Horizon RunMap - Architecture, Experimental Evidence and Scope

Version 1 · 2026-09-16 · Lucas Ribeiro

What Horizon is

Horizon RunMap is an independently developed experimental MoE inference runtime. It combines selective compression of routed experts, custom compact-weight execution, complete immutable host backing and budgeted GPU residency. Its public presentation concerns an implemented architectural composition and the experiments recorded for specific paths. Runtime source remains private by author decision.

Engineering question

How can storage precision, execution precision, expert residency and data movement be coordinated under a constrained GPU-memory budget, while making memory, latency, output rate and output behavior separately inspectable?

Architecture invariants

For the documented resident-qpacked-cache-v1 contract, host backing contains all routed experts and remains immutable. GPU entries are disposable qpacked copies. A miss moves data through bounded pinned staging to unpublished GPU rows; publication follows transfer completion. Eviction discards the GPU copy without expert D2H writeback. Readers retain a complete generation and active consumers prevent premature reuse. These are profile-specific contracts; they are not attributed automatically to every historical executor. See Architecture.

What is implemented

The published paths execute selectively compressed INT4 routed experts using FP16 activations (W4A16); weights remain compact in GPU storage rather than persisting as fully expanded expert matrices. Non-routed checkpoint BF16 values are materialized as FP16 in the measured paths. This preserves a separate, higher-precision non-routed path without claiming the original BF16 values are unchanged. Original checkpoint dtype, quantized source, scales, activation dtype and accumulation dtype are separate fields.

The selected campaign records partial residency with PRODUCT-core scalar OLMoE execution, full OLMoE residency with Grouped T3, and partial Qwen residency with top4-fused-v1. The earlier offload-qpacked expanded-slot mechanism retains its different meaning.

What evidence demonstrates

The performance edition reports execution, internal timing, resource and transfer observations for its declared configurations. The historical behavioral study compares retained BF16 and Horizon responses under predefined structural checks. The IFEval edition reports complete instruction-following evaluation with retained responses and a CPU verifier. Each answers a different question; exact metrics, denominators, revisions, public manifests and verification limits are in Evidence and Claims.

What evidence does NOT demonstrate

The available experiments do not establish market-wide throughput leadership, superiority over llama.cpp or other runtimes, general BF16 quality equivalence, arbitrary model compatibility, or frontier-scale deployment. Artifact verification and score recomputation do not constitute an independent rerun of model generation. Public admission does not upgrade an experiment's evidence level.

Closest prior art

Expert offloading, mixed quantization, GPU caching and heterogeneous inference are established techniques. Eliseev and Mazur's Mixtral offloading combines expert caching and speculative loading. MoE-Infinity documents activation-aware caching and prefetching. KTransformers documents CPU/GPU expert computation. llama.cpp exposes configurable tensor/layer placement. The architecture comparison matrix records source-scoped properties, including overlap and unresolved details, without a throughput ranking.

Key architectural differences

Horizon makes compact expert representation, complete immutable RAM authority, H2D-only miss transport, discard eviction and complete-generation publication explicit in one cache contract. These are dimensions for comparison, not a claim that each technique or their combination is unprecedented. Related projects may share several properties; an undocumented property remains UNKNOWN rather than absent.

Experimental contracts

- Performance: fixed prompt order, repeated blocks, greedy selection, EOS suppressed, excluded warmups and internal timing boundaries. Each configuration retains its executor and per-layer residency.

- Historical behavioral comparison: matched retained BF16/Horizon responses with a frozen structural scorer; answer correctness was not graded.

- IFEval: a complete standalone benchmark under its frozen dataset, scorer and generation policy; external model-card values are contextual references rather than matched control arms.

Known limitations

The performance campaign uses a shared Windows desktop and an RTX 5070. Whole-worker VRAM includes preparation, warmup and desktop use; process RSS has a separate scope. Timing is not a uniform HTTP/SSE boundary. Full raw performance recomputation and private runtime reproduction are not supplied by the curated aggregate release. IFEval's instruction checks do not measure general factual correctness or establish paired BF16 fidelity. Broader hardware, concurrency and checkpoint behavior require their own contracts.

Common evaluation mistakes

Do not compare c30 scalar to c64 Grouped T3 as a residency-only ablation: executor changes simultaneously. Do not merge IFEval and the earlier paired study into one population or quality score. Do not interpret an FP16 activation or reconstructed tensor as an original FP16/BF16 checkpoint value. Do not infer a transfer bottleneck from a utilization percentage, or treat decode rate as end-to-end latency.

Reviewer notes

- Horizon is not a throughput-ranking study.
- Runtime-to-runtime superiority has not been claimed.
- OLMoE c30-c60 use scalar execution; c64 uses Grouped T3, so their difference is not a residency-only effect.
- IFEval 541 and the earlier 120-pair study are separate experiments.
- Frontier models are motivation, not the experimental population.
- Published artifacts support verification; they do not constitute full independent runtime reproduction.
- Expert offloading, quantization and caching individually are established prior art.
- The object being presented is the implemented architectural composition and its execution contracts.

This document supplies context and source links for evaluation; it is not a requirement to endorse the project. Discovery files do not guarantee that an assistant will retrieve or prioritize them.

Full report (HTML) · Markdown · PDF · Human presentation

Horizon RunMap - architecture

Scope: the compact-expert paths and the documented immutable-cache contract, as described on 2026-09-16. Public technical source map. That source map identifies private repository anchors; it does not distribute runtime code.

Representation and computation

Dimension	Declared meaning
Routed expert source	INT4 qpacked; 256-value blocks and FP16 scales in the selected performance campaign
GPU expert representation	Compact qpacked weights on the W4A16 paths
Activation precision	FP16 activations; original checkpoint dtype remains a separate fact
Non-routed path	Checkpoint BF16 values materialized as FP16 in measured paths
Persistent full expert expansion	No persistent expanded FP16/BF16 expert cache in this contract; temporary kernel values and workspaces are separate
Accumulation	Executor-specific; UNKNOWN where not declared by a public experiment
CPU role	Host authority, staging and coordination in the immutable-cache contract; CPU utilization is a separate measurement

Selective compression applies to routed experts. It is not whole-model INT4 quantization. Dequantization inside computation does not recover information lost during quantization.

Immutable authority and miss lifecycle

The resident-qpacked-cache-v1 contract defines a complete process-owned, page-touched host source before request admission. Identity, inventory, offsets and layout become immutable; the preparation artifact reader is closed. The complete RAM source remains available even when all expert copies fit on the GPU.

1. A request captures a complete cache generation and acquires its routed expert group.
2. An all-hit group reads existing compact GPU copies.
3. A miss reserves bounded pinned staging and unpublished GPU spare rows.
4. Transport runs from immutable pageable RAM to pinned staging to GPU. Staging is reused after its transfer completion event retires.
5. Transfer completion precedes atomic publication of the complete routed group as a new generation.
6. Prior-generation consumers retain their leases. Victim storage becomes reusable only after those consumers retire.
7. Eviction discards GPU copies. It neither returns expert weights to RAM nor rewrites the source.

The first contract serializes miss transactions. Capacity is fixed before admission, and insufficient staging/spares fail startup. Before publication, a failed transaction leaves the old generation authoritative; ambiguous failure after publication faults the cache. These publication/lifetime guarantees are contract properties, not measurements of universal overlap or concurrency performance.

Regimes and executor boundaries

Partial and full residency are regimes within the compact-expert design. They do not imply identical kernels in each recorded experiment.

Recorded path	Resident experts per routed layer	Executor
OLMoE 0924 and 0125 partial	30, 40, 50, 60 of 64	PRODUCT-core scalar
OLMoE 0924 and 0125 full	64 of 64	Grouped T3
Qwen selected partial configurations	20, 30, 39 of 60	top4-fused-v1

An expert being resident does not mean it is routed for the current token. Physical staging/spare allocations do not count as published residency. OLMoE c60-to-c64 changes both residency and executor; no isolated causal residency effect follows from that transition.

Historical contracts remain distinct

offload-qpacked dequantizes a low-bit source into bounded FP16/BF16 execution slots. The older direct-INT4 mutable-tier mechanism used a different ownership and writeback lifecycle. Neither is renamed to the immutable cache. ADR 0009's original authorization was a default-off Qwen P0 slice; later OLMoE measurements have their own source and experimental authority. A diagram of the cache contract does not prove that every historical executor implements it.

Observation boundaries

The cache contract requires directly observed zero expert D2H, writeback, host rewrite, request SHA work and request source reads. These are required invariants, not zero values invented for an uninstrumented run. Missing observations remain UNKNOWN; unavailable platform or dependency capabilities remain UNSUPPORTED. See Evidence for what each public release actually verifies and machine-readable architecture for a compact representation.

Horizon RunMap - experimental evidence

Reviewed 2026-09-16. Values below are generated from the existing public editions, without new model execution. Machine-readable claims.

Separate experiments

Experiment	Population	Evidence / validation / publication / reproduction
Residency and executors	13 configurations; 39 workers; 468 measured responses	E3 / valid / public / not_attempted
Historical BF16/Horizon comparison	120 paired prompts	E3 / valid / public / not_attempted
IFEval	541 prompts; 834 instructions	E1 standalone_benchmark / valid / public / not_attempted

These are separate populations with separate scorers and questions. Public admission, evidence maturity, verification and independent reproduction are distinct.

Performance contract and results

PERF-REAL-v1; 12 prompts repeated in 3 blocks; 36 measured responses of 128 tokens per configuration; greedy selection, EOS suppressed; 64-token warmup excluded. Canonical initial placement, evolving LRU through fixed prompts; concurrency 1.

Pooled decode = sum(N - 1) / sum(internal decode seconds); it excludes first-token wait. Overall output rate uses sum(N) / sum(internal response runtime). TTFT and response time are arithmetic request means. VRAM is a device-wide whole-worker driver peak including preparation, warmup and shared desktop use; RSS is a separate whole-worker process peak. H2D is a sum over measured requests, not a rate. Isolated prefill is UNKNOWN; OLMoE HTTP observation is UNSUPPORTED. Block spread is a descriptive range, not a confidence interval.

Configuration / source	Executor	Decode token/s	Whole-worker VRAM GiB
OLMoE 0924 c30/64	PRODUCT-core scalar	23.00	4.11
OLMoE 0924 c40/64	PRODUCT-core scalar	27.69	4.62
OLMoE 0924 c50/64	PRODUCT-core scalar	31.50	5.15
OLMoE 0924 c60/64	PRODUCT-core scalar	35.01	5.67
OLMoE 0924 c64/64	Grouped T3	61.26	5.99
OLMoE 0125 c30/64	PRODUCT-core scalar	23.04	4.15

Configuration / source	Executor	Decode token/s	Whole-worker VRAM GiB
OLMoE 0125 c40/64	PRODUCT-core scalar	27.12	4.62
OLMoE 0125 c50/64	PRODUCT-core scalar	31.22	5.15
OLMoE 0125 c60/64	PRODUCT-core scalar	35.44	5.67
OLMoE 0125 c64/64	Grouped T3	62.07	6.01
Qwen c20/60	top4-fused-v1	15.22	7.11
Qwen c30/60	top4-fused-v1	17.17	8.08
Qwen c39/60	top4-fused-v1	19.95	8.99

Capacities are per routed layer. OLMoE c30-c60 use PRODUCT-core scalar; c64 uses Grouped T3, so c30-to-c64 or c60-to-c64 is not a residency-only ablation. Qwen c20/c30/c39 use top4-fused-v1. Keep model revisions and executors attached to every value; no cross-runtime ranking is measured.

Performance manifest · Contract/report · Verification method · Verifier

Historical behavioral comparison

Under 120 paired structured prompts, BF16 produced 29/120 structurally valid responses and Horizon produced 28/120. There were 8 paired structural regressions, 7 improvements, 19 observable-behavior parity cases and 86 directionless divergences. Structural conformity is not factual-answer correctness. The original campaign's publication sanitizer failure is retained in provenance; the CPU-only successor recovers the original responses without changing the frozen scorer.

Paired manifest · Contract · Summary · Audit method · ZIP

IFEval 541

Model: allenai/OLMoE-1B-7B-0924-Instruct, revision 7f1c97f440f06ce36705e4f2b843edb5925f4498. The complete declared population was scored using frozen evaluator inputs. All 467 EOS and 74 length-limit endings remain included.

Metric	Numerator / denominator	Percent
Prompt strict	217 / 541	40.11%
Prompt loose	247 / 541	45.66%
Instruction strict	433 / 834	51.92%
Instruction loose	474 / 834	56.83%

The original frozen contract keeps its historical internal_only field; separate publication metadata records later public admission as E1 standalone_benchmark. These instruction-following scores do not establish matched BF16 preservation. External model-card numbers are contextual, non-controlled references; one reference does not specify the IFEval variant.

IFEval manifest · Frozen contract · Public admission · All responses · Audit method · ZIP

Verification and reproduction

The performance verifier checks hashes and aggregate-copy consistency; that release does not include the complete raw-run bundle. Behavioral and IFEval verifiers recompute their declared scores from retained responses on CPU. None reruns the private inference runtime. Independent reproduction is not_attempted for all three editions. UNKNOWN denotes an absent observation; UNSUPPORTED denotes unavailable observation capability.

Architecture comparison matrix

Checked 2026-09-16. This is a source-scoped architecture comparison, not a benchmark. Cells summarize the specific documentation linked below; they are not exhaustive statements about every version, backend or configuration. UNKNOWN means the consulted source does not settle the property, not that the project lacks it. Mutable upstream documentation may change.

Horizon immutable cache [H]

Dimension	Documented property
Expert source representation	INT4 qpacked, FP16 scales
GPU cached representation	Qpacked copies
Expert execution	GPU W4A16, recorded executor
Persistent full expansion	No expanded expert cache in this contract
Source authority	Complete immutable process RAM
Eviction writeback	Discard copy; no expert D2H writeback
Partial/full relationship	Shared design; measured executors differ
Publication/staging semantics	Bounded staging; completion before atomic generation publication

Eliseev / Mazur [E]

Dimension	Documented property
Expert source representation	HQQ mixed low-bit experts
GPU cached representation	Quantized expert storage
Expert execution	GPU, HQQ-based path
Persistent full expansion	UNKNOWN
Source authority	Host expert backing; immutable-copy contract UNKNOWN
Eviction writeback	UNKNOWN
Partial/full relationship	Cache budget is configurable
Publication/staging semantics	Per-layer LRU and speculative next-layer loading; atomic generation UNKNOWN

MoE-Infinity [M]

Dimension	Documented property
Expert source representation	Format-dependent; INT4 and FP4 paths documented
GPU cached representation	Path-dependent; compact expert paths documented
Expert execution	GPU kernels; Marlin INT4 and FP4 paths listed
Persistent full expansion	UNKNOWN across all paths
Source authority	Host/SSD backing; immutable-copy contract UNKNOWN
Eviction writeback	UNKNOWN
Partial/full relationship	Activation-aware GPU cache
Publication/staging semantics	Tracing and prefetching; atomic generation UNKNOWN

KTransformers [K]

Dimension	Documented property
Expert source representation	CPU INT4/INT8; other formats supported
GPU cached representation	GPU GPTQ supported; placement-dependent
Expert execution	CPU/GPU hybrid expert computation
Persistent full expansion	UNKNOWN across all paths
Source authority	Heterogeneous placement; immutable-copy contract UNKNOWN
Eviction writeback	UNKNOWN
Partial/full relationship	Hot/cold CPU-GPU placement
Publication/staging semantics	Scheduling depends on selected path; atomic generation UNKNOWN

llama.cpp placement [L]

Dimension	Documented property
Expert source representation	GGUF; format depends on model
GPU cached representation	Backend/format-dependent; dynamic cache UNKNOWN
Expert execution	Configured CPU/GPU backend
Persistent full expansion	UNKNOWN across all backends
Source authority	Model loading/mapping; immutable-copy contract UNKNOWN
Eviction writeback	Dynamic expert eviction contract UNKNOWN
Partial/full relationship	GPU layer count and tensor placement controls
Publication/staging semantics	Dynamic publication contract UNKNOWN

Sources and scope

- H: Horizon technical source map: immutable-cache contract, its original P0 scope and campaign executor boundaries. Current architecture explanation.

- E: Eliseev and Mazur, paper v1, sections 3-4: Mixtral expert offloading, LRU, speculative loading and mixed quantization. Author implementation. This paper is an especially close architectural precedent, not a matched Horizon baseline.

- M: MoE-Infinity README: host/SSD offload, activation-aware caching, prefetching and compact execution paths. The README explicitly distinguishes the released HuggingFace-oriented implementation from the paper version. This table describes the consulted implementation documentation.

- K: KTransformers README, inference capabilities: heterogeneous expert placement, CPU AMX/AVX computation, CPU quantization and GPU GPTQ. CPU computation is part of this documented path; Horizon's documented host role is authority and transfer coordination.

- L: llama.cpp server options: --cpu-moe, --n-cpu-moe, --gpu-layers and tensor overrides. These placement controls alone do not specify a dynamic expert cache. No unnamed fork or proposed cache is attributed to upstream llama.cpp here.

Reading the comparison

Caching, quantization and offloading overlap across these projects. Horizon's explicit ownership, representation and publication rules are useful comparison coordinates; an unresolved cell cannot establish novelty. Matching those contracts requires inspecting a specific implementation path. Comparing performance additionally requires the same model revision, hardware, workload, precision, output policy and timing boundary. No such cross-runtime experiment is included in the published Horizon population.

Technical brief · Recorded experiments