mathrm{KV}}}$
Require: Base precision $b_B = 4$; patch precision $b_P \in \{4, 8\}$; quantization-group size n_{grp}
Ensure: Generated token sequence and request-private session $\mathcal{C}_q$

Phase I: Trusted canonical registration
$\theta \leftarrow \mathrm{Hash}(\mathsf{model\ weights/config}, \mathsf{tokenizer}, \mathsf{template}, \mathsf{RoPE}, \mathsf{adapter}, \mathsf{quantizer}, \mathsf{format})$
for $u = 1, \ldots, n_c$ **do**
  $\iota_u \leftarrow \mathrm{Hash}(\mathsf{namespace}, \mathsf{ACL\ epoch}, \mathrm{Hash}(d_u), \mathsf{document\ version})$
  $\kappa_u \leftarrow \mathrm{Hash}(\iota_u, \theta)$
  if $\neg$REGISTRY.CONTAINS(κ_u) **then**
    $(K_u^{(0)}, V_u^{(0)}, \mathbf{pos}_u^{(0)}) \leftarrow$ CANONICALPREFILL(d_u)
    $B_u \leftarrow$ PACK$(K_u^{(0)}, V_u^{(0)}, \mathbf{pos}_u^{(0)}; b_B, n_{\mathrm{grp}})$
    REGISTRY.INSERTIMMUTABLE$(\kappa_u, B_u, \mathrm{Hash}(B_u))$
  end if
end for
Phase II: Per-request initialization
$(B_u)_{u=1}^{n_c} \leftarrow$ REGISTRY.ACQUIREORDERED$((\kappa_u)_{u=1}^{n_c})$
VERIFYINTEGRITYANDACCESS$(\mathsf{namespace}, \mathsf{ACL\ epoch}, \{\mathrm{Hash}(B_u)\}_{u=1}^{n_c})$
$\overline{B}_q \leftarrow$ BINDLOGICALPOSITIONS$((B_u)_{u=1}^{n_c}, \mathsf{Seq}(D, q))$
$R_q \leftarrow$ ARCHNORMSELECT$(q, \overline{B}_q; \Xi, \{\mathcal{H}_h\}_{h=1}^{H_{\mathrm{KV}}}, r) \cap D_{\mathrm{tok}}$
$(K_q^P, V_q^P, Y_0, \tau_0) \leftarrow$ PARTIALPREFILL$(\mathsf{Seq}(D, q), \overline{B}_q, R_q)$
$I_q \leftarrow$ SUPPORTBITMAP(R_q, D_{tok})
$P_q \leftarrow$ PACK$(K_q^P, V_q^P; b_P, n_{\mathrm{grp}})$
$F_q \leftarrow$ MASKBASE$(\overline{B}_q, I_q)$ ▷ Masked view; no persistent base-KV copy
$\mathcal{C}_q \leftarrow$ SESSION$(\overline{B}_q, F_q, P_q, I_q, Y_0)$
EMIT(τ_0)
Phase III: Per-token direct composite decoding
for $t = 0, 1, \ldots$ while τ_t is nonterminal and budget remains **do**
  for $\lambda = 1, \ldots, N_{\mathrm{layer}}$ **do**
    $(Q_t^\lambda, K_t^\lambda, V_t^\lambda) \leftarrow$ CURRENTQKV$(\tau_t, t, \lambda, \mathcal{C}_q)$
    $G_t^\lambda \leftarrow Y_t^\lambda \dot\cup \{(K_t^\lambda, V_t^\lambda)\}$
    $\sigma_{F,t}^\lambda \leftarrow$ STREAMSTATE$(Q_t^\lambda, F_q^\lambda)$
    $\sigma_{P,t}^\lambda \leftarrow$ STREAMSTATE$(Q_t^\lambda, P_q^\lambda, I_q)$
    $\sigma_{G,t}^\lambda \leftarrow$ STREAMSTATE$(Q_t^\lambda, G_t^\lambda)$
    $O_t^\lambda \leftarrow$ GLOBALONLINESOFTMAXMERGE$(\sigma_{F,t}^\lambda, \sigma_{P,t}^\lambda, \sigma_{G,t}^\lambda)$
    ADVANCELAYERSTATE$(\mathcal{C}_q, \lambda, O_t^\lambda)$
    $Y_{t+1}^\lambda \leftarrow G_t^\lambda$
    UPDATEDYNAMIC$(\mathcal{C}_q, \lambda, Y_{t+1}^\lambda)$
  end for
  $\tau_{t+1} \leftarrow$ SAMPLE$(\mathcal{C}_q)$; EMIT(τ_{t+1})
end for
return generated token sequence and $\mathcal{C}_q$

The variables $\sigma_{F,t}^\lambda$, $\sigma_{P,t}^\lambda$, and $\sigma_{G,t}^\lambda$ are the local online-softmax states of the three streams at layer λ, and O_t^λ is the attention output at that layer. The superscript λ on F_q, P_q, Y_t, or G_t denotes the corresponding layerwise KV slice. Every local state and $\mathrm{GlobalOnlineSoftmaxMerge}$ is computed independently for each query head. For concision, the algorithm suppresses the head index. It never shares a maximum or normalization denominator across query heads.

The collections $K_u^{(0)} = \{K_{u,\lambda}^{(0)}\}_{\lambda=1}^{N_{\mathrm{layer}}}$ and $V_u^{(0)} = \{V_{u,\lambda}^{(0)}\}_{\lambda=1}^{N_{\mathrm{layer}}}$ contain the canonical keys and values of d_u over all layers, and $\mathbf{pos}_u^{(0)}$ contains its within-chunk positions. The variables K_q^P and V_q^P are the all-layer replacement KV states on R_q, whereas Q_t^λ, K_t^λ, and V_t^λ are the query, key, and value of τ_t at layer λ. During partial prefill, only selected token states are propagated across layers. At each layer, a selected position attends causally to canonical KV for unrepaired predecessors and to recomputed KV for repaired predecessors. Thus, unselected hidden states need not be regenerated and no dense request-level cache is materialized. Within the overlay buffer,

replacement states are packed contiguously in selection order. Keys are grouped along the packed selected-token axis and values along the channel axis. When the final key group is short, it is padded to the configured group size by replicating the last entry. These padding entries do not belong to the logical replacement support, and their storage bytes are included in physical-memory accounting. During request initialization, $\mathrm{PartialPrefill}$ writes to Y_0 every valid non-document prompt KV state that belongs to neither the base nor the overlay, including the system prefix, template separators, and query but excluding padding. It also samples τ_0 from the final prompt state. Consequently, Y_0 does not contain the KV states of τ_0, and Phase III begins by processing τ_0 to predict τ_1. If τ_0 is already a terminal token, Phase III is skipped.

The algorithm details only the cache-related path. Standard residual connections, normalization, and multilayer perceptron operations remain unchanged from the underlying model. $\mathrm{CurrentQKV}$ reads the current layer input, while $\mathrm{AdvanceLayerState}$ performs the residual, normalization, multilayer perceptron, and inter-layer state transitions. All other objects agree with B_u, $\overline{B}_q$, R_q, P_q, I_q, F_q, G_t, and O_t in Section 3. The input D contains only chunks admitted by the trusted builder. In Phases I and II, each index $u = 1, \ldots, n_c$ denotes the corresponding ordered position in D, and the same ι_u and θ determine κ_u both when registering an object and when retrieving it from the registry.

B ADDITIONAL EXPERIMENTAL RESULTS

Appendix tables follow the comparison scope stated in each caption. We bold a numerical entry only when a meaningful direction and a common comparison domain exist. Attribution, sensitivity, and error-definition tables therefore do not force a single ranking across incomparable ownership regimes, budgets, or metrics. The main paper reports the complete model–task matrix. The appendix adds controlled ablations, mechanism measurements, and length scaling. Qwen results use the frozen architecture-normalized selector on Qwen3-8B or Qwen3-14B.

B.1 Ablations and Sensitivity

The ablations test repair, selection, quantization, storage ownership, replacement semantics, global normalization, and direct execution. These mechanisms do not form independent binary factors because removing the replacement overlay also removes the support on which the selector acts. Table A1 adds matched low-bit combinations of Fusion-style and CacheBlend-style reuse. Table A2 varies the private repair budget. Table A3 freezes every sample's repair plan, while Table A4 uses a 3×3 base–patch precision design. Table A5 varies bit width and group size. Table A6 verifies disjoint replacement, global softmax, and direct composition. Table A7(a) isolates selector portability, while panel (b) isolates cross-request ownership. The complete executable-component ablation appears in Table 5.

Matched low-bit reuse baselines. Table A1 compares OneBaseKV with direct low-bit combinations of Fusion-style and CacheBlend-style reuse under aligned quality and storage protocols. OneBaseKV attains the highest macro quality among the low-bit rows. Relative to the strongest complete per-request INT4 row, it improves quality by 0.66 points while reducing persistent KV by 40.8%. With the same B4P4 storage footprint, it improves over the matched CacheBlend selector by 0.95 points.

Table A1: Matched low-bit reuse baselines on four aligned 100-example cells. Higher quality and lower two-model mean 12K fanout-16 persistent KV are better; boldface marks the best low-bit result.

Method	Qwen3-8B		Llama-3.1-8B		Overall		KV/COW
	HotpotQA	Qasper	HotpotQA	Qasper	Macro quality	Persistent KV (GiB)	
Fusion-style BF16	56.11	37.51	53.49	45.72	48.21	14.87	1.000
CacheBlend-style BF16	59.33	39.99	55.89	44.50	49.93	14.89	1.001
Fusion-style + per-request INT4	55.98	37.60	51.86	44.55	47.50	9.48	0.638
CacheBlend-style + per-request INT4	59.09	40.52	55.70	**45.69**	50.25	9.48	0.638
CacheBlend selector + matched B4P4	59.23	40.22	55.70	44.66	49.96	**5.61**	**0.377**
OneBaseKV	**63.33**	**40.82**	**56.60**	42.90	**50.91**	**5.61**	**0.377**

BF16 rows are references and are excluded from low-bit boldface. Persistent KV is the equally weighted mean of Qwen3-8B and Llama-3.1-8B physical 12K fanout-16 measurements. KV/COW normalizes each row by the corresponding two-model mean of Fusion-style BF16.

Repair budget. The main evaluation uses the prespecified $r = 0.30$ operating point. At this point, OneBaseKV attains 0.985 macro retention without enlarging the private overlay. Table A2 reports the broader sensitivity sweep from $r = 0.1$ to 0.5; B4P8 is included only as a precision sensitivity control.

Table A2: Sensitivity to the repair ratio. Retention is normalized to Full BF16 within each model–task cell. In the macro-retention column, boldface marks the stronger low-bit OneBaseKV operating point at each ratio.

Repair ratio	Representation	Qwen3-8B			Llama-3.1-8B			Macro retention
		2WikiMQA	HotpotQA	Qasper	2WikiMQA	HotpotQA	Qasper	
0.1	BF16-COW	0.600	0.939	0.868	1.040	0.885	0.994	0.888
0.1	B4P8	0.600	0.940	0.877	1.040	0.821	1.056	**0.889**
0.1	**OneBaseKV**	0.589	0.981	0.872	0.963	0.856	1.063	0.887
0.2	BF16-COW	0.766	1.049	0.994	1.065	1.008	0.975	0.976
0.2	B4P8	0.766	0.975	0.936	0.989	0.969	1.013	0.941
0.2	**OneBaseKV**	0.766	0.996	0.986	1.027	0.948	0.958	**0.947**
0.3	BF16-COW	0.870	1.010	0.946	1.117	0.939	0.969	0.975
0.3	B4P8	0.870	0.935	0.999	1.002	0.966	1.012	0.964
0.3	**OneBaseKV**	0.870	1.030	1.019	1.040	0.949	1.004	**0.985**
0.4	BF16-COW	1.038	0.998	0.944	1.065	0.922	1.042	1.001
0.4	B4P8	1.000	1.029	0.964	1.027	0.925	1.033	**0.996**
0.4	**OneBaseKV**	1.038	0.984	0.875	1.065	0.942	1.032	0.989
0.5	BF16-COW	1.000	0.999	0.951	0.871	0.924	1.063	0.968
0.5	B4P8	1.009	0.999	0.983	0.857	0.939	1.134	**0.987**
0.5	**OneBaseKV**	0.962	0.999	0.918	0.948	0.939	1.121	0.981

Each cell contains 30 samples. The main experiments fix $r = 0.30$. BF16-COW is a high-precision representation reference and is not included in the low-bit boldface comparison. At 30 samples per cell, isolated non-monotone points (e.g., Qwen3-8B Qasper at $r = 0.4$) are indistinguishable from their neighbors under paired bootstrap intervals; the prespecified $r = 0.30$ operating point is not adjusted on their basis.

Matched-mask representation isolation. All rows in Table A3 consume the same per-sample repair masks, so differences cannot be attributed to selection. Across 4 models $\times$ 4 Core-RAG tasks $\times$ 100 samples, OneBaseKV averages 44.54, close to BF16-COW's 44.65; B4P8 scores 44.40. Low-bit base and patch storage therefore introduce little additional representation loss once support is held fixed.

Table A3: Representation isolation under exactly matched per-sample repair masks. Higher scores are better. Boldface marks the stronger of the two low-bit OneBaseKV variants in each column.

Representation	2WikiMQA				HotpotQA				MuSiQue				Qasper				Macro avg.
	Q3-8B	L3.1-8B	Mis-7B	Q3-14B	Q3-8B	L3.1-8B	Mis-7B	Q3-14B	Q3-8B	L3.1-8B	Mis-7B	Q3-14B	Q3-8B	L3.1-8B	Mis-7B	Q3-14B	
BF16-COW	43.06	47.38	39.78	56.15	64.13	54.17	51.28	62.09	34.69	32.88	30.19	39.22	40.85	43.89	35.74	38.87	44.65
B4P8	39.55	**50.58**	37.93	**57.15**	62.53	55.01	**50.65**	62.55	**34.11**	30.22	**31.69**	**37.91**	**40.83**	42.57	**37.24**	**39.88**	44.40
OneBaseKV	**43.71**	49.33	**39.25**	54.89	**63.33**	**56.60**	49.31	**62.93**	33.11	**31.06**	31.31	37.85	40.82	**42.90**	36.67	39.50	**44.54**

All three representations read the identical per-sample repair plan: four models × four tasks × 100 samples. BF16-COW is a high-precision representation reference and is not included in the low-bit boldface comparison.

Base–patch precision factorization. Table A4 crosses $\mathrm{base}, \mathrm{patch} \in \{\mathrm{BF16}, \mathrm{INT8}, \mathrm{INT4}\}$ on fixed samples and masks. B4P4 provides the largest persistent-KV saving, 62.5%, while changing macro quality by only 1.25 points relative to B16P16. All nine precision pairs obtain 67.05 on Qwen3-8B HotpotQA; the record-level audit confirms identical task scores on all 30 aligned examples, showing that this subset is insensitive to base–patch precision under the frozen repair support. B4P8 remains a higher-precision control rather than a second method variant.

Table A4: 3×3 base–patch precision ablation under identical repair masks. Higher quality and larger persistent-KV saving are better. Boldface marks the best value in each quality column and the largest saving.

Representation	Qwen3-8B			Llama-3.1-8B			Overall	
	2WikiMQA	HotpotQA	Qasper	2WikiMQA	HotpotQA	Qasper	Macro avg.	KV saving
B16P16	**50.79**	**67.05**	43.35	**53.41**	64.92	42.76	53.71	0.0%
B16P8	**50.79**	**67.05**	43.35	**53.41**	64.92	43.16	53.78	8.6%
B16P4	47.46	**67.05**	42.57	50.91	65.20	41.33	52.42	14.3%
B8P16	**50.79**	**67.05**	43.16	**53.41**	64.92	42.33	53.61	28.9%
B8P8	48.96	**67.05**	43.16	**53.41**	66.03	**43.29**	53.65	37.4%
B8P4	47.46	**67.05**	**45.39**	50.91	66.03	42.12	53.16	43.2%
B4P16	**50.79**	**67.05**	44.32	52.58	67.70	**43.29**	**54.29**	48.2%
B4P8	47.46	**67.05**	44.45	52.58	**67.98**	43.08	53.77	56.7%
OneBaseKV (B4P4)	47.46	**67.05**	45.18	48.91	66.03	40.16	52.46	**62.5%**

Scores average six model–task cells with 30 samples per cell. KV saving includes payload and quantization metadata.

Bit width and group size. Table A5 varies base precision, patch precision, and quantization group size on aligned 30-sample Qwen3-8B development subsets from 2WikiMQA and HotpotQA, with the repair ratio fixed at $r = 0.30$. Quality is the task score, while logical MiB counts the payload and quantization metadata of one request before cross-request storage deduplication. The INT4-base, INT4-overlay configuration with group size 32 gives the highest two-task mean and the smallest logical representation in this local sensitivity study. The confirmatory OneBaseKV configuration retains the prespecified group size of 16 so that this development result does not alter the frozen evaluation protocol.

Table A5: Bit width and group-size ablation. Higher quality and lower logical footprint are better.

Representation	Group	2WikiMQA		HotpotQA		Mean	
		Quality	Logical MiB	Quality	Logical MiB	Quality	Logical MiB
BF16-COW	–	39.72	1411.61	69.76	2482.05	54.74	1946.83
B8P8	16	**49.72**	886.08	69.76	1555.28	59.74	1220.68
B4P8	16	46.39	616.25	69.76	1079.65	58.08	847.95
OneBaseKV	16	43.06	535.25	69.76	936.78	56.41	736.02
B8P8	32	43.06	798.37	**70.87**	1400.66	56.97	1099.52
B4P8	32	43.06	528.55	**70.87**	925.03	56.97	726.79
OneBaseKV	32	**49.72**	**447.55**	**70.87**	**782.15**	**60.30**	**614.85**

Qwen3-8B, $r = 0.30$, and 30 aligned development samples per task. Logical MiB counts payload and quantization metadata for a single request; it is not fanout-deduplicated resident memory. Boldface marks column-wise optima, including ties.

Composite-attention verification. For Qwen3-8B and Llama-3.1-8B on 2WikiMQA, MuSiQue, and Qasper, we record the first, middle, and last layers for 30 samples per task, yielding 540 layer observations and 180 generation endpoints. Let O_{cmp} denote attention over a base with replacement positions masked plus its patch under one global softmax, and let O_{mat} denote standard attention after writing the same patch into the corresponding logical positions. We report:

$$e_\infty = \|O_{\mathrm{cmp}} - O_{\mathrm{mat}}\|_\infty, \qquad e_{\mathrm{rel},2}(O_{\mathrm{var}}) = \frac{\|O_{\mathrm{var}} - O_{\mathrm{cmp}}\|_2}{\|O_{\mathrm{cmp}}\|_2 + 10^{-8}}. \tag{B.1}$$

Table A6 reports the resulting composite–materialized verification. The first metric tests the correct implementation, whereas the second compares an intentionally incorrect variant with the correct composite output. Their magnitudes are not directly comparable. The correct path agrees on all 180 top-1 endpoints and has a maximum absolute error of 4.05×10^{-6}.

Table A6: Composite–materialized numerical verification. The intended direct path and matched materialization use disjoint supports and one global softmax. Incorrect variants isolate overlapping replacement support and independent per-stream normalization. Lower errors are better.

Implementation	Error metric	540 layer records			Endpoint agreement
		Mean	Median	Maximum	
Disjoint direct composition + global softmax	e_∞	8.57×10^{-7}	4.17×10^{-7}	4.05×10^{-6}	**180/180**
Repair positions retained in base	$e_{\mathrm{rel},2}$	0.224	0.134	0.962	–
Independent per-stream softmax and summation	$e_{\mathrm{rel},2}$	1.31	1.07	3.31	–

Endpoint agreement includes identical predictions, task scores, and eight-step top-1 decisions for two models, three tasks, and 30 examples per cell. Error metrics use different references and are not compared across rows.

Selector architecture and ownership controls. Table A7(a) crosses absolute or relative-depth probes with query-head or KV-group aggregation. The two aggregation orders select identical positions under uniform GQA grouping. Relative depths instantiate without model-specific layer indices on all four architectures; their quality effect varies by model, so we use them as a portability rule rather than a same-model accuracy claim. Table A7(b) compares shared overlays with a complete per-request INT4 cache.

Table A7: **(a)** Selector architecture factorization. Scores average HotpotQA and Qasper with 30 examples per task. Absolute-all probes every layer; relative uses the two geometry-resolved depths $\{0.50N_{\text{layer}}, 0.75N_{\text{layer}}\}$.

Probe rule	Head aggregation	Qwen3-8B	Llama-3.1-8B	Mistral-7B	Qwen3-14B	Four-model mean
Absolute-all	Query-head mean	59.18	56.34	46.03	51.47	53.26
Relative depth	Query-head mean	56.11	53.84	47.00	51.30	52.06
Absolute-all	KV-group mean	59.18	56.34	46.03	51.47	53.26
Relative depth	**KV-group mean**	56.11	53.84	47.00	51.30	52.06

Under uniform GQA grouping, query-head mean and KV-group mean select identical positions for all 240 paired samples at each probe rule. Every selector cell passes the nonzero-score and query-diversity gates. The final row is used for its fixed two-probe architecture rule and KV-head semantics, not as a same-model accuracy maximizer.

Table A7: **(b)** Ownership control: All-13 quality and two-task 12K physical memory for per-request quantization versus cross-request sharing. Higher quality and more sessions per budget are better; lower persistent KV is better.

Representation	Four-model All-13	Rel. Full quality	Persistent KV (GiB)	KV/COW	Sessions per budget
BF16-COW	43.60	99.39%	15.305	1.000	0.64×
Per-request INT4 (unshared)	**43.96**	**100.21%**	9.756	0.637	1.00×
B4P8	43.50	99.18%	7.696	0.503	1.27×
OneBaseKV	43.47	99.10%	**5.774**	**0.377**	**1.69**×

Relative quality is the ratio of the unrounded four-model All-13 macro-average score to that of Full BF16, expressed as a percentage. Values above 100% are finite-sample point estimates and do not establish a systematic quality gain over Full BF16.

Lifecycle accounting. Table A8 separates persistent shared state from active generated KV. OneBaseKV minimizes both resident-state columns. These quantities define the persistent-capacity comparison; transient execution allocation is outside this table and does not enter the persistent-KV budget. The detailed measurement boundary is specified in Appendix C.2.

Table A8: Resident-state lifecycle ledger. Persistent KV includes the canonical base, request-private overlay, indices, and reference metadata; active KV additionally includes generated KV. Lower memory is better, and boldface marks the minimum in each model and column.

Model	Representation	Persistent KV (MiB)	Active KV (MiB)
Llama-3.1-8B	B16P16	1965.5	1971.4
	B4P8	850.9	856.8
	OneBaseKV	**737.6**	**743.4**
Qwen3-8B	B16P16	2236.1	2243.3
	B4P8	967.4	974.6
	OneBaseKV	**838.6**	**845.8**

B.2 Mechanism Measurements

We test whether the fixed-size repair set concentrates evaluation attention and changes across natural queries for the same document. Figure A1 summarizes both spatial concentration and cross-query variation. Let D_{tok} be the nonempty set of document positions and $N_D = |D_{\text{tok}}|$. Let $\chi_i^{\text{eval}} \geq 0$ denote attention aggregated from held-out evaluation layers and renormalized over D_{tok}, so that $\sum_{i \in D_{\text{tok}}} \chi_i^{\text{eval}} = 1$. For $\varnothing \neq R_q \subseteq D_{\text{tok}}$, define the realized repair ratio $\widehat{r}_q = |R_q|/N_D$, its evaluation-attention coverage, and enrichment as:

$$\text{cov}_q^{\text{eval}} = \sum_{i \in R_q} \chi_i^{\text{eval}}, \qquad \mathcal{E}_q = \frac{\text{cov}_q^{\text{eval}}}{\widehat{r}_q}. \tag{B.2}$$

Uniform attention gives $\mathcal{E}_q = 1$. Values above one mean that the selected positions carry more document attention than their cardinality share. If rounding yields an empty repair set, enrichment is reported as undefined.

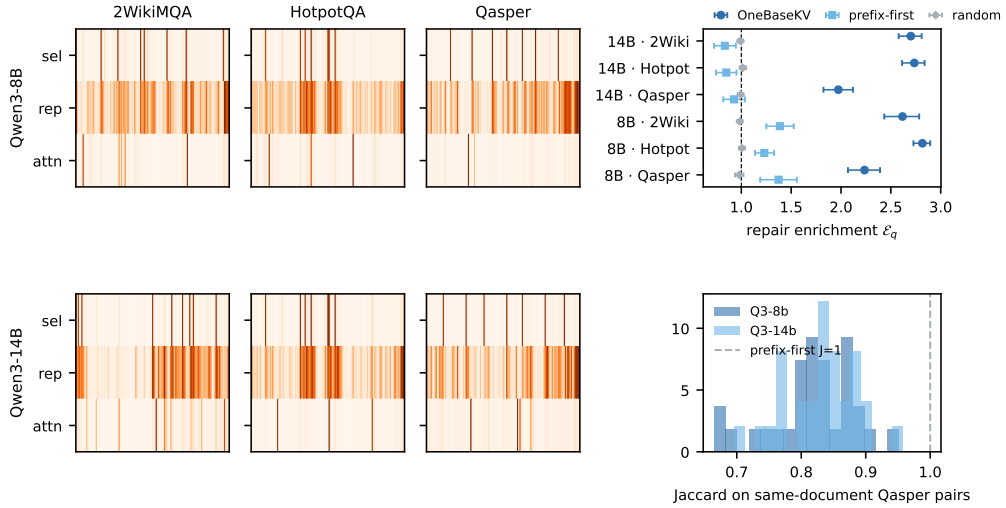


Figure A1: Query-conditioned repair patterns and cross-query support variation. The left heatmaps align selector response ("sel"), repair occupancy ("rep"), and held-out evaluation attention ("attn") for Qwen3-8B and Qwen3-14B on 2WikiMQA, HotpotQA, and Qasper. The upper-right panel compares attention enrichment with prefix-first and random selection in the same six model–task cells. Values above one indicate concentration beyond the selected-position fraction. The lower-right panel shows repair-set Jaccard for natural same-document Qasper query pairs on both models; values below one indicate query-dependent support.

Attention enrichment. Evaluation attention is taken near relative depths $0.25N_{\text{layer}}$ and $0.90N_{\text{layer}}$, distinct from the selector probes. Random masks use 20 fixed seeds. Confidence intervals use 10,000 paired sample-level bootstrap draws. OneBaseKV obtains the highest attention enrichment in all six model–task cells, and all six OneBaseKV–Random intervals have positive lower bounds. This result supports concentration on independently measured attention without equating enrichment with end-task quality. Table A9 reports the corresponding enrichment statistics and paired confidence intervals.

Table A9: Repair enrichment under attention from held-out evaluation layers. Larger values indicate that a fixed repair budget captures more attention mass.

Model	Task	n	Repair-attention enrichment			
			OneBaseKV $\mathcal{E}_q$ [95% CI]	Prefix-first	Random	OneBaseKV − Random [95% CI]
Qwen3-8B	2WikiMQA	30	**2.6168 [2.4341, 2.7845]**	1.3869	0.9855	**1.6314 [1.4473, 1.8010]**
	HotpotQA	30	**2.8166 [2.7268, 2.8945]**	1.2305	1.0056	**1.8110 [1.7229, 1.8911]**
	Qasper	30	**2.2362 [2.0700, 2.3913]**	1.3740	0.9792	**1.2569 [1.0829, 1.4225]**
Qwen3-14B	2WikiMQA	30	**2.7004 [2.5788, 2.8090]**	0.8344	0.9913	**1.7091 [1.5747, 1.8276]**
	HotpotQA	30	**2.7354 [2.6126, 2.8386]**	0.8492	1.0154	**1.7200 [1.6024, 1.8207]**
	Qasper	30	**1.9731 [1.8238, 2.1209]**	0.9258	0.9934	**0.9796 [0.8410, 1.1187]**

The target budget is $r = 0.30$. OneBaseKV is compared with prefix-first and random masks at the same realized size. All six paired OneBaseKV–Random confidence intervals have a positive lower bound.

Cross-query support variation. For natural query pairs (q, q') sharing a document, define $J(q, q') = |R_q \cap R_{q'}|/|R_q \cup R_{q'}|$. Table A10 reports the resulting cross-query support overlap. OneBaseKV's Jaccard is below the query-invariant prefix-first value of one on both Qwen3 models, directly showing that the selected repair support changes with the query.

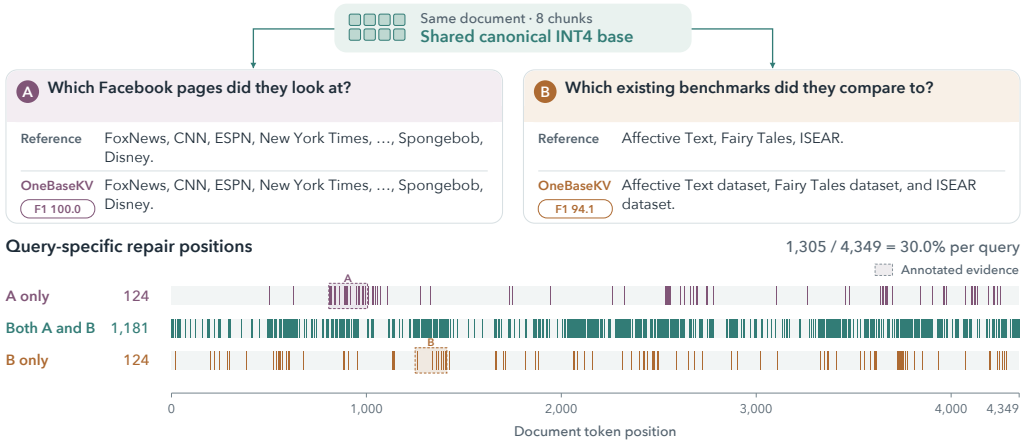


Figure A2: Two natural queries sharing one canonical INT4 base. The upper panels show each query, reference, OneBaseKV prediction, and answer F1. The three lower tracks show A-only, shared, and B-only repair positions in document-relative token coordinates. Each query's complete 30% repair support combines the shared positions with its query-only positions. Dashed boxes on the A-only and B-only tracks mark the respective annotated evidence spans, independently of which positions are selected. The case is selected by proximity to the median repair-set Jaccard, independently of answer quality.

Table A10: Repair-set overlap for natural query pairs sharing the same document. Lower Jaccard than the query-invariant prefix-first mask indicates query-dependent support.

Model	Task	Evaluation scale		Repair-set Jaccard	
		Natural query pairs	Documents	OneBaseKV [95% CI]	Prefix-first
Qwen3-8B	Qasper	30	30	**0.8158 [0.7919, 0.8385]**	1.0
Qwen3-14B	Qasper	30	30	**0.8322 [0.8124, 0.8520]**	1.0

Same-document queries and answers. We further evaluate 30 natural Qasper query pairs with Qwen3-8B and OneBaseKV. Within each pair, both queries share one canonical base and the same decoding protocol, with separate repair supports at a fixed 30% budget. Figure A2 shows the pair whose repair-set Jaccard is closest to the sample median. Its document contains 4,349 tokens across eight source chunks. Each query selects 1,305 positions, giving a realized ratio of $1{,}305/4{,}349 \approx 30.0\%$. The supports share 1,181 positions and contain 124 positions unique to each query, yielding a Jaccard of 0.826. OneBaseKV achieves answer F1 scores of 100.0 and 94.1 for queries A and B, respectively.

Table A11 summarizes all pairs. Every pair retains the same canonical base while producing different selection hashes. Mean repair-set Jaccard is 0.840, and evidence enrichment is 1.250. OneBaseKV achieves the highest F1 in this comparison, outperforming the external Fusion-style baseline by 0.43 points and the equal-budget prefix-first and random controls by 10.15 and 4.61 points. These results support query-dependent selection and evidence concentration.

Table A11: Same-document multiquery results on Qasper with Qwen3-8B. Higher F1, evidence coverage, and evidence enrichment are better. Repair-set Jaccard below one indicates query-dependent support; all 30 pairs also produce distinct selection hashes.

Method	F1	OneBaseKV F1 gain over row	Evidence coverage	Evidence enrichment	Repair Jaccard
OneBaseKV	**46.24**	Ref.	**0.375**	**1.250**	0.840
Fusion-style (paper-spec, BF16)	45.80	0.43	N/A	N/A	N/A
Prefix-first, 30%	36.09	**10.15**	0.307	1.024	1.000
Random, 30%	41.62	4.61	0.300	1.002	0.179

All 30 pairs share one canonical base within the pair and produce different selection hashes. Evidence metrics use the 55 queries with mapped gold-evidence spans. "Ref." identifies the reference row for the gain calculation. Fusion-style does not expose a repair mask under the same OneBaseKV support definition, so its evidence and overlap entries are not applicable rather than zero. Repair Jaccard is a structural overlap statistic, not a quality score. Fusion-style uses a BF16 base and patch under the paper-spec adapter, so it is not a same-budget comparison.

B.3 Length Scaling on RULER

The RULER suite (Hsieh et al., 2024) comprises `niah_single_2`, `niah_multikey_1`, `niah_multiquery`, `vt`, `cwe`, and multi-document `qa`. These tasks cover single-target retrieval, multikey retrieval, multiquery retrieval, variable tracking, global aggregation, and question answering. For this non-RAG length test, the reusable portion of each input is partitioned into 20 contiguous equal-width chunks plus one exact-prefix chunk, with chunk width $\lceil N_{\mathrm{doc}}/20 \rceil$ tokens, and processed by the same canonical-base, query-repair, and composite-attention path; query-specific suffix states remain private. Fanout is one, so the experiment isolates representation and execution as context grows. Qwen3-8B and Llama-3.1-8B use 50 samples per task; Qwen3-14B and Mistral-7B-v0.3 use 30. All four models run all six tasks at 4K, 8K, 16K, and 32K. Table A12 reports the six formal comparison methods used for this length test.

Table A12: RULER macro-average quality over six tasks by context length. Higher scores and 32K/4K retention are better. Boldface marks the best efficient method at each endpoint; Full BF16 and BF16-COW are reference rows.

Model	Method	4K	8K	16K	32K	32K/4K retention
Qwen3-8B	Full BF16	95.0	93.0	91.0	89.0	93.7%
	KIVI	**92.6**	90.0	**89.2**	**87.6**	94.6%
	ChanMix	**92.6**	**90.4**	88.4	87.2	94.1%
	PyramidKV	52.0	26.1	9.7	9.6	18.4%
	BF16-COW	89.4	90.4	88.1	85.5	95.5%
	OneBaseKV	89.4	90.3	88.0	85.2	**95.4%**
Llama-3.1-8B	Full BF16	94.3	92.4	89.5	76.0	80.6%
	KIVI	**94.2**	92.1	82.2	75.8	80.5%
	ChanMix	94.0	**92.5**	83.6	75.2	80.0%
	PyramidKV	69.1	33.4	16.2	14.9	21.5%
	BF16-COW	91.0	91.3	86.7	79.5	87.3%
	OneBaseKV	91.3	91.4	**85.8**	**79.9**	**87.5%**
Mistral-7B-v0.3	Full BF16	94.3	93.1	86.3	84.8	89.9%
	KIVI	**94.4**	89.9	78.6	73.9	78.3%
	ChanMix	93.6	89.9	80.2	74.0	79.1%
	PyramidKV	44.9	26.6	17.9	12.1	26.8%
	BF16-COW	90.9	90.1	86.1	77.7	85.5%
	OneBaseKV	90.8	**90.1**	**85.3**	**78.3**	**86.3%**
Qwen3-14B	Full BF16	96.1	95.5	93.0	93.9	97.7%
	KIVI	93.8	**94.4**	91.6	89.6	95.6%
	ChanMix	**94.8**	**94.4**	**91.7**	90.2	95.2%
	PyramidKV	54.4	26.4	15.6	14.4	26.5%
	BF16-COW	91.5	92.8	90.9	90.7	99.1%
	OneBaseKV	91.6	92.1	91.4	**91.0**	**99.3%**

Qwen3-8B and Llama-3.1-8B use 50 examples per task; Mistral-7B-v0.3 and Qwen3-14B use 30. Retention is computed from unrounded endpoint means and normalizes each method by its own 4K score.

At the four-model 32K endpoints, OneBaseKV uses 48.7% of BF16-COW's single-request persistent KV. Its 32K quality exceeds every efficient external method on Llama-3.1-8B, Mistral-7B-v0.3, and Qwen3-14B. Table A12 reports every formal quality endpoint; the persistent-KV ratio summarizes separately measured physical records at the same endpoints. On Llama-3.1-8B at 32K, both BF16-COW and OneBaseKV exceed Full BF16 (79.5/79.9 vs. 76.0). At 32K on Llama-3.1-8B, OneBaseKV remains close to BF16-COW (79.9 vs. 79.5), indicating no additional degradation from the low-bit cache representation at this endpoint.

All 23,040 quality evaluations complete successfully and pass the validity checks. They use greedy decoding with frozen operating points.

C Implementation and System Details

C.1 Exact Groupwise Quantizer

For a nonempty group Γ of finite real scalars x_g and bit width $b \geq 1$, let $N_{\mathrm{code}}(b) = 2^b - 1$, $x_\Gamma^- = \min_{g\in\Gamma} x_g$, and $x_\Gamma^+ = \max_{g\in\Gamma} x_g$. With nearest-integer rounding under a fixed tie rule, OneBaseKV uses:

$$
\begin{aligned}
s_\Gamma &= \begin{cases} (x_\Gamma^+ - x_\Gamma^-)/N_{\mathrm{code}}(b), & x_\Gamma^+ > x_\Gamma^-, \\ 0, & x_\Gamma^+ = x_\Gamma^-, \end{cases} \qquad s_\Gamma^{\mathrm{div}} = \begin{cases} s_\Gamma, & s_\Gamma > 0, \\ 1, & s_\Gamma = 0, \end{cases} \\
\zeta_{\Gamma,g}^{(b)} &= \mathrm{clip}\left(\mathrm{round}\left(\frac{x_g - x_\Gamma^-}{s_\Gamma^{\mathrm{div}}}\right), 0, N_{\mathrm{code}}(b)\right), \qquad \widehat{x}_g = x_\Gamma^- + s_\Gamma \zeta_{\Gamma,g}^{(b)}.
\end{aligned}
\tag{C.1}
$$

In this definition, s_Γ is the step size, s_Γ^{div} prevents division by zero, $\zeta_{\Gamma,g}^{(b)}$ is the unsigned code, and $\widehat{x}_g$ is its reconstruction. Under the ideal arithmetic definition in Eq. C.1, constant groups are reconstructed exactly. Ignoring storage rounding of the scale and offset, every nonconstant group obeys $|\widehat{x}_g - x_g| \le (x_\Gamma^+ - x_\Gamma^-)/(2N_{\mathrm{code}}(b))$. All metadata-rounding error is included in the perturbation envelopes of Proposition 2.

C.2 Physical Storage and Ownership Accounting

We measure physical memory under a common object lifetime. The quality–persistent-KV results appear in Table 2. Overall persistent KV is the macro average of measured physical bytes across models. KV/COW is computed within each model–task cell before averaging. Consequently, the reported ratio need not equal the quotient of two rounded GiB means, and its reciprocal is not an exact concurrency multiplier for a heterogeneous request mixture.

The physical matrix contains 192 valid records. Qwen3-8B covers all ten system configurations at fanouts 8 and 16. Qwen3-14B covers all configurations at fanout 16 and the OneBaseKV precision controls at fanout 8. Llama-3.1-8B and Mistral-7B-v0.3 fanout-16 records use the same packed-storage and storage-pointer audit. Only the default INT4-base/INT4-overlay configuration is reported as OneBaseKV outside the precision ablations.

An ownership audit inspects physical objects in 18 frozen fanout-16 execution cells. Shared chunks account for 75% of the document-chunk tokens referenced by each request. Every cell has one storage owner per shared chunk, different document orders across requests, cross-request references to the same storage, and no persistent request-level dense base copy. This audit establishes persistent ownership. The unfused implementation nevertheless uses transient per-layer workspaces.

Lifecycle accounting. Table A8 reports medians of three repetitions at 12K context on HotpotQA with fanout 1, excluding model weights. Persistent KV includes only base, patch, index, and reference metadata that survive across phases. Active KV additionally includes generated KV. Transient probe, dequantization, and attention allocations are excluded from both columns and from every persistent-capacity claim.

C.3 Reproduction Scope and Runtime Boundary

All methods use the same model, task, sample manifest, and generation settings. KIVI, ChanMix, PyramidKV, CacheBlend, and Fusion-style use source-audited official implementations or paper-specification adapters. An adapter label does not imply execution through an unmodified official runtime. The anonymous code package contains only the implementation, frozen experiment configurations, evaluation and aggregation programs, and a README with reproduction commands.

The "CacheBlend selector + matched B4P4" row in Table A1 shares the canonical store, replacement-overlay representation, and executor with OneBaseKV, so only the repair selector changes. The per-request INT4 rows instead retain the complete query-conditioned cache and isolate ownership. Runtime arms use the same 4-bit NormalFloat (NF4) double-quantized weights, generate 32 tokens per query, and perform an eight-step materialization-consistency endpoint check. NF4 double-quantized weights are used only in the runtime and trace arms of Tables 4 and A13, whereas the main quality matrix in Tables 1–3 uses BF16 weights.

C.4 Fixed-Capacity Serving Traces and Selector Validation

Fixed-capacity trace audit. Selector probing uses eager attention, generation restores scaled dot-product attention (SDPA), and online probing and patch construction are included in time to first token (TTFT). One-time canonical-base construction is recorded separately. Each seed reuses one frozen arrival manifest across Fusion-style and OneBaseKV. Both configurations pass endpoint, latency, bit-width, storage-identity, disjoint-overlay, no-dense-shadow, nonzero-score, score-variance, and selection-hash checks. Table A13 reports TTFT percentiles, SLO fractions, and transient workspace peaks. Workspace is measured separately from the persistent-KV pool. The current executor peaks at 6.6 GiB of transient workspace, while persistent-KV accounting follows the ownership boundary defined in Appendix C.2.

Per-request runtime overhead. Beyond the fixed capacity traces, we measure per request latency with the current unfused prototype across 22 endpoints. OneBaseKV-B4P4 shows a 12.7% TTFT increase over BF16-COW while saving 62.44% median resident KV. OneBaseKV-B4P8 shows a 19.2% TTFT increase with 41.76% resident KV savings.

Table A13: Fixed-capacity trace details. Values are means over three seeds. SLO fraction is the fraction of completed requests with TTFT $\leq$ 600 ms. Workspace peak is transient and excluded from the persistent-KV budget. Lower TTFT and workspace and higher SLO fraction are better. Boldface marks the best value within each workload.

Workload	Method	TTFT p50 (ms)	TTFT p95 (ms)	SLO fraction	Workspace peak (MiB)
16 documents / 48 sessions	Fusion-style	4494	4535	0.028	8280
	OneBaseKV	**958**	**2819**	**0.139**	**6663**
32 documents / 64 sessions	Fusion-style	4505	4521	0.026	8280
	OneBaseKV	**1577**	**2819**	**0.104**	**6582**

D Scope, Deployment Considerations, and Future Work

OneBaseKV targets retrieval-augmented generation workloads in which knowledge chunks recur across sessions while queries, document orders, and retrieved combinations vary. In this setting, persistent KV can limit concurrency, and recurring document state creates an opportunity for cross-request deduplication. OneBaseKV represents each reusable chunk with a shared canonical base while retaining query-conditioned differences in private replacement overlays, supporting both shared ownership and request-specific execution beyond byte-identical prefix reuse.

The current executor performs INT4 base reads, bitmap filtering, patch gathers, and online-softmax composition as separate operations. The reported TTFT includes online probing and patch construction, while one-time canonical-base construction is accounted for separately. The fixed-capacity experiments therefore evaluate the complete online cache path under the persistent-KV accounting and execution protocol defined in Appendix C.

In future work, we will extend OneBaseKV with I/O-aware kernel fusion, adaptive precision for private overlays, and runtime policies that jointly optimize repair budgets, cache residency, and request scheduling under changing workloads.

E Cache Trust Boundary and Multi-Tenant Isolation

Position-independent KV reuse can carry user-conditioned state into a public cache and create cross-request contamination or cache-hijacking paths (Zhang et al., 2026). OneBaseKV admits canonical bases only through a trusted offline builder. Query-conditioned KV, replacement overlays, and generated KV remain request-private, while public objects are immutable and verified by content digest when loaded. Table A14 tests the implementation-level contract that enforces this separation.

Table A14: Implementation tests for public-cache identity and private-overlay security invariants.

Security invariant	Implementation-level coverage	Result
An online adversarial prefix cannot rewrite a public base	Object identity, payload digest, and registry entry remain unchanged	**PASS**
Request-derived KV cannot be registered as public	Private overlays and forged-base registration are rejected	**PASS**
Any identity-field change must cause a cache miss	17 bound field classes; 18 content, order, or configuration mutations	**PASS**
Cross-tenant or cross-knowledge-base access must miss	Foreign tenant, stale ACL epoch, unknown namespace, and post-rotation access	**PASS**
Overlays are invisible across concurrent requests	Two sessions keep disjoint repairs while the base digest remains stable	**PASS**
Metadata mismatch must fail closed	Forged authentication codes and malformed metadata are rejected and rebuilt	**PASS**

A registry key binds the tenant or knowledge-base namespace, access control list (ACL) epoch, document identity, model and tokenizer configuration, chat template, positional policy, adapter, quantizer, and cache-format version. Any mismatch causes a cache miss, and distinct namespaces cannot share a physical object. A public knowledge base such as Wikipedia can nevertheless be served as its own shared namespace. Tenants of the same public namespace share physical objects, isolation

applies only across namespaces, and the gate forbids request-private state from entering public objects. These safeguards require trusted registration, access control, and integrity verification, and they do not constitute a complete adversarial defense. SHA-256 digests and hash-based message authentication code with SHA-256 (HMAC-SHA-256) authenticate identities and manifests. All six invariant suites pass.